

SavaPage User Manual

Version 1.0.0



Rijk Ravestein

SavaPage User Manual : Version 1.0.0



SavaPage User Manual by Rijk Ravestein is licensed under [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/)¹.

SavaPage Open Print Portal Software by [Dataverse B.V.](https://www.dataverse.com)² is [OSI Certified Open Source Software](https://opensource.org/)³, licensed under the terms of the [GNU Affero General Public License \(AGPL\)](https://www.gnu.org/licenses/agpl.html)⁴ version 3, or any later version, in compliance with [Third Party Software Licenses](https://www.savapage.org/docs/licenses/)⁵. “OSI Certified” is a certification mark of the [Open Source Initiative](https://opensource.org/)⁶.

¹ <https://creativecommons.org/licenses/by-sa/4.0/>

² <https://www.dataverse.com>

³ <https://opensource.org/>

⁴ <https://www.gnu.org/licenses/agpl.html>

⁵ <https://www.savapage.org/docs/licenses/>

⁶ <https://opensource.org/>

Table of Contents

Preface	xix
1. About this Manual	xix
2. Expectations and Prerequisites	xix
3. Conventions used in this Document	xix
3.1. Typographical Conventions	xix
3.2. Notes	xx
4. Notice	xxi
5. Your Feedback	xxi
1. Introduction	1
1.1. What is SavaPage?	1
1.1.1. Open Source Software	1
1.1.2. Benefits	1
1.1.3. Key Features	2
1.2. System Requirements	3
1.2.1. Server	3
1.2.2. Clients	5
1.3. How does SavaPage work?	6
1.3.1. Key Concepts	6
1.3.2. The SavaPage Work Flow	8
1.3.3. Architecture Overview	9
2. Server Installation	11
2.1. Step 1 - System Requirements	11
2.2. Step 2 - Create System Account	11
2.3. Step 3 - Configure CUPS and Samba	12
2.3.1. CUPS Remote Printer Browsing	13
2.3.2. CUPS Job History	13
2.3.3. CUPS Job Privacy	14
2.3.4. CUPS Web Interface	14
2.3.5. CUPS systemd service	14
2.3.6. Test CUPS	15
2.4. Step 4 - Optional System Settings	15
2.4.1. Set Default Paper Size	15
2.5. Step 5 - Check Firewall Settings	15
2.6. Step 6 - Download and Install	16
2.7. Step 7 - Save Encryption Keys	16
2.8. Step 8 - Configure	17
2.8.1. Step 1 - Login	17
2.8.2. Step 2 - Change Admin Password	17
2.8.3. Step 3 - Set Locale	17
2.8.4. Step 4 - Set Currency Code	17
2.8.5. Step 5 - Set User Source	18
2.8.6. Step 6 - User Synchronization	18
2.8.7. Step 7 - Set Mail Options	18
2.8.8. Step 8 - Driverless Printing	18
2.9. Step 8 - Share SavaPage Client Files	19
2.10. Step 9 - Testing	19
2.11. What's next?	19
3. User Web App	22
3.1. Login	22
3.1.1. About	23
3.1.2. Select Language	23

3.1.3. Single Web App Session	24
3.1.4. Login Alternatives	24
3.1.5. Card Self Association Dialog	26
3.2. SafePages	26
3.2.1. Document Expiration	29
3.2.2. Footer	29
3.2.3. Document Details	32
3.2.4. Browser	34
3.3. PDF	35
3.3.1. PDF Filters	37
3.3.2. Document Scope	37
3.3.3. Description	37
3.3.4. Security	38
3.3.5. Passwords	39
3.3.6. Letterhead	40
3.3.7. Download	40
3.3.8. Send	40
3.4. Print	41
3.4.1. Printer Selection	41
3.4.2. Selected Printer	42
3.4.3. Delegated Print	42
3.4.4. Print Job Settings	42
3.4.5. Direct Print Release	45
3.4.6. Full Print Scope and Jobs	46
3.4.7. Delegated Print Edit	46
3.4.8. Job Ticket Print	51
3.5. Printer Settings	54
3.5.1. Page Scaling	55
3.5.2. Print Preview	56
3.5.3. Printer Setting Options	57
3.6. Letterheads	57
3.7. Delete	59
3.8. Log	60
3.8.1. Documents	60
3.8.2. Transactions	61
3.9. Sort	65
3.10. User Details	66
3.10.1. Internet Printer	67
3.10.2. Pagometers	67
3.10.3. Financial	68
3.10.4. Redeem Voucher	68
3.10.5. Transfer Credit	68
3.10.6. Transfer Money	69
3.10.7. Send Bitcoins	69
3.11. Upload	70
3.11.1. Upload Dialog	70
3.11.2. Upload Drop Zone	72
3.12. GDPR Dialog	72
4. Admin Web App	74
4.1. Login	74
4.2. Menu	74
4.3. Dashboard	76
4.3.1. Status	77
4.3.2. Services	79

4.3.3. News	79
4.3.4. Pagometers	79
4.3.5. Environmental Impact	80
4.3.6. Financial Summary	80
4.3.7. Activity	81
4.4. Users	81
4.4.1. User List	81
4.4.2. Download Personal Data	83
4.4.3. Erased Users	84
4.4.4. Edit User	84
4.4.5. Create Internal User	89
4.4.6. Deleted Users	90
4.4.7. Administrator Role	90
4.5. Groups	90
4.5.1. Built-in Groups	90
4.5.2. Group List	91
4.5.3. Add & Remove Groups	92
4.5.4. Edit Group	93
4.6. Accounts	96
4.6.1. Account List	96
4.6.2. Edit Account	97
4.7. Queues	99
4.7.1. Reserved Queues	99
4.7.2. Queue List	99
4.7.3. Edit Queue	101
4.8. Proxy Printers	101
4.8.1. Proxy Printer List	101
4.8.2. Edit Proxy Printer	105
4.8.3. Printer Groups	109
4.8.4. Rename Proxy Printer	110
4.9. Devices	110
4.9.1. Network Card Reader	113
4.9.2. Terminal	117
4.9.3. Custom User Login	118
4.10. Options	120
4.10.1. User Source	120
4.10.2. User Creation	124
4.10.3. User Authentication	126
4.10.4. Mail	130
4.10.5. PaperCut Integration	132
4.10.6. Google Cloud Printer	133
4.10.7. Mail Print	137
4.10.8. Web Print	139
4.10.9. Internet Print	140
4.10.10. Proxy Print	141
4.10.11. Eco Print	143
4.10.12. Financial	145
4.10.13. Backups	148
4.10.14. Advanced	149
4.11. Documents	157
4.12. Log	161
4.13. About	163
4.13.1. Version	164
4.13.2. License	165

4.13.3. Community	165
4.13.4. Support	166
4.13.5. Java	167
4.13.6. Host System	167
4.13.7. Host Packages	167
4.14. Vouchers	168
4.14.1. Voucher Actions	170
4.14.2. Create Vouchers	172
4.14.3. Voucher Usage	172
5. Job Tickets Web App	174
5.1. Open Tickets	174
5.1.1. Select and Sort	174
5.1.2. Ticket List	175
5.1.3. Job Ticket Bulk Actions	175
5.1.4. Job Ticket Edit	176
5.1.5. Job Ticket Print	177
5.1.6. Job Ticket Settle	180
5.2. Closed Tickets	180
5.3. Ticket Configuration Properties	181
6. Point-of-Sale Web App	183
6.1. Deposit	183
6.2. Receipts	184
7. User Client	186
7.1. User Client Options	187
7.2. User Client Deployment	188
7.2.1. Deployment on Windows	188
7.2.2. Deployment on macOS	188
7.2.3. Deployment on GNU/Linux	188
8. SavaPage Financial	189
9. SavaPage on GNU/Linux	190
9.1. The Installation Process	190
9.1.1. Manual extraction	190
9.1.2. The install process	190
9.2. Logs	192
9.3. Advanced Configuration	192
9.3.1. Alternative TCP/IP Settings	193
9.3.2. Alternative File Locations	193
9.3.3. Miscellaneous Settings	193
9.4. Upgrading SavaPage	194
9.5. Removing SavaPage from a GNU/Linux server	194
10. SavaPage as Printer	195
10.1. Printing with a Driver	195
10.1.1. SavaPage Printer Driver	195
10.1.2. SavaPage Printer Installation	195
10.2. Printing with AirPrint	199
10.2.1. Step 1: Enable IPv4 in Avahi	199
10.2.2. Step 2: Create AirPrint Queue	199
10.2.3. Step 3: Create Avahi Service File	200
10.3. Printing from iOS	200
10.3.1. Step 1: Install iOS Web Clip	200
10.3.2. Step 2: Test	201
10.4. Printing from Android and Chrome OS	202
10.4.1. SavaPage Google Cloud Ready Printer	202
10.4.2. PrinterShare™ Mobile Print	203

10.5. Driverless Printing	203
10.6. IP Restricted Printing	204
10.6.1. CIDR Notation	204
10.6.2. CIDR Set	204
10.7. Printing Encrypted PDF	204
11. Authenticated Printing	206
11.1. Key Concepts	206
11.1.1. User	206
11.1.2. Person	206
11.1.3. Abstract User	206
11.1.4. Domain User	206
11.1.5. Synchronized User	206
11.1.6. Synchronized Person	206
11.1.7. Internal Person	206
11.1.8. Authenticated User	206
11.1.9. Authenticated Abstract User	207
11.1.10. Authenticated Person	207
11.1.11. Trusted SavaPage Queue	207
11.1.12. Public SavaPage Queue	207
11.1.13. IP Based Authentication	207
11.1.14. Mail Print Authentication	208
11.1.15. Local User	208
11.1.16. Local Abstract User	208
11.1.17. Local Person	208
11.1.18. User Alias	208
11.2. Single Sign-On Domains	208
11.2.1. Authentication Loopholes	209
11.2.2. Unauthenticated Users	210
11.3. Peer to Peer Networks	211
11.4. User Name Aliases	212
12. Printing Impact	214
12.1. Financial Impact	214
12.2. Environmental Impact	214
12.2.1. Printed Sheet Units	214
12.2.2. Trees	215
12.2.3. Energy	215
12.2.4. Carbon	215
13. Security	216
13.1. User Authentication	216
13.1.1. Login Passwords	216
13.1.2. PIN Codes	216
13.1.3. Authentication Tokens	216
13.1.4. One-time Authentication Tokens	217
13.1.5. User Dialog	217
13.2. Access over Internet	217
13.3. Web Sessions	217
13.3.1. Web Session Timeout	217
13.3.2. Web Session Cookies	218
13.4. SSL Passwords	218
13.5. Secured JMX Connection	218
13.6. Encrypted Secrets	218
13.7. Document Signature	219
13.8. User Client	219
13.9. Server Commands	220

13.10. Web Services	220
13.11. Log Files	220
13.12. Network Card Reader	220
13.13. Internal Services	221
13.14. External Services	221
13.14.1. Google Cloud Print Service	221
13.15. Vouchers	221
14. Privacy	223
14.1. Open Source	223
14.2. General Data Protection Regulation	223
14.2.1. Data Portability	223
14.2.2. Data Erasure	223
14.3. Secure Print Release	223
14.4. CUPS Privacy	224
15. Internationalization	225
15.1. Localization	225
15.2. Internal Fonts	225
15.2.1. Default Font	226
15.2.2. CJK Font	226
15.2.3. Unifont	226
16. Customization	227
16.1. Custom Web App	227
16.1.1. Web App Look-and-feel	227
16.2. Email Templates	230
16.2.1. Email Template Syntax	230
16.2.2. Email Stationary Template	232
16.2.3. Email Message Template	232
16.2.4. Email Placeholders Objects	233
16.2.5. Email Stationary Types	234
16.2.6. Email Message Types	234
16.2.7. Custom Template Locations	235
17. Using an External Database	236
17.1. Supported Databases	236
17.2. Migrating to an External Database	236
17.2.1. Step 1 - Stop SavaPage	236
17.2.2. Step 2 - Create a Backup	237
17.2.3. Step 3 - Create new Database in External DBMS	237
17.2.4. Step 4 - Change SavaPage Connection Parameters	237
17.2.5. Step 5 - Initialize new Database	237
17.2.6. Step 6 - Restore Backup into new Database	238
17.2.7. Step 7 - Restart SavaPage	238
18. Tuning	239
18.1. Linux Kernel Parameters	239
18.1.1. IP Ports	239
18.1.2. TCP Buffer Sizes	240
18.1.3. Queue Sizes	240
18.1.4. Congestion Control	240
18.1.5. Setting Linux kernel parameters with sysctl	241
18.2. Linux User Limits	241
18.3. JVM Tuning	242
18.3.1. JVM Memory Allocation	242
18.3.2. JVM Garbage Collection	243
18.3.3. JVM Temporary Files	243
19. SavaPage Community	245

19.1. Visitor Period	245
19.2. Registered Member	245
19.3. Importing the Member Card	246
A. Proxy Print Scenarios	247
A.1. Personal Print Scenarios	247
A.1.1. Non-secure Print Scenario	247
A.1.2. Authenticated Print Scenarios	247
A.2. Delegated Print Scenarios	247
A.2.1. Delegated Print - (Non) Secure & Job Ticket Scenarios	247
A.2.2. Delegated Print - Job Ticket - PaperCut - Scenario	248
A.2.3. Delegated Print - PaperCut Scenario	249
B. NFC Authentication	251
B.1. Card Number Format	251
B.2. Local Card Reader	251
B.3. Network Card Reader Service	252
C. Tools	253
C.1. Server Commands	253
C.1.1. Common Options	254
C.1.2. addInternalUser	256
C.1.3. addUserGroup	257
C.1.4. changeBaseCurrency	257
C.1.5. deleteUser	258
C.1.6. deleteUserGroup	258
C.1.7. eraseUser	258
C.1.8. getConfigProperty	259
C.1.9. listUsers	259
C.1.10. listUserGroups	260
C.1.11. listUserGroupMembers	260
C.1.12. listUserGroupMemberships	260
C.1.13. listUserSourceGroups	261
C.1.14. listUserSourceGroupMembers	261
C.1.15. listUserSourceGroupNesting	261
C.1.16. printerAccessControl	262
C.1.17. printerSnmp	262
C.1.18. setConfigProperty	263
C.1.19. setUserProperties	263
C.1.20. setUserGroupProperties	264
C.1.21. syncUserGroup	265
C.1.22. syncUsersAndGroups	265
C.1.23. systemStatus	266
C.2. Web Services	266
C.2.1. XML-RPC	266
C.2.2. JSON-RPC	267
C.3. Database Commands	270
C.3.1. db-config-get	270
C.3.2. db-config-set	271
C.3.3. db-delete-logs	271
C.3.4. db-export and db-export-to	271
C.3.5. db-import	271
C.3.6. db-init	272
C.4. Stopping and Starting the Server	272
C.5. SSL Key Generation	273
C.5.1. Re-Create the Self-Signed Certificate	273
C.5.2. Importing an Existing SSL Certificate	273

C.5.3. Installing the Keystore	274
D. Capacity Planning	276
D.1. Database Sizing and Growth	276
D.2. SafePages Sizing and Growth	277
D.3. Network Bandwidth Planning	277
E. URL Cheat Sheet	278
F. File Locations	280
G. Printable File Types	282
G.1. Standard File Types	282
G.1.1. XPS to PDF Installation Instructions	283
G.2. Advanced File Types	283
H. Upgrading from a Previous Version	285
H.1. Upgrading the Server	285
H.2. Upgrading Client Printer Drivers	285
H.3. Testing the Upgrade	285
I. Migrating to a New Server	286
I.1. Upgrade Old Server	286
I.2. Install New Server	286
I.3. Migrate Data to New Server	286
I.4. Rename Printers	287
I.5. Update SavaPage Printers	287
J. Advanced LDAP Configuration	288
J.1. LDAP Server Default Configuration	288
J.1.1. OpenLDAP	289
J.1.2. Apple Open Directory	289
J.1.3. Novell eDirectory Defaults	290
J.1.4. Microsoft Active Directory Defaults	290
K. PPD Extensions	292
K.1. PPD to IPP Mappings	292
K.1.1. Mapping PPD to IPP	292
K.1.2. Mapping PPD to IPP Extensions	294
K.1.3. Restricting Standard Options	294
K.2. PPD Rules	294
K.2.1. Generic PPD Rules	294
K.2.2. Custom PPD Rules	296
K.3. IPP Rules	298
K.3.1. SPConstraint	298
K.4. Job Ticket Extensions	299
K.4.1. Job Ticket Media Options	300
K.4.2. Job Ticket Sheet Options	301
K.4.3. Job Ticket Copy Options	302
K.4.4. Job Ticket Set Options	303
K.5. Tips and Tricks	303
K.5.1. Fast Print A4 and Letter to Single Tray	303
K.6. proxy-print.log	304
L. IPP Extensions	305
L.1. Internal IPP Extensions	305
L.1.1. Internal IPP - PPD Mapping Extensions	305
L.1.2. Internal IPP Job Ticket Extensions	307
L.1.3. Internal IPP Marker	309
L.2. External IPP Extensions	309
L.3. IPP Localization	310
M. SavaPage Plug-ins	311
M.1. Web API Callback Plug-in	311

M.1.1. Payment Gateway Plug-in	311
M.2. OAuth Client Plug-in	314
N. PaperCut Integration	315
N.1. Delegated Print to PaperCut	315
N.1.1. PaperCut Configuration	315
N.1.2. PaperCut Delegated Print Processing	316
N.1.3. PaperCut Delegated Print Accounting	317
N.1.4. PaperCut Queries and Reports	318
N.1.5. Integration Pitfalls	319
N.2. PaperCut Integration Limitations	319
O. Job Scheduling	320
O.1. Cron Trigger Format	320
P. GNU Affero General Public License (AGPL)	321

List of Figures

1.1. SavaPage High-Level Architecture	9
2.1. CUPS Job Privacy	14
3.1. Web App: Login Dialog	22
3.2. Web App: Select Language Dialog	23
3.3. Same type Web App session detected	24
3.4. Web App type change detected	24
3.5. Web App: Login Dialog - ID Number	25
3.6. Web App: Login Dialog - Local NFC Card	25
3.7. Web App: Login Dialog - Network NFC Card	25
3.8. Web App: Login Dialog - Card Self Association	26
3.9. User Web App: Main View	26
3.10. User Web App: SafePages	28
3.11. User Web App: SafePages - Aggregated	28
3.12. User Web App: Footer Base	29
3.13. User Web App: Hold Print Jobs	31
3.14. User Web App: Hold Copy Job	31
3.15. User Web App: Hold Job Transactions	32
3.16. User Web App: Document Details	32
3.17. User Web App: Landscape Job	33
3.18. User Web App: Rotated Pages	33
3.19. User Web App: SafePage Browser (8 pages)	34
3.20. User Web App: SafePage Browser - Detailed View (4 of 8)	35
3.21. User Web App: PDF - Overview	36
3.22. User Web App: PDF - Document Scope	37
3.23. User Web App: PDF - Description	37
3.24. User Web App: PDF - Security	38
3.25. User Web App: PDF - Passwords	39
3.26. User Web App: PDF - Letterhead	40
3.27. User Web App: PDF - Send	40
3.28. User Web App: Print - Select Printer	41
3.29. User Web App: Print - Selected Printer	42
3.30. User Web App: Print - Job Settings	43
3.31. User Web App: Print - Delete Pages	44
3.32. User Web App: Printer - Direct Print Release	45
3.33. User Web App: Print Delegation Copies	46
3.34. User Web App: Delegated Print - Group Invoicing	47
3.35. User Web App: Delegated Print - User Invoicing (Groups)	48
3.36. User Web App: Delegated Print - Personal Invoicing (Users)	48
3.37. User Web App: Delegated Print - Shared Account Invoicing (Groups)	49
3.38. User Web App: Print - Delegated Print - Shared Account Invoicing (Extra)	50
3.39. User Web App: Print - Select Job Ticket Printer	51
3.40. User Web App: Print - Job Ticket Settings - Print	51
3.41. User Web App: Print - Job Ticket Settings - Copy	52
3.42. User Web App: Print - Job Ticket Settings	52
3.43. User Web App: Print - Job Ticket - Sent	53
3.44. User Web App: Printer - Settings	54
3.45. User Web App: Printer - Settings - Finishings	55
3.46. User Web App: Printer - Settings - Job Ticket	55
3.47. User Web App: Print - Page Scaling (None)	56
3.48. User Web App: Print - Page Scaling (Fit)	56
3.49. User Web App: Print - Preview (4-up, landscape, staple, punch)	56

3.50. User Web App: Letterheads	57
3.51. User Web App: Letterhead - New	58
3.52. User Web App: Letterhead - Detail	59
3.53. User Web App: Delete SafePages	59
3.54. User Web App: Log - Documents	61
3.55. User Web App: Log - Document Transactions	61
3.56. User Web App: Log - Transactions	62
3.57. User Web App: Log - Transactions	64
3.58. User Web App: Sort	65
3.59. User Web App: User Details - Internet Printer Device URI	67
3.60. User Web App: User Details - pagometer	67
3.61. User Web App: User Details - Environmental Impact	67
3.62. User Web App: User Details - Financial	68
3.63. User Web App: Redeem Voucher	68
3.64. User Web App: Transfer Credit	69
3.65. User Web App: Transfer Money from Credit Card	69
3.66. User Web App: Send Bitcoins	70
3.67. Web Print: Upload File	71
3.68. Web Print: Drop Zone - Upload Dialog	72
3.69. Web Print: Drop Zone - Main	72
3.70. User Web App - GDPR Dialog	73
4.1. Admin Web App: Login	74
4.2. Admin Web App: Menu	75
4.3. Admin Web App: Action Pop-up Menu	76
4.4. Admin Web App: Dashboard - Status	77
4.5. Admin Web App: Dashboard - Change System Mode	77
4.6. Admin Web App: Dashboard - Technical Information	78
4.7. Admin Web App: Dashboard - Services	79
4.8. Admin Web App: Dashboard - Pagometer	79
4.9. Admin Web App: Dashboard - Pagometer Trend	80
4.10. Admin Web App: Dashboard - Environmental Impact	80
4.11. Admin Web App: Dashboard - Financial Summary	80
4.12. Admin Web App: Dashboard - Activity	81
4.13. Admin Web App: User - List	82
4.14. Admin Web App: User - Select and Sort	83
4.15. Admin Web App: User Data Portability	84
4.16. Admin Web App: Erased User	84
4.17. Admin Web App: Edit External User - Identity	85
4.18. Admin Web App: Edit User - Roles	85
4.19. Admin Web App: Edit User - Email	86
4.20. Admin Web App: Edit User - Card	87
4.21. Admin Web App: Edit User - UUID	87
4.22. Admin Web App: Edit User - Financial	88
4.23. Admin Web App: Internal User - Password Actions	88
4.24. Admin Web App: Internal User - Password Reset	89
4.25. Admin Web App: Edit User - Delete	89
4.26. Admin Web App: User Group - List	91
4.27. Admin Web App: Group - Select and Sort	92
4.28. Admin Web App: User Groups - Add & Remove	92
4.29. Admin Web App: User Group - Edit - Roles	93
4.30. Admin Web App: User Group - Edit - User Privileges	94
4.31. Admin Web App: User Group - Edit - Admin Privileges	95
4.32. Admin Web App: User Group - Edit - New User Settings	95
4.33. Admin Web App: Account - List	96

4.34. Admin Web App: Account - List - Sub Accounts	96
4.35. Admin Web App: Account - List - Select and Sort	97
4.36. Admin Web App: Account - Edit	98
4.37. Admin Web App: Queue - List	99
4.38. Admin Web App: Queue - Select and Sort	100
4.39. Admin Web App: Queue - Edit	101
4.40. Admin Web App: Proxy Printer - List	102
4.41. Admin Web App: Proxy Printer - Select and Sort	104
4.42. Admin Web App: Proxy Printer - Edit - Identity	105
4.43. Admin Web App: Proxy Printer - Edit - Media Source	107
4.44. Admin Web App: Proxy Printer - Edit - Manual Media Source	108
4.45. Admin Web App: Proxy Printer - Edit - Manual Media Size (Simple)	108
4.46. Admin Web App: Proxy Printer - Edit - Manual Media Size (Advanced)	109
4.47. Admin Web App: Proxy Printer - Rename	110
4.48. Admin Web App: Device - List	111
4.49. Admin Web App: Device - Select and Sort	112
4.50. Admin Web App: Devices - Network Card Reader - Custom User Login	113
4.51. Admin Web App: Devices - Network Card Reader - Proxy Print Authentication	115
4.52. Admin Web App: Devices - Terminal - Custom Proxy Print	117
4.53. Admin Web App: Devices - Terminal - Custom Proxy Print	118
4.54. Admin Web App: Devices - Terminal - Custom User Login	119
4.55. Admin Web App: Devices - Terminal - Custom User Login - Default	119
4.56. Admin Web App: Options - User Source	120
4.57. Admin Web App: Options - User Source - LDAP	121
4.58. Admin Web App: Options - User Source - LDAP	122
4.59. Admin Web App: Options - Internal Users	123
4.60. Admin Web App: Options - User Creation - Import	124
4.61. Admin Web App: Options - User Creation - From Group	124
4.62. Admin Web App: Options - User Creation - Synchronize	125
4.63. Admin Web App: Options - User Creation - On Demand	126
4.64. Admin Web App: Options - User Authentication	127
4.65. Admin Web App: Options - User Authentication - Login Methods	128
4.66. Admin Web App: Options - User Authentication - Username Login	128
4.67. Admin Web App: Options - User Authentication - ID Number Login	129
4.68. Admin Web App: Options - User Authentication - Local NFC Card Login	129
4.69. Admin Web App: Options - User Authentication - YubiKey Login	129
4.70. Admin Web App: Options - User Authentication - Default Login	130
4.71. Admin Web App: Options - Mail - SMTP	130
4.72. Admin Web App: Options - Mail - Messages	131
4.73. Admin Web App: Options - Mail - Test	131
4.74. Admin Web App: Options - PaperCut Integration	132
4.75. Admin Web App: Options - PaperCut Server	132
4.76. Admin Web App: Options - PaperCut Database	132
4.77. Admin Web App: Options - Google Cloud Print - Status	133
4.78. Admin Web App: Options - Google Cloud Print - OAuth	135
4.79. Admin Web App: Options - Google Cloud Print - Notifications	137
4.80. Admin Web App: Options - Mail Print (IMAP)	138
4.81. Admin Web App: Options - Mail Print (Attachments)	139
4.82. Admin Web App: Options - Web Print	140
4.83. Admin Web App: Options - Internet Print	140
4.84. Admin Web App: Options - Proxy Print General	141
4.85. Admin Web App: Options - Proxy Print Modes	142
4.86. Admin Web App: Options - Proxy Print Delegation	143
4.87. Admin Web App: Options - Eco Print	145

4.88. Admin Web App: Options - Financial - Currency	145
4.89. Admin Web App: Options - Financial - General	146
4.90. Admin Web App: Options - Financial - POS	146
4.91. Admin Web App: Options - Financial - Vouchers	147
4.92. Admin Web App: Options - Financial - Transfer funds	147
4.93. Admin Web App: Options - Backups	148
4.94. Admin Web App: Options - Automatic Backups	148
4.95. Admin Web App: Options - Advanced - User Client	149
4.96. Admin Web App: Options - Advanced - Reset Admin Password	150
4.97. Admin Web App: Options - Advanced - JMX Agent	150
4.98. Add JMX Connection with Java VisualVM	151
4.99. Connecting to Remote Process with JConsole	151
4.100. Admin Web App: Options - Advanced - Locale	152
4.101. Admin Web App: Options - Default Paper Size	152
4.102. Admin Web App: Options - Default Paper Size	153
4.103. Admin Web App: Options - Converters	153
4.104. Admin Web App: Options - Advanced - Proxy Printing	154
4.105. Admin Web App: Options - Advanced - Pagometers	155
4.106. Admin Web App: Configuration Editor - List	156
4.107. Admin Web App: Configuration Property - Edit	156
4.108. Admin Web App: Documents - List	157
4.109. Admin Web App: Documents - Select and Sort - All	159
4.110. Admin Web App: Documents - Select and Sort - In	159
4.111. Admin Web App: Documents - Select and Sort - Out	160
4.112. Admin Web App: Documents - Select and Sort - PDF	160
4.113. Admin Web App: Documents - Select and Sort - Print	161
4.114. Admin Web App: Documents - Select and Sort - Ticket	161
4.115. Admin Web App: Log - List	162
4.116. Admin Web App: Log - Select and Sort	162
4.117. Admin Web App: Log - Select Date	163
4.118. Admin Web App: About	164
4.119. Admin Web App: About - Version	164
4.120. Admin Web App: About - License	165
4.121. Admin Web App: About - Community	165
4.122. Admin Web App: About - Import Member Card	166
4.123. Admin Web App: About - Support	167
4.124. Admin Web App: About - Host Packages	168
4.125. Admin Web App: Voucher List	169
4.126. Admin Web App: Vouchers - Select and Sort	170
4.127. Admin Web App: Voucher Actions	171
4.128. Admin Web App: Create Vouchers	172
5.1. Job Tickets: Open Tickets - Select and Sort	174
5.2. Job Tickets: Open Tickets - List	175
5.3. Job Tickets: Cancel All	176
5.4. Job Tickets: Print All	176
5.5. Job Tickets: Edit Ticket	177
5.6. Job Tickets: Print Ticket	178
5.7. Job Tickets: Print Pending	178
5.8. Job Tickets: Print Processing	179
5.9. Job Tickets: Print Canceled	179
5.10. Job Tickets: Print Completed	179
5.11. Job Tickets: Settle Ticket	180
5.12. Job Tickets: Closed Ticket List	180
5.13. Job Tickets: Refund	181

5.14. Job Tickets: Refund Transactions	181
6.1. Point-of-Sale: Deposit Start	183
6.2. Point-of-Sale: Deposit Completed	184
6.3. Point-of-Sale: Receipts	185
10.1. SavaPage Printer on Ubuntu: Choose Driver	196
10.2. SavaPage Printer on Ubuntu: Printer Properties	196
10.3. SavaPage Printer on macOS: Add Printer	197
10.4. SavaPage Printer on macOS: Select PPD	197
10.5. SavaPage Printer on macOS: Print & Fax	197
10.6. SavaPage Local Printer on Windows	198
10.7. SavaPage Shared Local Printer on Windows	198
10.8. SavaPage Network Printer on Windows	199
10.9. iPad App Sharing Options	201
10.10. SavaPage Printer Options on iPad	202
10.11. Select SavaPage Printer on iPad	202
11.1. SavaPage in a Single Sign-On Domain	209
11.2. IP Based Authentication for Abstract User	210
11.3. IP Based Authentication for Unauthenticated User	211
11.4. IP Based Authentication in Peer-to-Peer Network	212
16.1. User Web App: Custom CSS - Sample #1	228
16.2. User Web App: Custom CSS - Sample #2	229
C.1. JSON-RPC : Basic Error	267
C.2. JSON-RPC : systemStatus (request)	268
C.3. JSON-RPC : systemStatus (response)	268
C.4. JSON-RPC : authUserSource (request)	269
C.5. JSON-RPC : authUserSource (response)	269

List of Tables

1. Typographical conventions	xix
3.1. Print Job Settings Configuration Properties	45
3.2. Delegated Print Configuration Properties	50
3.3. Job Ticket Print Configuration Properties	53
3.4. Basic IPP Printer Attributes	57
3.5. Document Type Select: Configuration Property	60
3.6. Configuration Properties for Web Print Document Types	71
3.7. Configuration Properties for User Web App GDPR	73
4.1. User Roles	85
4.2. YubiKey Configuration Properties	130
4.3. LibreOffice Configuration Properties	153
5.1. Job Ticket Print Configuration Properties	181
7.1. User Client Access Configuration Properties	186
7.2. User Client Options Configuration Properties	187
9.1. Secured Application Areas	191
9.2. Server Properties for Alternative TCP/IP Settings	193
9.3. Server Properties for Alternative File Locations	193
9.4. Server Properties: Miscellaneous Settings	193
13.1. Default Web Session Timeout Values	217
13.2. Web Session Timeout Configuration Properties	217
16.1. Stock Image Identifiers	231
16.2. Placeholder: Stationary	233
16.3. Placeholder: Application	233
16.4. Placeholder: Ticket	234
16.5. Placeholder: User	234
16.6. Email Stationary Types	234
16.7. Placeholder Objects: JobTicketCanceled	234
16.8. Placeholder Objects: JobTicketCompleted	235
16.9. Configuration Properties for Custom Template Locations	235
A.1. Delegated Print - (Non) Secure & Job Ticket Scenarios	247
A.2. Delegated Print - Job Ticket - PaperCut Scenario	248
A.3. Delegated Print - PaperCut Scenario	249
C.1. XML-RPC method: onetime-auth.createToken	266
C.2. JSON-RPC Configuration Properties	267
D.1. Database size increase metrics per document flow.	276
E.1. SavaPage URL Cheat Sheet	278
F.1. SavaPage File Locations	280
G.1. Standard Printable File Types	282
G.2. Advanced Printable File Types	283
J.1. LDAP Configuration Properties	288
J.2. OpenLDAP Default Settings	289
J.3. Apple Open Directory Default Settings	289
J.4. Novell eDirectory Default Settings	290
J.5. Microsoft Active Directory Default Settings	290
J.6. Microsoft Active Directory Custom Settings	291
L.1. Internal IPP Attribute: org.savapage-finishings-staple	305
L.2. Internal IPP Attribute: org.savapage-finishings-punch	306
L.3. Internal IPP Attribute: org.savapage-finishings-fold	306
L.4. Internal IPP Attribute: org.savapage-finishings-booklet	307
L.5. Internal IPP Attribute: org.savapage-finishings-jog-offset	307
L.6. Internal IPP Attribute: org.savapage-finishings-ext	308

L.7. Internal IPP Attribute: org.savapage-cover-type	308
L.8. Internal IPP Attribute: org.savapage-job-sheets	308
L.9. Internal IPP Marker Attribute: org.savapage.int-page-rotate180	309
M.1. Web API Callback Configuration Property	311
M.2. Payment Gateway Configuration Property	311
O.1. Cron Trigger Format - simplified	320

Preface



1. About this Manual

The SavaPage User Manual covers the setup, management and configuration of *SavaPage Open Print Portal*. Please take a few moments prior to installing the application to read the key sections of this manual. The latest version of this manual in HTML, PDF and EPUB format are available from the SavaPage [website](https://www.savapage.org/)¹.

2. Expectations and Prerequisites

SavaPage is a network based application. Experience with basic network concepts, such as server administration and network connectivity is expected. Prior to installing or evaluating SavaPage you should be familiar with the concepts of:

- Basic GNU/Linux Systems Administration.
- Client/Server computing.
- Internet Printing Protocol (IPP).
- IP Printing (JetDirect/RAW)².
- Common Unix Printing System (CUPS).
- Lightweight Directory Access Protocol (LDAP).

3. Conventions used in this Document

3.1. Typographical Conventions

This is a list with examples of the typographical conventions used in this manual.

Convention	Example
Executable program with options.	Enter <code>ls --reverse</code> to get a directory listing in reverse order.
A character or string indicating the start of an input field.	Enter you name after the <code>Username:</code> prompt.
User input.	John entered john as his login name.
A button.	Press the <code>Cancel</code> button.
A text prompt.	Enter your full name after the Name prompt.
Content that may or must be replaced by the user.	Please enter filename to save the content to.
Filename.	Please open the file <code>server.properties</code> in your favorite editor.
Directory.	<code>/opt/savapage</code>

¹ <https://www.savapage.org/>

² JetDirect is the most common Socket API, and a *de facto* standard, introduced by Hewlett-Packard. It allows a TCP/IP connection via port 9100 to a printer attached to a Local Area Network similar to a connection to its serial or parallel port. Windows supports this protocol as *Standard TCP/IP port* monitor for print server attached print devices.

Convention	Example
A question-and-answer set.	Q: To be, or not to be? A: That is the question.
Key on a keyboard	Press F1 for help.
A combination of input actions.	Press CTRL+C to abort the program.
A selection or series of selections from a menu.	Select Print → Settings to open the dialog.
An inline code fragment.	Text in this format is used to show code examples, the contents of files, and console output, as well as the names of variables, commands, and other code excerpts in the text.
Code (listing).	<pre> Line A <i>Inline annotation on A</i> Line B ❶ <i>Inline annotation on B</i> Line C ❷ </pre> <p>❶ Comment for Line B. ❷ Comment for Line C.</p>
Link (external).	A link to an URI is formatted like this: https://wiki.savapage.org and mailto:support@savapage.org Or as an alternative: Visit our website ³ or write an email ⁴
Link (internal).	See Chapter 2, Server Installation [11] of this manual.
Name of a variable.	In Perl, @ARGV contains the command line parameters used when the script was run.
Inline text that is some literal value.	When debug is activated more detailed logging is produced.

Table 1. Typographical conventions

3.2. Notes

You should pay special attention to notes set apart from the text with the following icons:



Important

Important notes are marked like this.



Note

Notes provide extra background information.



Tip

Tips provide useful advice to make your life easier.



Caution

Indicate situations where you have to be careful what you are doing.

³ <https://wiki.savapage.org>

⁴ <mailto:support@savapage.org>



Warning

Where extreme care has to be taken.

4. Notice

While every effort has been taken to ensure the accuracy and usefulness of this manual, we cannot be held responsible for occasional inaccuracy or typographical errors. If you find an inaccuracy or error, please let us know.

Information in this document is subject to change without notice. The names of companies, products, people, characters, and data mentioned in the examples are fictitious and are in no way intended to represent any real individual, company, product, or event, unless otherwise noted.

5. Your Feedback

This manual isn't “done”. In fact, this manual will probably never be completely “done”. The subject it covers is expected to change and expand over time, and we consider this work a reflection of our ongoing conversation with the [SavaPage Community](https://www.savapage.org/)⁵. Publication of this manual highlights the openness of the product, and that you, as a user, can play a pivotal role in helping to maintain and improve the product. If you see anything in this manual that can be improved: spelling, examples, explanations, then you should tell us, and send us an [email](mailto:support@savapage.org)⁶. Also, if you have ideas about improving the product in general, please let us know. All feedback will be rewarded with a gracious response.

⁵ <https://www.savapage.org/>

⁶ <mailto:support@savapage.org>

Chapter 1. Introduction



1.1. What is SavaPage?

SavaPage is an Open Print Portal that uses Open Standards and Commodity Hardware for Secure Pull-Printing, Pay-Per-Print, Delegated Print, Job Ticketing, Auditing and PDF Creation.

SavaPage is implemented as **Print Server** on GNU/Linux. Any workstation or device can print to SavaPage.

- Devices supporting Internet Printing Protocol (IPP) or IP Printing (JetDirect), like Windows, Mac and GNU/Linux workstations, can print to SavaPage.
- macOS and iOS devices can use AirPrint® ¹.
- Android and Chrome OS devices can use Google Cloud Print.
- Any device can use *Web Print* and *Mail Print* to print.

Printed pages are shown in the SavaPage **Web Application**. Print jobs are accumulated in a single personal preview, where they can be manipulated and pruned, before storing or routing them as PDF document.

In the Web App, documents are routed to a “real” printer, optionally via intermediate Job Tickets, with Common Printing Dialogs of server-side installed printers (*proxy printers*). This makes SavaPage the Central Print Portal where documents to be printed are acquired and routed.

Pay-per-Print functions charge printing costs to individuals, groups, or shared accounts. With **Delegated Print** authorized users can print on behalf of other users.

SavaPage Web App is optimized for desktop as well as mobile browsers. This opens up many useful scenario's. For instance, a user can walk up to a printer and send a print job on the spot, by pushing a button on his smartphone. With a special Web App, administrators on the go can easily monitor the system on their tablet.

SavaPage turns printing into a unified experience, abstracted from platform specifics. It is the logical stopover where users are guided through sustainable print scenario's that help reduce overall printing costs.

1.1.1. Open Source Software

SavaPage is *OSI Certified Open Source Software* ², licensed under the **GNU Affero General Public License (AGPL)** ³ as published by the **Free Software Foundation** ⁴, either version 3 of the License, or (at your option) any later version.

1.1.2. Benefits

The key benefits of SavaPage are:

- *Less administration.* SavaPage is the one printer you need to print to any printer in your organization.
- *Zero install.* A generic PostScript driver and web browser is all you need to print from Windows, Mac and GNU/Linux and preview the result.

¹ *AirPrint* is a registered trademark of Apple Inc.

² OSI Certified is a certification mark of the **Open Source Initiative** [<https://opensource.org/>].

³ <https://www.gnu.org/licenses/agpl.html>

⁴ <https://www.fsf.org/>

- *Multi-platform.* Corporate printers are sandboxed in the Web App Preview and thus available on all mobile and desktop platforms for pass-through (proxy) printing.
- *Easy follow-me printing.* Several hold-release scenarios, optionally with NFC cards, are supported. Users can even use their own mobile device as print release terminal.
- *Mobile printing.* Google Cloud Print, iOS AirPrint®, Web Print and Mail Print is supported out of the box.
- *Think before you print.* SavaPage Web App shows a print preview that makes you think twice. Do you really need to print all these pages?
- *Eco-friendly.* Create environmental awareness by drawing end-user attention to the cost of printing, and save precious paper, trees and money along the way.
- *Reduction of printing costs.* Remove unnecessary pages and graphics. Save as PDF, or route to a "real" printer with n-up, gray-scale and duplex to reduce printing costs.
- *No pre-printed paper needed.* Eliminate the cost of pre-printed paper. Create virtual letterheads and apply them to any print job.
- *Advanced Print Services* with Job Ticketing and Delegated Print.
- *Open Source Software* and *Open Standards* above Proprietary Software.
- *Commodity Hardware* above expensive Proprietary Devices.
- *Peer-To-Peer Cooperation* above Centralized Corporation. The [SavaPage Community](#) is there to help.

1.1.3. Key Features

The key features of SavaPage are:

- One SavaPage Printer Driver
 - Generic [PostScript Driver](#) print from Windows, macOS and GNU/Linux.
 - Secure [Internet Print](#).
- Mobile Print
 - [Google Cloud Print](#) from Android and Chrome OS.
 - [AirPrint®](#) from iOS (iPad, iPhone).
- Driverless Printing
 - [Web Print](#) and [Mail Print](#) to print from any device.
- [Follow-me Printing](#)
 - [Release Terminals](#)
 - [NFC Authentication](#)
- Web Apps for Desktops *and* Mobile Devices
 - Easy authentication
 - [Username/Password](#), [ID/PIN](#) and [NFC Card](#) authentication.
 - [LDAP](#) (Active Directory) Integration.
 - [Raspberry Pi](#) Network Card Reader.
 - [User Web App](#)
 - Real-time print preview with [Browse](#), [Sort](#) and [Delete](#) options.
 - Server-side [Proxy Printing](#) (no local drivers needed).
 - [PDF Download](#) or [Email](#) of accumulated print jobs.
 - Multi-page [Letterheads](#).
 - Option to [remove graphics](#) from PDF and proxy print output.
 - Innovative [Eco Print](#) to reduce ink and toner cost.
 - [Delegated Print](#) for delegates to proxy print for other users and groups.

- **Job Ticket Print** for voluminous proxy print jobs.
- **Admin Web App**
 - Comprehensive Web App to configure the SavaPage environment.
- **Multi-language** support.
- **Customizable Web Interface**.
- **SSL Encryption**.
- **SavaPage Financial**
 - **Pay-per-Print**
 - **Vouchers**
 - **Point-of-Sale Web App**
 - **Online Payments** (credit cards, bank accounts, Bitcoin).
- Tooling and Tuning
 - **Command-Line Interface** to server methods.
 - **Web Services API**.
 - **Third party Database support**.
- Third Party Integration
 - Microsoft Active Directory
 - Single-Sign-On (Moodle, OAuth)
 - Third Party Print Management Systems (**PaperCut**).
- Comprehensive Documentation
 - User Manual in PDF, EPUB and HTML format.
- **GDPR Compliant**.

1.2. System Requirements

1.2.1. Server

SavaPage Open Print Portal can be installed on any modern GNU/Linux system that supports **systemd** service manager like distributions based on Debian and Red Hat Enterprise Linux (RHEL), and openSUSE. Debian based distributions that use **SysV** init scripts are also supported.



Note

Throughout this manual GNU/Linux command and file examples are given for Debian based systems. Commands and files might differ for other distributions, but not in function. For example, the Debian **apt-get** command has a RHEL **yum** and openSUSE **yast** equivalent. It is trusted that system administrators can translate the examples to their own environment. If applicable, functional differences between distributions will be explained.

1.2.1.1. Java

SavaPage is a Java program and needs JDK 8 to be installed. Check the installation by executing both the **java** and **javac** commands: they should echo usage information.

On Debian based systems you can install the package with one of these commands ⁵:

⁵ On Debian "Jessie" you need the **Debian Backports repository** [<https://backports.debian.org/>] to install OpenJDK 8.

```
sudo apt-get install openjdk-8-jdk
```

1.2.1.2. CUPS

SavaPage uses local CUPS printer queues for [Proxy Printing](#). CUPS 1.4 or higher must be installed ⁶.

On Debian based systems you can install CUPS with these commands:

```
sudo apt-get install cups      ❶
sudo apt-get install cups-bsd  ❷
```

- ❶ This package provides the parts of CUPS which are needed for using printer drivers.
- ❷ This package provides the BSD commands for interacting with CUPS, like **lpr**.



Important

SavaPage will automatically add any local CUPS printer as proxy printer. So, for proxy printing to work, first add each proxy printer as CUPS printer.



Tip

Modern GNU/Linux distributions have everything prepared for using most printers. For USB printers it is often enough to simply plug them in. For network printers you simply start the distribution's printer setup tool out of the system administration menu or out of a system administration application, click on a button for adding a new printer and then follow the screen instructions. If this does not work, usually there is no suitable driver installed on your system. Verify in the [OpenPrinting database](#)⁷ whether your printer is supposed to work and whether there is a driver or PPD file available. See the [OpenPrinting CUPS Quick Start](#)⁸ for more details.

1.2.1.3. Database

SavaPage is packaged with an internal database that offers you the opportunity to evaluate the product on a small scale right away. However, when promoting SavaPage to a production environment with multiple users, we strongly advise you to use [PostgreSQL](#)⁹ as external database server. See [Chapter 17, Using an External Database](#) [236].



Warning

Using the internal database in situations with concurrent use will inevitably lead to locking, deadlock and out-of-memory errors, which can make the system totally unresponsive.

1.2.1.4. Poppler

SavaPage needs the PDF utilities based on Poppler to function properly.

[Poppler](#)¹⁰ is a PDF rendering library based on Xpdf PDF viewer. The command line utilities are used to get information of PDF documents, convert them to other formats, or manipulate them. SavaPage uses **pdftoppm** to convert PDF pages to images.

⁶ CUPS 1.4 or higher provides the Job Template attribute “fit-to-page” that is used by SavaPage proxy printing to scale documents to fit the size of selected media. See <https://www.cups.org/doc/options.html>. Also see [Section K.1.1.5, “print-scaling”](#) [293].

⁷ <https://www.openprinting.org/printers>

⁸ <https://www.linuxfoundation.org/collaborate/workgroups/openprinting/database/cupsdocumentation>

⁹ <https://www.postgresql.org/>

¹⁰ <https://poppler.freedesktop.org/>

Check if the package is installed by entering the following command:

```
pdftoppm -v
```

On Debian based systems you can install the package with this command:

```
sudo apt-get install poppler-utils
```

1.2.1.5. ImageMagick

SavaPage needs the **convert** command of the ImageMagick software suite to manipulate images.

ImageMagick is a software suite to create, edit, compose and convert bitmap images in a variety of formats (over 100) .

Check by entering the following command:

```
convert --version
```

On Debian based systems you can install the package with this command:

```
sudo apt-get install imagemagick
```

1.2.1.6. Avahi

Avahi is needed if you want to print to SavaPage from iOS devices (iPad, iPod, iPhone). See [Section 10.3, “Printing from iOS” \[200\]](#).

[Avahi](#)¹¹ is a system which facilitates service discovery on a local network via the mDNS/DNS-SD ¹² protocol suite. Any modern GNU/Linux system has Avahi installed, but to be sure you can check by entering the following command:

```
avahi-browse --version
```

On Debian based systems you can install the package with this command:

```
apt-get install avahi-daemon avahi-discover libnss-mdns
```

1.2.1.7. Hardware

The SavaPage server process requires a minimum of 2 CPU cores, 2GB of RAM and 1 GB of free disk space.



Note

Depending on the expected print quota you should reserve extra disk-space for each SavaPage user. See [Appendix D, Capacity Planning \[276\]](#).

1.2.2. Clients

On *desktop and mobile* clients you need:

- An HTML5 compatible browser.

For printing to SavaPage on *GNU/Linux and Mac* clients you need:

¹¹ <http://avahi.org/>

¹² mDNS/DNS-SD enables you to plug your laptop or computer into a network and instantly be able to view other people who you can chat with, find printers to print to or find files being shared. Compatible technology is found in Apple macOS (branded [Bonjour](#) [<https://www.apple.com/support/bonjour/>] and sometimes Zeroconf).

- IPP printer support.

For printing to SavaPage from *Windows* you can choose between:

- A Local Printer on a *Standard TCP/IP Port*, when you want to share a SavaPage printer (e.g. on a Windows Print Server).
- A Network Printer using Internet Printing Protocol (IPP).

To AirPrint to SavaPage on devices like *iPad*, *iPhone* and *iPod Touch* you need:

- iOS 4.2 or higher.

To AirPrint to SavaPage from macOS you need:

- macOS 10.7 Lion or higher.



Note

The SavaPage WebApps use **jQuery Mobile**¹³ which offers broad support for the vast majority of all modern desktop, smartphone, tablet, and e-reader platforms.

1.3. How does SavaPage work?

To explain how SavaPage works we first introduce the key concepts for the usage scenarios. After that we will describe a typical work flow, and end with a high-level overview of the application's architecture.

1.3.1. Key Concepts

Each concept is described in an abstract definition with its SavaPage implementation.

1.3.1.1. Print Server

A print server is a system responsible for hosting print queues and sharing printer resources to client workstations. Client users submit print jobs to a print server rather than directly to the printer itself. A print server may be a dedicated server. However, on many networks this server may also perform other tasks such as file serving.

SavaPage is a regular Print Server in the technical sense, but is special in the sense that it shares multiple print queues of the one SavaPage virtual printer. The GNU/Linux host where SavaPage is deployed on may offer file services on its own account.

1.3.1.2. Print Queue

A print queue is first-in-first-out queue holding all jobs pending on a given printer.

*SavaPage virtual queues redirect print jobs to the originating user's personal queue called **SafePages**. The SavaPage Web App is the viewport on these SafePages.*

1.3.1.3. User ID/Username

In a multi-user environment, users login to a network or computer using a username and password. Often these are managed by services like Active Directory or LDAP. The username is known as the user's identity.

SavaPage uses this identity for authentication and auditing purposes.

¹³ <https://jquerymobile.com/>



Note

User authentication is a topic of its own. Please see [Chapter 11, *Authenticated Printing* \[206\]](#) for more elaboration on the *User* concept.

1.3.1.4. Client/Server Model

Client software is a small application that runs on each workstation and communicates with a central server. The printing process on most networks works according to a client/server model with clients (workstations) submitting jobs to a server.

SavaPage utilizes the client/server model with standard components on the workstation, i.e. an IPP or JetDirect printing client and a Web browser.

1.3.1.5. Application Server

An application server is a server program responsible for centrally processing “business logic” and providing services to end-users.

SavaPage is an application server since it provides “business logic” for showing, editing and routing printed documents.

1.3.1.6. Information Provider

A provider is a software component or program responsible for providing information to an Application Server.

SavaPage uses an integrated IPP and JetDirect Server to capture Driver Print jobs from client workstations and devices. It communicates with IMAP to capture Mail Print jobs and uses HTTP upload to capture Web Print jobs. The generic information provider for capturing print jobs is called the “Print Provider”. Other important providers are “User Directory Provider”, “Authentication Provider” and “CUPS Information Provider”.

1.3.1.7. Web Application Interface

A Web Application, or Web App for short, is a software program that interacts with end-users via a web browser. A Web App gives flexibility because it allows access from any location on the network and avoid the need for installation of separate software.

SavaPage provides a web-based interface for end-users and system administrators. Since it is optimized for desktops and mobile devices an even greater flexibility is achieved.

1.3.1.8. SafePages

SafePages is the SavaPage term for the personal user space with accumulated jobs from SavaPage printer queues. See [Section 1.3.1.2, “Print Queue” \[6\]](#).

1.3.1.9. Proxy Printer

Proxy Printer, or *ProxyPrinter*, is the SavaPage term for a printer that is available in the SavaPage Web App for printing selected SafePages.



Important

It is important to understand that using a Proxy Printer does *not* require its printer driver on the client workstation. Proxy Printer queues are CUPS queues located on the GNU/Linux SavaPage host and are *not* shared

on the local network, hence *not* visible for client workstations. Proxy Printer queues can only be selected and used in the SavaPage Web App sandbox for pass-through printing.

1.3.2. The SavaPage Work Flow

To illustrate what SavaPage is about and how it works we'll start with a simple use case.



Note

Advanced user scenario's are described in [Appendix A, Proxy Print Scenarios \[247\]](#).

1.3.2.1. End-user perspective

1. John opens a web browser, clicks on the SavaPage bookmark and logs into SavaPage with his regular Active Directory credentials.
2. John prints a document from his favourite editor to his *SavaPage Network Printer*.
3. John sees the printed pages appear as thumbnails in his web browser.
4. John browses through the thumbnails and zooms in on page 15 and 16 to see more detail.
5. Things look good, apart from two void pages at the end. So, John deletes these pages using the Delete dialog.
6. After selecting the company letterhead as standard background, John selects the Brand-X Multi-functional Proxy Printer located down the hall, checks the settings (duplex and grayscale), and presses the Print button.
7. Since John also wants to save a PDF document of the result, he sets the PDF properties (title, author, subject, keywords, encryption) and presses the Download button.



Note

John could also have opened a web browser on his smartphone and do exactly the same things.

1.3.2.2. Technical perspective

This is what happens behind the scenes.

1. When John prints to SavaPage from his editor, his workstation transfers the print job to the [SavaPage Print Server](#).
2. The SavaPage [Print Provider](#) handles the print job, analyzes the information and retrieves:
 - a. The identity of the *user* who printed the document.
 - b. The identity of the *queue* the job was printed to.
3. The Print Provider submits the information about the job to the [Application Server](#) to process the business logic.
4. The Application Server approves the print request, transfers the job to the user's [SafePages](#), and signals John's browser session that a new job has arrived.
5. The [Web Application](#) in John's browser picks up the signal, handles the information and displays the newly printed pages.
6. The Web Application transfers each of John's editing actions (delete, letterhead) to the Application Server where the *state* of the SafePages is saved.
7. When John selects the Brand-X Multi-functional [Proxy Printer](#), the Web App asks the Application Server for the printer options, so it can display the Printer Settings dialog for this specific printer.

8. When John presses the Print button, the print action plus the selected printer options are passed to the Application Server. The server composes the print job (applying editing actions and selected letterhead) and sends the result to the Proxy Printer, using the printer options John selected.
9. John's download request is fulfilled by the Application Server with a PDF document holding the edited SafePages, including the letterhead, and the chosen PDF settings.

1.3.3. Architecture Overview

Figure 1.1, “SavaPage High-Level Architecture” [9] shows a high level view of the components and communication involved.

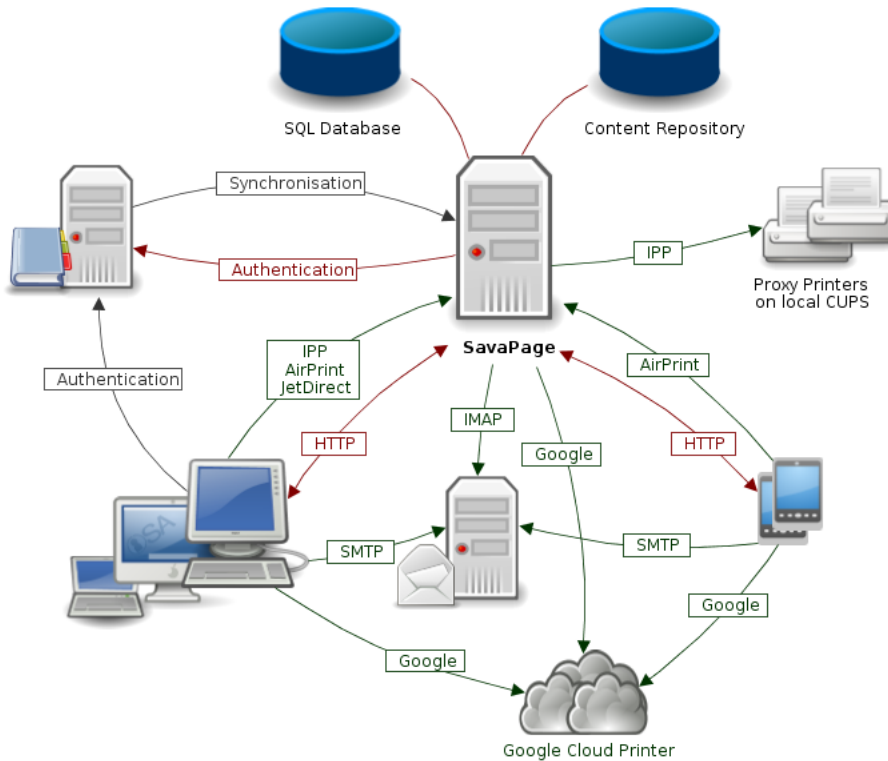


Figure 1.1. SavaPage High-Level Architecture

- The SavaPage **Print Server** synchronizes users from the LDAP/NIS source to its own SQL Database.
- The Client **Web Application** on desktops, laptops and mobile devices communicates with the **Application Server** using HTTP.
- Desktop and laptops users can be forced by their OS to login and be authenticated at the LDAP/NIS source.
- SavaPage Web App users on desktops, laptops and mobile devices are authenticated by the SavaPage Print Server at the LDAP/NIS source.
- Desktop and laptop users print to SavaPage with the SavaPage **printer driver** using IPP or JetDirect protocol.
- macOS and iOS users can print to SavaPage with AirPrint®.
- Every user can use SMTP to **Mail Print** to SavaPage.
- SavaPage uses IMAP to monitor the Web Print Inbox.
- Every user can use HTTP to **Web Print** to SavaPage.
- Every user can print to the **Google Cloud Ready** SavaPage Printer.
- The Content Repository holds letterhead documents.

- A print command in the Web App to a **Proxy Printer** is executed by SavaPage with an IPP operation to local CUPS.

Chapter 2. Server Installation



This chapter covers the *initial* installation and configuration of SavaPage in your network environment.

- If you are installing a new version over an existing installation please consult [Appendix H, *Upgrading from a Previous Version* \[285\]](#).
- If this installation is part of a migration from an old server please consult [Appendix I, *Migrating to a New Server* \[286\]](#) before going on.

Initial installation takes only a few minutes on a prepared server. This guide will walk you through installation and configuration step-by-step. The process is summarized below:

1. System requirements check.
2. Downloading and installing SavaPage.
3. Completing the configuration.
4. Testing the software.



Tip

If you would like to know the technical details behind the SavaPage installer, take a look at [Section 9.1, “The Installation Process” \[190\]](#).



Important

By installing the program, you are accepting and agreeing to the terms of the GNU Affero General Public License (AGPL). Please review [Appendix P, *GNU Affero General Public License \(AGPL\)* \[321\]](#) before continuing.

2.1. Step 1 - System Requirements

Before proceeding with the installation you should take a few moments to verify system requirements. Is the operating system version supported and are patches up-to-date? Take a few minutes to verify the system is current and supported (see [Section 1.2, “System Requirements” \[3\]](#)).

The SavaPage installation program needs the commands **which**, **strings**, **gunzip** and **perl**. So, make sure the **binutils**, **debiantutils** (for Debian based systems), **perl** and **gzip** packages are installed.

2.2. Step 2 - Create System Account

SavaPage runs and installs under a system user account called `savapage`. This account is fixed, you cannot choose another name. You are free though to pick a location for the application. However, GNU/Linux Filesystem Hierarchy Standard (FHS) dictates that the application is installed in the `/opt/savapage` directory.

Create the system user account at the command prompt by entering:

```
sudo useradd -r savapage
```

The syntax for `useradd` may differ slightly on different versions of GNU/Linux. It may also be called `adduser`.

Next, create the install directory and set the ownership to the `savapage` account:

```
sudo mkdir -p /opt/savapage
```

... and set the ownership to the `savapage` account. For Debian and RHEL based systems:

```
sudo chown savapage:savapage /opt/savapage
```

... and for openSUSE:

```
sudo chown savapage:users /opt/savapage
```

For convenience, set the login shell and login directory:

```
sudo usermod -s /bin/bash savapage
sudo usermod -d /opt/savapage savapage
```

Some GNU/Linux distributions impose strict resource usage limits on user accounts (`ulimit`). The `savapage` account is a dedicated account used for hosting the SavaPage application and hence should be granted sufficient resource limits such as the ability to open many files. Please consult [Section 18.2, “Linux User Limits” \[241\]](#) on how to change these limits.

2.3. Step 3 - Configure CUPS and Samba

Make sure to *not* publish shared printers in CUPS *and* Samba. Publishing shared printers creates a loophole by which users can access a printer directly from their workstation and print outside the control of SavaPage.

For Samba, just edit the `/etc/samba/smb.conf` file and disable the `[printers]` share definition.

In CUPS, do *not* enable the “Publish shared printers connected to this system” option as offered in the Print Server Settings dialog. When no such dialog is available you can make the adaption in the CUPS Administrator Web interface (“Share printers connected to this system”), or manually in the `cupsd.conf`¹ file.



Important

Before editing `cupsd.conf` first stop CUPS by entering this command:

```
sudo service cups stop
```

When editing `cupsd.conf` change this content snippet, that publishes local printers and allows access from all machine on the local network...

```
# Allow remote access
Port 631

# Share local printers on the local network.
Browsing On

<Location />
  # Allow access to the server...
  Order allow,deny
  Allow @LOCAL
```

¹ On Debian, RHEL and openSUSE systems `cupsd.conf` is located in the `/etc/cups/` directory.

```
</Location>
```

... to this snippet that restricts CUPS access from localhost only ...

```
# Only listen for connections from the local machine.
Listen localhost:631

# Disable printer sharing.
Browsing Off

<Location />
    Order allow,deny
</Location>
```

... and leave all other content as it is.



Important

Each individual proxy candidate CUPS printer must be *shared* locally so the savapage system account can access it. Enabling the shared option can be done in a printer GUI dialog, in the CUPS Administrator Web interface, or directly in the `printers.conf`² file by setting the `Shared Yes` option for a printer.

2.3.1. CUPS Remote Printer Browsing

To prevent remote printers that DNSSD broadcast themselves, to be discovered by CUPS and synchronized into SavaPage, edit `cups-browsed.conf`³ file, and change it as follows:

```
BrowseRemoteProtocols none
```

2.3.2. CUPS Job History

An active SavaPage server captures print job statuses real-time, but when the server is restarted it needs CUPS job history to catch up with the latest statuses. To avoid lost job statuses, CUPS must be told to “Preserve job history”.

You can set the Job History option in the Print Server Settings dialog (“Preserve job history but not files”, or optionally “Preserve job history (allow reprinting)”), in the CUPS Administrator Web Interface (Advanced settings, “Retain Metadata : Yes”, and optionally “Retain Documents : Yes”), or manually by changing the `cupsd.conf` file as follows:

```
MaxJobs 0           ❶
PreserveJobHistory Yes  ❷
PreserveJobFiles No  ❸
```

- ❶ `MaxJobs` specifies the maximum number of simultaneous jobs that are allowed. Set to 0 to allow an unlimited number of jobs.
- ❷ `PreserveJobHistory` specifies whether *metadata* is preserved after a job is printed. A value of `Yes` will preserve history, a value of `No` will not. If a numeric value is specified, history is preserved for the indicated number of seconds after printing. Set to **Yes**.
- ❸ `PreserveJobFiles` specifies whether *job files* (documents) are preserved after printing. A value of `Yes` will preserve files, a value of `No` will not. If a numeric value is specified, files are preserved for the indicated number of seconds after printing. Set this option as you wish, but remember that (spool) files can get big. If you run SavaPage on a host system with limited storage (for instance on a virtual machine) you better set this value to **No**.

² On Debian, RHEL and openSUSE systems `printers.conf` is located in the `/etc/cups/` directory.

³ On Debian, RHEL and openSUSE systems `cups-browsed.conf` is located in the `/etc/cups/` directory.

2.3.3. CUPS Job Privacy

CUPS makes “job-name”, “job-originating-host-name” and “job-originating-user-name” private by default. This means that personal data are anonymized in the CUPS Web interface, as shown below.

▼ ID ▼	Name	User	Size	Pages	State
Xerox-D125-8586	Unknown	Withheld	1k	1	completed at Mon 30 May 2016 10:47:27 AM CEST

Figure 2.1. CUPS Job Privacy

You can restore the default by changing the `JobPrivateValues` directive in the `cupsd.conf` file as follows:

```
<Policy default>
  JobPrivateValues default
```

2.3.4. CUPS Web Interface

If you want to use the CUPS Web Interface for administration from all machines on the local network you should adapt `cupsd.conf` as follows:

```
# Allow remote access
Port 631

# Disable printer sharing.
Browsing Off

WebInterface Yes

<Location />
  # Allow shared printing...
  Order allow,deny
  Allow @LOCAL
</Location>

<Location /admin>
  Order allow,deny
  Allow @LOCAL
</Location>

<Location /admin/conf>
  AuthType Default
  Require user @SYSTEM
  Order allow,deny
  Allow @LOCAL
</Location>
```

2.3.5. CUPS systemd service

The scheduler for CUPS is called **cupsd**. When run from `systemd` some systems pass the `-l` parameter, so `cupsd` is run on demand by socket and path activation. The advantage of this setup is that CUPS is activated when needed, saving precious boot time and resources, and deactivated again after being idle for a while. This lazy activation scenario is efficient for desktop systems that print occasionally and for which printing is not time critical. However, dedicated print systems like SavaPage, that intensively use IPP to communicate with CUPS, need CUPS to be full-time activated. Therefore the `systemd cups.service` unit must effectively start `cupsd` with the `-f` parameter, so it runs steadily in the foreground.

Check the `/lib/systemd/system/cups.service` unit: `ExecStart` must start `cupsd` with the `-f` parameter. If not, edit the CUPS service unit with this command:

```
sudo systemctl edit cups
```

This launches a text editor for creating the file:

```
/etc/systemd/system/cups.service.d/override.conf
```

Add the following lines:

```
[Service]
ExecStart=
ExecStart=/usr/sbin/cupsd -f
```

Save the file and close the editor. Usually, after you edited a systemd unit file, for it to take effect, you need to run:

```
sudo systemctl daemon-reload
```

However, the `systemctl edit` command automatically did this for you. You can check the effect of the override with this command:

```
systemctl cat cups.service | grep Exec
```

... it should show:

```
ExecStart=/usr/sbin/cupsd -l
ExecStart=
ExecStart=/usr/sbin/cupsd -f
```

You can check if `/usr/sbin/cupsd -f` is active with this command:

```
systemctl status cups.service
```

2.3.6. Test CUPS

When CUPS was stopped earlier you need to start CUPS again with this command:

```
sudo service cups start
```

Now you can test if the CUPS print queues to be used as [Proxy Printer](#) work as expected.

2.4. Step 4 - Optional System Settings

2.4.1. Set Default Paper Size

You can optionally set the default system paper size in the file `/etc/papersize`. This default is used by SavaPage, but can again be overridden in the Admin Web App. See [Section 4.10.14.5, “Default Paper Size” \[152\]](#).

The format of the `/etc/papersize` file is very simple: whitespace and anything starting with “#” is ignored, and the name of the paper is the first string found; the case in the name of the paper is irrelevant. Commonly valid paper size values are: a3, a4, a5, b5, letter, legal, executive, note and 11x17.

For example, use this command to set the default to a4:

```
sudo su -c 'echo a4 > /etc/papersize'
```

2.5. Step 5 - Check Firewall Settings

SavaPage uses TCP/IP port 8631 (for HTTP), port 8632 (for HTTPS/SSL) and port 9100 (for JetDirect/RAW printing) by default.



Note

You can change the TCP/IP port defaults in the `/opt/savapage/server/server.properties` file after installation. See [Section 9.3.1, “Alternative TCP/IP Settings” \[193\]](#).

For [Proxy Printer](#) access the standard IPP port 631 of local CUPS is used.

For [iOS printing](#) (AirPrint) UDP port 5353 is used.

Depending on the [Mail](#) settings common SMTP ports are 25, 465 and 587.

Common IMAP ports used for [Mail Print](#) are 143 and 993.

The [Secure JMX Connection](#) uses port 8639.

SavaPage [Google Cloud Printer](#) uses XMPP port 5222 .

Many GNU/Linux distributions have strict default firewall policies, so take some time now to ensure that these ports are open. Consult your distribution documentation for details on how to open firewall TCP and UDP ports.

2.6. Step 6 - Download and Install

SavaPage is supplied as a self-extracting and self-installing archive for 64-bit (x64 , x86_64, amd64) systems. The installation must be performed as the newly created `savapage` user in the `/opt/savapage` directory. Some parts of the installation need to be executed as `root`. When you choose to do so during the main install, please have the `root` password or `sudo` password handy. You can also choose to execute the `root` tasks *after* the main install. In that case you can simply `sudo` execute a post-install script. For more detail about the install process please see [Chapter 9, SavaPage on GNU/Linux \[190\]](#).

Change to the newly created `savapage` user, download and execute the installer in `/opt/savapage`, and follow the instructions.

```
sudo su savapage
cd /opt/savapage
rm ./savapage-setup-*-linux-*.bin ❶
wget URL ❷
sh ./savapage-setup-*-linux-*.bin ❸
```

❶ Remove old downloads, if any, first, to prevent that a new download gets `.bin.1` suffix.

❷ Use the `URL` from the [SavaPage Wiki](#)⁴ download section.

❸ For example `savapage-setup-1.0.0-linux-x64.bin`

The installation process will take between one and two minutes depending on the speed of the system. A system restart is not required. When installing on a *live* production system, administrators are advised to choose a period of low activity - for example, not during backup operations or other administration activities.

2.7. Step 7 - Save Encryption Keys

SavaPage creates the `/opt/savapage/server/data/encryption.properties` file at first installation. The encryption keys held in this file are used to store [Encrypted Secrets](#) and [Document Signatures](#) in the database.

Make a backup of this file now, and store it at a secure place, so you can restore it when you need to [migrate to a new server](#).

⁴ <https://wiki.savapage.org>



Warning

The `encryption.properties` file is *crucial* for decrypting secret data in the database and verifying the authenticity of document signatures. When you lose this file you won't be able to use any database copy which was based on its encryption keys **ever**.

2.8. Step 8 - Configure

After installation, you will be prompted to open a web browser at `https://savapage:8632/admin` to complete the configuration. The configuration steps are explained below.

2.8.1. Step 1 - Login

Login with username `admin`. This is the built-in administration account. Enter `admin` as password. This is the standard password as set by the installer.

After login the **Dashboard** is shown where you will notice the system status “Setup is needed”. The next steps guide you in configuring the system so that the status will change to “Ready to use”.



Note

As long as system setup is needed login attempts at the User Web App are blocked with a message saying “Application setup is required”.

2.8.2. Step 2 - Change Admin Password

As a first security measure change the master password for the built-in `admin` account. This account is independent and not related to the operating system or domain. The password needs to meet minimum strength requirements, and must contain at least six characters. Select **Options** → **Advanced** → **Reset internal admin password**, enter and confirm the new password and press the **Apply** button.



Caution

Make sure that this password is kept at a secure place since it is the key to your system.

More information about the admin password can be found in [Section 13.1.1.3, “Internal Admin Password” \[216\]](#).

2.8.3. Step 3 - Set Locale

Set the system's locale; ensure that these are correct before proceeding. Select **Options** → **Advanced** → **Locale**, and enter the locale string. Some examples are: `en`, `en-GB`, `en-US`, `nl`, `nl-NL`, `nl-BE`. You can leave the locale empty to accept the system default. The locale is applied to all system messages which are logged in the system log or send by email. See [Section 15.1, “Localization” \[225\]](#).

2.8.4. Step 4 - Set Currency Code

Set the system's currency code; ensure that these are correct before proceeding. Select **Options** → **Financial**, and enter the ISO 4217 Currency code. Some examples are: `USD`, `EUR`, `GBP`. The currency *symbol* is determined in the context of the user or system locale. See [Section 15.1, “Localization” \[225\]](#).

2.8.5. Step 5 - Set User Source

SavaPage optionally imports user information from a **Unix** (PAM, NIS, etc.) or **LDAP** source.

Select [Options → User Source](#).

Select **Unix** if the user accounts are setup and defined on the local system as standard Unix accounts or mapped into the system from a central directory service such as LDAP via `nsswitch.conf` and PAM. Most large established networks will use this option.



Note

For administrators wishing to customize the PAM authentication method at the application level, SavaPage reports itself as “savapage”.

The **LDAP** option is appropriate for large networks with existing LDAP domains. This includes networks running OpenLDAP, Apple Open Directory, Novell eDirectory and Microsoft Active Directory.

More information on LDAP is available in [Section 4.10.1.2, “LDAP” \[121\]](#).

After selecting the source, enter the necessary parameters (LDAP only) and press the **Apply** button.

2.8.6. Step 6 - User Synchronization

Skip this step if you did not set an external User Source in the previous step. Otherwise, select [Options → User Creation → Synchronization → Synchronize now](#) to import users.



Important

An option exists to import a subset of users from the source by selecting a *group*. This option is relevant if only a subset of users will ever use SavaPage. Select [Options → User Creation → Change Group](#) to select the group.



Tip

Test the import first by pushing the **Test** button. A simulated import will start, with each step echoed below the button, so you can verify the effect of your action.

2.8.7. Step 7 - Set Mail Options

Select [Options → Mail](#), enter the SMTP and Message options and press the **Apply** button. Data from the Messages section is used for system generated mail messages.

You can send a test mail message to a recipient of your choice by pressing the **Test** button after you applied the changes.

2.8.8. Step 8 - Driverless Printing

Mail Print and Web Print are disabled by default. You can enable and configure these options at [Options → Mail Print](#) and [Options → Web Print](#).

If you enabled one of the driverless printing options, decide which PDF converters you want to enable at [Options → Advanced → Converters](#). Beware that you might need to install the converter software on the SavaPage host.

2.9. Step 8 - Share SavaPage Client Files

SavaPage client files are located in directory `/opt/savapage/client`. This includes the [SavaPage Printer Driver](#) and [JMX related files](#). It is useful to share this directory over the network so users can use, copy or install the files they need on their workstation. Common sharing methods include:

- *Samba* - used to share files to Windows based workstations. GUI tools are available on GNU/Linux to help you with sharing the client directory via Samba. However, some system administrators may be more comfortable creating the share by hand-editing the `/etc/samba/smb.conf` file. The following configuration will share the directory in read-only form:

```
[savapage-client]
path = /opt/savapage/client
comment = SavaPage Client
public = yes
only guest = yes
read only = yes
```

- *NFS* - a popular sharing method used for GNU/Linux and Unix based workstations.



Note

The `/opt/savapage/client` directory is standard shared via the `client/` URL. See [Appendix E, URL Cheat Sheet \[278\]](#).

2.10. Step 9 - Testing

Now the installation is complete, it is time to do a basic test to check if the system is ready-to-use.

- Pick a workstation and login as a user that is part of the user source as configured in [Section 2.8.5, “Step 5 - Set User Source” \[18\]](#).
- Install the SavaPage Printer Driver. See the instructions at [Section 10.1, “Printing with a Driver” \[195\]](#).
- Open a Web Browser and go to the User Web App at `https://savapage:8632/`
- Login to the Web App with the same credentials as used in the workstation login.
- Print a test document such as a web page or basic document to the SavaPage printer.
- Thumbnail images of the printed pages should appear in the Web App.

2.11. What's next?

Congratulations! At this point you have a ready-to-use SavaPage system. This concludes the Install Guide.

If you like, take some time to further explore the features of SavaPage in a more extensive free-format test drive.

Or, continue reading about the user interface details at [Chapter 3, User Web App \[22\]](#), [Chapter 4, Admin Web App \[74\]](#), [Chapter 5, Job Tickets Web App \[174\]](#) and [Chapter 6, Point-of-Sale Web App \[183\]](#).

At this point you can also proceed with the configuration of [Google Cloud Print](#), [AirPrint](#), [Mail Print](#) and [Web Print](#).

[Chapter 7, User Client \[186\]](#) explains how to use a *system tray* notifier of SavaPage print events for desktops and notebooks.

Chapter 8, *SavaPage Financial* [189] introduces the main pay-per-print concepts with references to more detailed parts of the manual.

Chapter 9, *SavaPage on GNU/Linux* [190] offers an in-depth explanation of the GNU/Linux installation process, the directory layout and tools involved.

Chapter 10, *SavaPage as Printer* [195] explains how for print from different platforms.

Chapter 11, *Authenticated Printing* [206] describes how SavaPage determines the digital identity of users in different settings like Single Sign-On (SSO) Domains and Peer to Peer Networks.

Chapter 12, *Printing Impact* [214] explains the metrics used when giving users feedback about the costs and environmental impact of their printing habits.

Chapter 13, *Security* [216] discussed security issues and precautions.

Chapter 14, *Privacy* [223] explains how digital freedom and privacy is secured in the SavaPage domain.

Chapter 15, *Internationalization* [225] explains how SavaPage is adapted to various languages and regions.

Chapter 16, *Customization* [227] explains how SavaPage can be customized to fit your corporate identity.

Chapter 17, *Using an External Database* [236] explains how to use an alternative *external* relational database.

Chapter 18, *Tuning* [239] discusses performance optimization and parameter tuning.

Chapter 19, *SavaPage Community* [245] describes the SavaPage Community and explains how to use the Member Card.

Appendix A, *Proxy Print Scenarios* [247] summarizes several Proxy Print scenarios in a shorthand catalogue.

Appendix B, *NFC Authentication* [251] explains how SavaPage supports RFID as authentication method.

Appendix C, *Tools* [253] explains the command-line interface for calling server methods, manipulate the database, stop and start the server, and for applying SSL certificates for secure HHTTP connections.

Appendix D, *Capacity Planning* [276] discusses how SavaPage uses disk space and network resources.

Appendix E, *URL Cheat Sheet* [278] offers a Quick Reference Card of the available Web Interface URLs.

Appendix F, *File Locations* [280] is an outline of locations and files in `/opt/savapage/`.

Appendix G, *Printable File Types* [282] gives a summary of the file formats supported by **Driverless Printing**.

Appendix H, *Upgrading from a Previous Version* [285] describes the procedure to install a new version.

Appendix I, *Migrating to a New Server* [286] describes the procedure to move your current SavaPage installation to a new server.

Appendix J, *Advanced LDAP Configuration* [288] gives an in depth explanation of the LDAP configuration options.

Appendix K, *PPD Extensions* [292] explains how to map vendor specific PPD keywords to IPP attributes.

Appendix L, *IPP Extensions* [305] gives a summary of IPP attributes and values as extensions to the IANA registered ones.

Appendix M, *SavaPage Plug-ins* [311] explains how to deploy software components that add specific features to SavaPage.

Appendix N, *PaperCut Integration* [315] explains how functions not present in PaperCut can be implemented with SavaPage as pre-processor and integrator.

Appendix O, *Job Scheduling* [320] describes SavaPage background job scheduling.

Appendix P, *GNU Affero General Public License (AGPL)* [321] contains the full text of the AGPL.

Chapter 3. User Web App



The User Web App can be reached at `https://savapage:8632/`. For all URL options see [Appendix E, URL Cheat Sheet](#) [278].



Important

When using the User Web App concurrently with the [User Client](#) and [Proxy Print Authentication](#) you are strongly advised to use an external database like PostgreSQL. See [Chapter 17, Using an External Database](#) [236].

3.1. Login

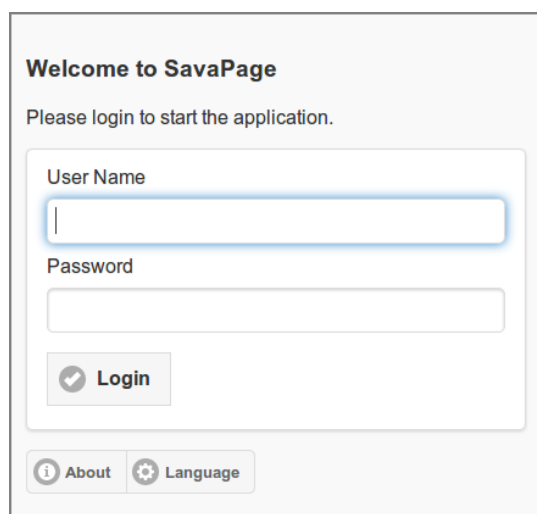


Figure 3.1. Web App: Login Dialog



Note

When a user opens the Web App the login dialog is skipped when an [Authentication Token](#) is present in *local storage* of the browser. The login dialog is also skipped when the Web App is opened from a trusted and authenticated [User Client](#) or with a [One-Time Authentication Token](#).

For a description of the global user authentication defaults see [Section 4.10.3, “User Authentication”](#) [126].

- The language of the dialog defaults to the language setting of the browser.
- You can overrule the default language and country or preselect a user by using the URL parameters. See [Appendix E, URL Cheat Sheet](#) [278]
- Version and copyright information is shown when you press the `About` button. See [Section 3.1.1, “About”](#) [23].
- You can choose an alternative language by pressing the `Language` button. See [Section 3.1.2, “Select Language”](#) [23].

- The top of the Login dialog can be customized: see [Section 16.1.1.3, “Custom HTML”](#) [229].
- [Login Alternatives](#) appear at the bottom of the dialog.

Some invariants:

- Only Persons can login.
- [Disabled users](#) are not allowed to log in.
- The internal "admin" user is not allowed to log in as user.
- As long as system setup is needed user login attempts are blocked with a message saying “Application setup is required”.



Tip

You can use an *alias* as **User Name**. See [Section 11.4, “User Name Aliases”](#) [212].

3.1.1. About

The About dialog shows version and copyright information. The top of the dialog can be customized: see [Section 16.1.1.3, “Custom HTML”](#) [229].



Note

The dialog contains a Printer Driver section with a download link for the CUPS [SAVAPAGE.ppd](#) file. This section can be enabled or disabled by setting the configuration key `webapp.about.driver-download.enable` to Y or N. See [Section 4.10.14.10, “Config Editor”](#) [155] on how to change this value.

3.1.2. Select Language

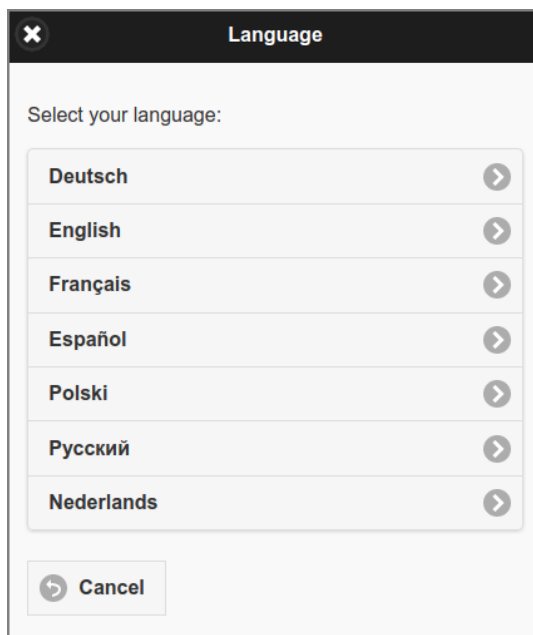


Figure 3.2. Web App: Select Language Dialog

- At the moment [English](#), [German](#) and [Dutch](#) are fully supported. Press the language of your choice. This will reload the login dialog in the newly selected language.

- Press **Cancel** to return to the login dialog.



Note

You can use the `webapp.language.available` configuration property to enter a comma separated list of selectable ISO 639 language codes. For instance: `de,en,fr,es,ru,nl`. See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to enter this value.

When the list is restricted to a single language, the **Language** button in is not shown in the **Login** dialog, and the Web App Locale is set to the single available language.

3.1.3. Single Web App Session

A warning message is shown when a *desktop*¹ user tries to open the *same* Web App type a second time in the same browser instance. In rare occasions, for instance, due to network connectivity issues, this statement might be false. In that case a Login will bring you back on track.

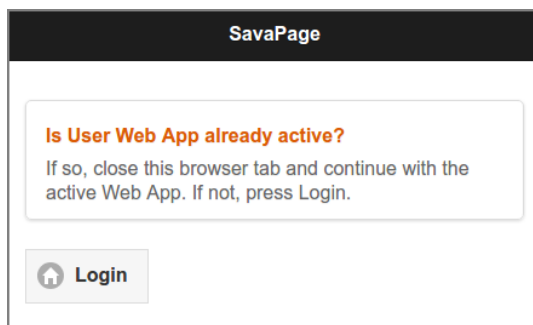


Figure 3.3. Same type Web App session detected

When a user opens a second Web App session of *another* type this message is shown:

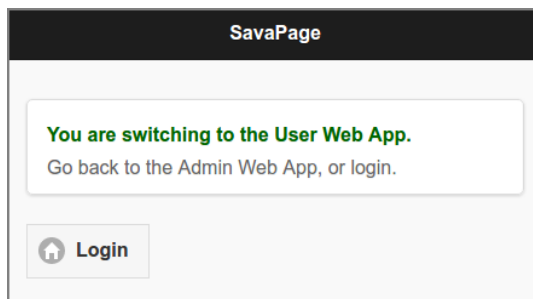


Figure 3.4. Web App type change detected

In both situations, either go back to the active Web App session or press **Login** to login to the intended Web App type. This will invalidate any other SavaPage session in the same browser instance.

3.1.4. Login Alternatives

The appearance of the Login dialog on a device depends on the following settings:

- The globally activated Login Methods. See [Section 4.10.3, “User Authentication” \[126\]](#).

¹The Single Web App Session check is solely done for certain *desktop* browser sessions. Sessions on macOS and mobile devices are *not* checked.

- The Terminal settings for the device. See [Section 4.9.3, “Custom User Login” \[118\]](#).
- The login URL parameter with the preferred login method. See [Appendix E, *URL Cheat Sheet* \[278\]](#).
- Active [OAuth Client Plug-ins](#).

Terminal settings overrule global settings, and the URL parameter overrules the defined default. When available, alternative login methods can be selected by tapping the method button at the bottom of the dialog. Some sample Login dialogs are shown below.

Welcome to SavaPage

Please login to start the application.

ID Number

PIN

☐ Login

Login with

User Name NFC Card YubiKey

Figure 3.5. Web App: Login Dialog - ID Number

Welcome to SavaPage

Please login to start the application.

Swipe your card . . .

Login with

User Name ID Number YubiKey

Figure 3.6. Web App: Login Dialog - Local NFC Card

Welcome to SavaPage

Please login to start the application.

(((• Swipe your card . . .

Login with

User Name ID Number

Figure 3.7. Web App: Login Dialog - Network NFC Card



Tip

A special URL path is available to use OAuth as SSO for User Web App. This URL can be presented on the site of an OAuth provider, to seamlessly link to SavaPage. See [Appendix E, URL Cheat Sheet \[278\]](#).

3.1.5. Card Self Association Dialog

When an unknown card is swiped, and Card Self Association is enabled, the user is presented this dialog to associate the new card.

Figure 3.8. Web App: Login Dialog - Card Self Association

There is a time limit to enter the Username and Password. The remaining seconds are shown and when counted down to zero the dialog is automatically closed. The time limit (seconds) is contained in configuration key `webapp.card-assoc.dialog-max-secs`. See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change this value.

3.2. SafePages

This is the main view with the acquired SafePages since the last login.

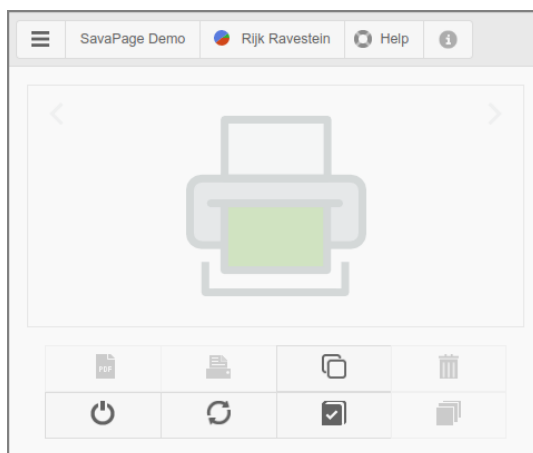


Figure 3.9. User Web App: Main View

The top bar has the following buttons (from left to right):

- An action button that opens a pop-up menu. For now, the SavaPage logo with the **GDPR** and **About** button is available. See [Section 3.12, “GDPR Dialog” \[72\]](#).
- The **Community Member** name, in our case "SavaPage Demo". Pressing it opens the **About** dialog with version and copyright information.
- A button with the inline pagometer Pie-Chart followed by the id of the logged in user. The *blue* color in the chart represents the number of pages the user printed to SavaPage. The *green* color represents the number of pages exported to PDF. The *red* color depicts the pages printed to Proxy Printers. The button opens a dialog with **User Details**, including pagometer details.
- The **Help** button opens a custom URL in a new tab. The URL is set with configuration property `"webapp.user.help.url"` and enabled with `"webapp.user.help.url.enable"` Y (default) or N.
- The **i** button opens the **About** dialog.

The two-line center bar holds buttons for the main work-flow. When no SafePages are present, the SavaPage logo is shown, and irrelevant buttons are disabled (these buttons are described at [Figure 3.10, “User Web App: SafePages” \[28\]](#)).


By default, the center bar shows button text on mobile devices only: on desktops a hover text is shown. You can change this behavior by setting configuration property `webapp.user.main.nav-button-text` with value ON (button text is always shown), OFF (button text is never shown) or AUTO (button text is shown on mobile devices only).


See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change configuration properties.





Important

Depending on **User Privileges** and **User Roles** some buttons might not be shown, replaced (relocated).

The  **Letterhead** button brings you to a dialog where you can create, browse and select letterheads See [Section 3.6, “Letterheads” \[57\]](#).

Press the  **Log** button to get a list of the See [Section 3.8, “Log” \[60\]](#).

The  **Logout** button brings you back to the login screen.

Press the  **Refresh** button when, due to whatever reason, the automatic detection of SavaPage changes fails. This will update the view with the latest state.



Note

Each print to SavaPage is logged as **Document**. SafePages that do not match a logged Document are removed. This can happen when a database is restored, or when old documents are deleted after a **Database Backup**, or **Database Command**.

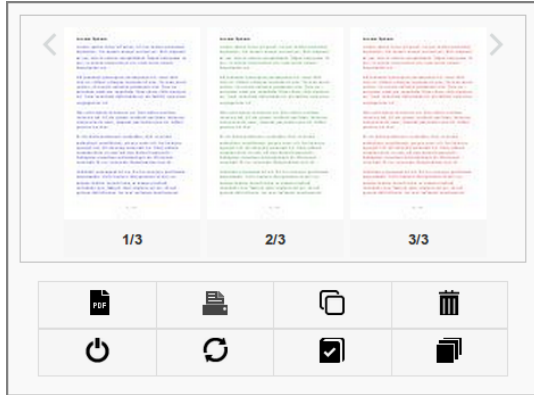


Figure 3.10. User Web App: SafePages

This screen shows the result of a user who printed 3 pages to the SavaPage printer. The following actions on thumbnails are defined:

- You can scroll through the thumbnails by dragging them horizontally.
- A *tap* on the transparent area "<" or ">" in the top corner of the thumbnail view port scrolls the view a single page to the left or right. A *taphold* brings the start or the end of the page range in view.
- A *tap* (click) opens up a detailed view of the page in the Page Browser: [Figure 3.19, “User Web App: SafePage Browser \(8 pages\)” \[34\]](#).
- A *tap* on the page number underneath the thumbnail or a *taphold* on the thumbnail itself, opens the Document Details dialog with rotate, delete and undo actions: [Section 3.2.3, “Document Details” \[32\]](#).

The following sections describe the actions for the newly enabled buttons:

 [Section 3.3, “PDF” \[35\]](#).

  When the user is a [Job Ticket Creator](#) and *not* a [Print Job Creator](#), or [Job Ticket Printers](#) are the only available printers, a Ticket button is shown instead of a Printer button. [Section 3.4, “Print” \[41\]](#).

 [Section 3.7, “Delete” \[59\]](#).

 [Section 3.9, “Sort” \[65\]](#).

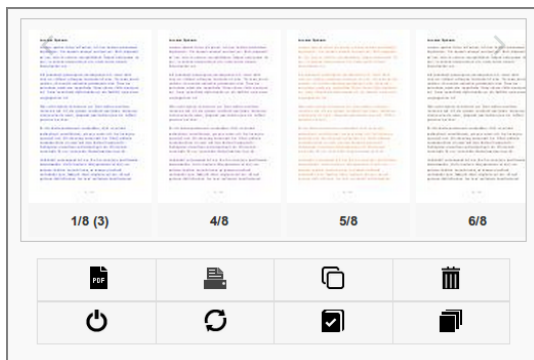


Figure 3.11. User Web App: SafePages - Aggregated

This screen shows the result of a user printing 8 pages to the SavaPage printer, and illustrates *thumbnail aggregation*.

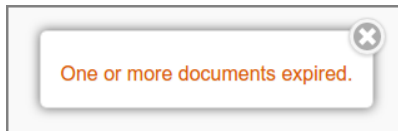
- Note that only 6 thumbnails are displayed, and that the first thumbnail tells by its numbering that it is the first of a (3) page aggregation.
- A *tap* on the first thumbnail will bring the (3) aggregated thumbnails in view. As a side-effect an aggregate will appear at another location in the thumbnail sequence.
- Thumbnail aggregation is a protection against information (and resource) overload. Imagine what would happen if you printed a 500 page document to the SavaPage printer and ended up with 500 thumbnails. Aggregation gives you the high-level means to easily zoom in and out.
- As always, a *tap* on a single thumbnail will bring you to the Page Browser, where you can navigate to any page, sequentially or directly. See [Figure 3.19, “User Web App: SafePage Browser \(8 pages\)” \[34\]](#).

3.2.1. Document Expiration

When the document expiration time signal is set and expiration for a document is due, every thumbnail of the document is marked with a clock icon in a colored (orange) footer.



When the document is auto-deleted after expiration a notification message is shown.



See [Section 4.10.14.8, “SafePages” \[154\]](#).

3.2.2. Footer

The footer is positioned at the bottom of the main user panel. The base items are depicted in the figure below.

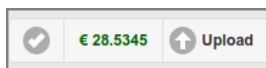


Figure 3.12. User Web App: Footer Base

From left to right:

- A *check-mark* icon as indication that the Web App is connected to the SavaPage server. Other icons are shown when the connection is being (re) established or lost. When the server is not down, this usually is a temporary condition due to a network hiccup. Don't worry, SavaPage will automatically restore the connection when the network permits.
- The account balance of the logged in user. When you press the button a dialog with [User Details](#) is shown, including pagometer details. This button might not show depending on [Financial User Privileges](#).
- A tap on the **Upload** button shows the [Upload File](#) dialog. This button may have moved to the center button bar, depending on [User Privileges](#).



Note

Depending on **User Privileges** some buttons might changed, not shown or moved to the main button bar.

3.2.2.1. Paper Size Indicator

When SafePages are present the unique paper sizes of the jobs are depicted in the footer. The text color indicates if a paper size is supported by the selected printer or not. The examples below illustrate how this works.



A4 and A3 jobs are present: a printer is selected that supports both paper sizes.



A4 and A3 jobs are present: a printer is selected that supports solely the A4 paper size. The A3 page is *cropped* to A4.



A4 and A3 jobs are present: a printer is selected that supports solely the A4 paper size. The A3 page is *shrunk* to A4.



A4 and A3 jobs are present: a printer is selected that supports none of the paper sizes (or no printer is selected yet).

3.2.2.2. Hold Print Jobs

A summary of **Hold Print Mode** jobs and **Job Tickets** as a result of **Proxy Printing** is shown in the footer. The example below explains the layout.



From left to right:

- The shortest remaining job release time.
- The total number of hold jobs.
- The total number of pages to be printed.
- The total cost charged for printing.

A tap on the summary show a dialog where all Hold Jobs are shown in detail. Each job has three button, from left to right:

- Show invoicing information.
- Preview the PDF document to be printed.
- Cancel the job.

And for all jobs:

- **Cancel all** jobs.
- **Extend** the expiry time of **Hold Print Mode** jobs.

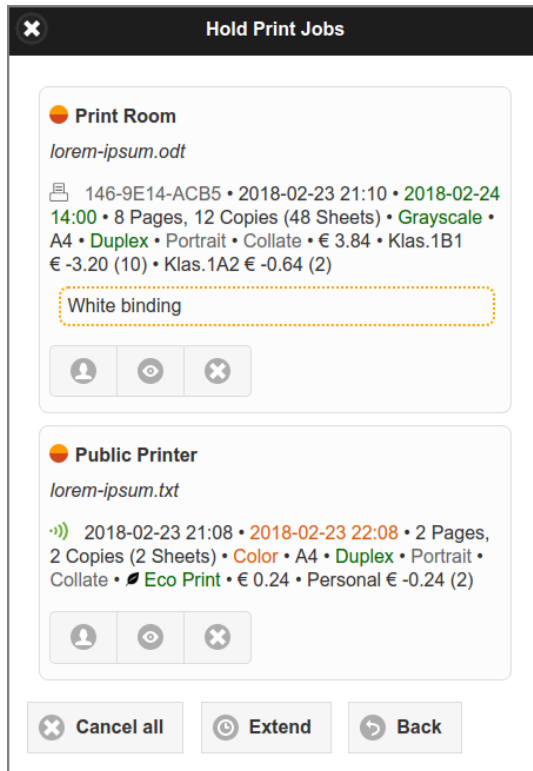


Figure 3.13. User Web App: Hold Print Jobs

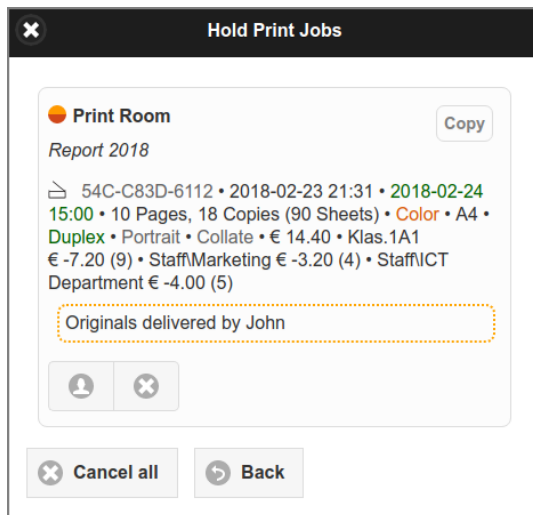


Figure 3.14. User Web App: Hold Copy Job

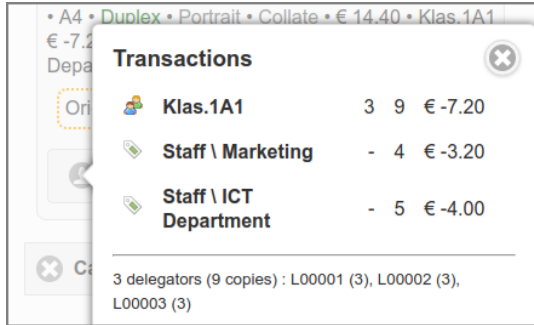


Figure 3.15. User Web App: Hold Job Transactions

More information can be found at:

- [Section 3.4.4, “Print Job Settings” \[42\].](#)
- [Section 3.4.8, “Job Ticket Print” \[51\].](#)

3.2.3. Document Details

A *Tap* on the page numbering below the image pops up the Document Details dialog.

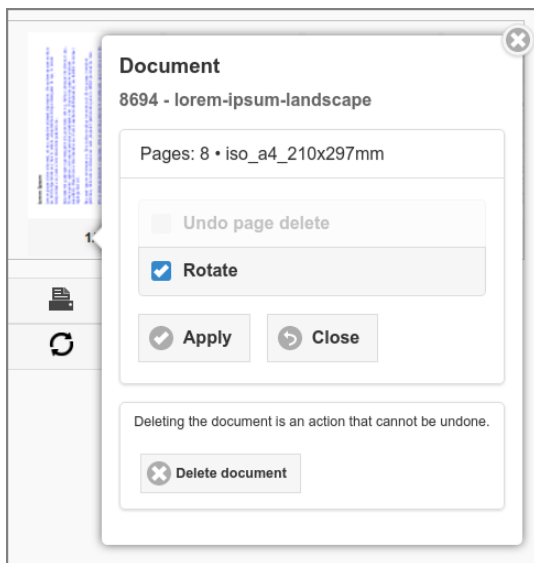
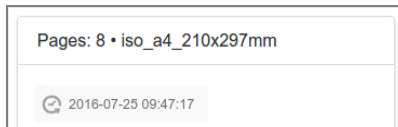


Figure 3.16. User Web App: Document Details

When document expiration is set the expiration time is shown. See [Section 4.10.14.8, “SafePages” \[154\].](#)



3.2.3.1. Delete and Undo

Press the **Delete document** button in this dialog to *delete* all pages of a SafePages document, or check and apply **Undo page delete** to restore the full job in case document pages were deleted. See [Section 3.9, “Sort” \[65\].](#)

3.2.3.2. Rotation

When a user prints to a printer and selects landscape orientation, the print manager of the originating application will translate and rotate the printed content to fit the dimensions of the hard copy target page.

When doing so, it makes assumptions about the (0,0) origin of the logical space on this page. The SavaPage printer driver provides a hint to the print manager about the origin, so it can rotate and translate the pages in a way that is compatible with the SavaPage printer.

Contrary to real printers, where hard copies can easily be rotated by hand, pages produced by the virtual SavaPage printer need special attention, since landscape oriented prints will display rotated in portrait oriented images and PDF pages. Probably this is not what you want, so you can ad-hoc rotate job pages in SavaPage to landscape display orientation.



Figure 3.17. User Web App: Landscape Job

- This what you might see when you print a job in landscape orientation.
- Select the *Rotate* option and press the **Apply** button to rotate the page and all sibling pages belonging to the same job.
- The result after rotation is shown in [Figure 3.18, “User Web App: Rotated Pages” \[33\]](#).



Note

Although the Rotate dialog is triggered from a *single* SafePage, the rotation affects *all* SafePages within the same print job.



Figure 3.18. User Web App: Rotated Pages

- The SafePages after rotation.



Warning

When two WebApps are opened for the same user, the result of a page rotation performed in one Web App will not automatically be shown in the other. The user should do a manual refresh to update the SafePages preview.

3.2.4. Browser

A *tap* on a non-aggregated SafePage thumbnail image will show the page detail in the Browser.

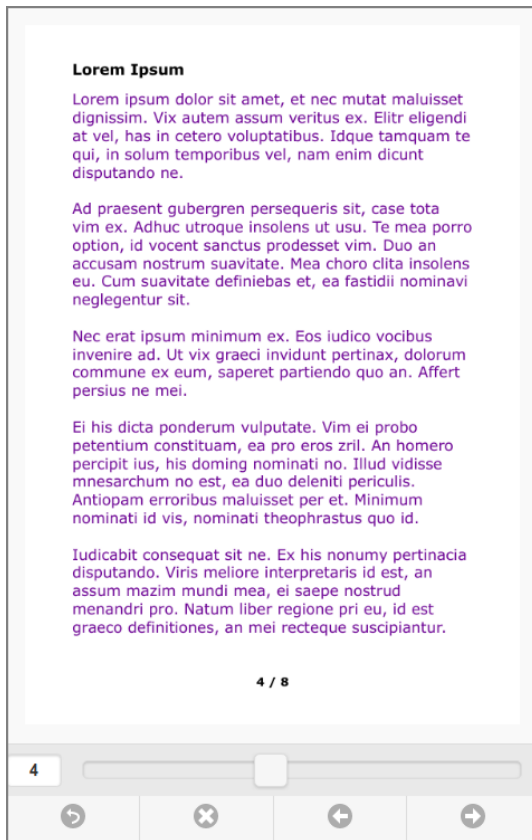


Figure 3.19. User Web App: SafePage Browser (8 pages)

- A *tap* on the page image zooms in, extending the image width to the available screen width. See [Figure 3.20, “User Web App: SafePage Browser - Detailed View \(4 of 8\)” \[35\]](#).
- There are several ways to browse the pages:
 - *Swipe* the page image to the left or right to view the next or previous page. A *swipe-left* on the last page brings you back to the first page. Vice versa, a *swipe-right* on the first page brings the last page into view.
 - The *arrow-right* and *arrow-left* buttons in the navigation bar below are an alternative for swiping to a next or previous page.
 - Use the *slider* control to directly jump to the page of your choice.
- The *X* button deletes the page in view.
- The leftmost *Return* button brings you back to the main SafePages screen: [Section 3.2, “SafePages” \[26\]](#).

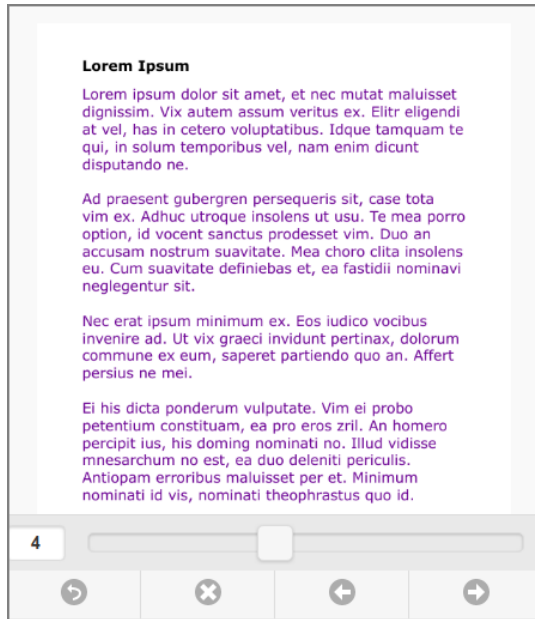


Figure 3.20. User Web App: SafePage Browser - Detailed View (4 of 8)

- This screen shows a zoomed-in detailed page view. The image width is extended to the available screen width. Use the standard page scrolling of your browser to scroll the image up and down.
- A *tap* on the page image zooms out again, adjusting the image height to the available screen height. See [Figure 3.19, “User Web App: SafePage Browser \(8 pages\)” \[34\]](#).



Tip

The detailed view automatically adjusts itself when the available screen width changes, either by tilting your mobile device from portrait to landscape orientation (vice versa) or by resizing your desktop browser window.

3.3. PDF

A *tap* on the **PDF** button in the main SafePages view shows a dialog with PDF properties and export actions. See [Section 3.2, “SafePages” \[26\]](#).



Note

PDF properties are preserved when the dialog is closed or a PDF is generated, and are re-used when needed in current or future sessions.

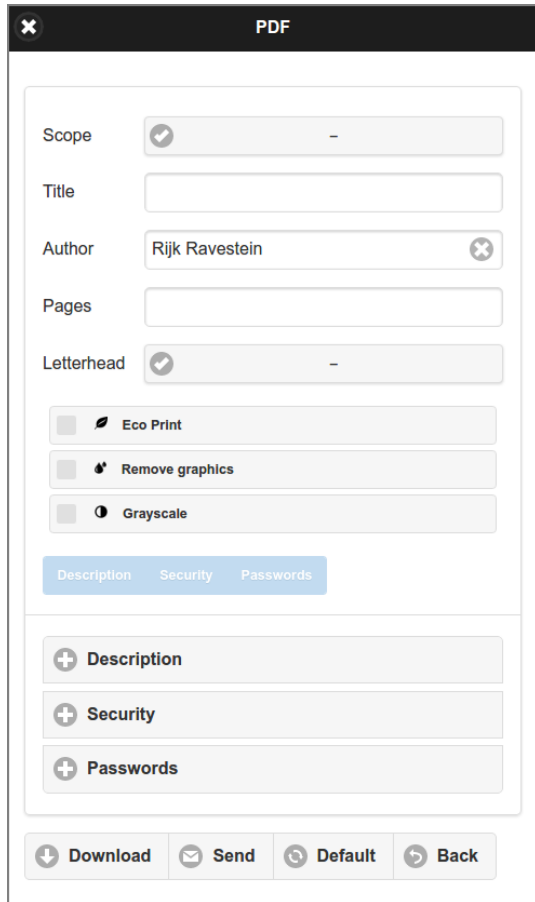


Figure 3.21. User Web App: PDF - Overview

- This screen shows the full PDF dialog.
- The *Author* defaults to the name of the authenticated user at login, and can be edited.
- Details are discussed at:
 - [Section 3.3.1, “PDF Filters” \[37\]](#)
 - [Section 3.3.2, “Document Scope” \[37\]](#).
 - [Section 3.3.3, “Description” \[37\]](#).
 - [Section 3.3.4, “Security” \[38\]](#).
 - [Section 3.3.5, “Passwords” \[39\]](#).
 - [Section 3.3.6, “Letterhead” \[40\]](#).
 - [Section 3.3.7, “Download” \[40\]](#).
 - [Section 3.3.8, “Send” \[40\]](#).
- A selection of SafePages to incorporate into the PDF output can be entered as a range of **Pages**. For example: 1–4 , 6 , 8–10. The value can be a single page, a range of pages, or a collection of page numbers and ranges separated by commas. The pages will always be exported in ascending order, regardless of the order of the pages in the page-ranges option. The page range is automatically emptied after printing. Be aware that the page ordinals are related to the [Document Scope](#).
- Check one of the [PDF Filters](#) [Eco Print](#) , [Remove graphics](#) or [Grayscale](#) .
- You can activate a [Description](#) , [Security](#) and [Passwords](#) setting by toggling the corresponding button in the so-called **Apply** section. A toggle button is disabled when no setting is specified.



Note

SavaPage tries to translate URL formatted text like “www.example.com” and “info@example.com” to PDF links. Implicit URLs in the source document, such as those contained in text like “click here”, are not sent to the SavaPage printer, and therefore not preserved in the PDF document.

3.3.1. PDF Filters

Activate *either* the **Grayscale**, **Eco Print** or **Remove graphics** filter option. The last option removes all graphic images from the PDF. When choosing **Eco Print** the selected SafePages will be ad-hoc converted in the background. While conversion is busy a message box will tell you to wait a while and retry later. Take about 3 seconds per page waiting time into consideration. Automatic filtering may help to diminish waiting times: see [Section 4.10.11.2, “Eco Print Settings”](#) [145].



Note

At the moment a *single* filter can be selected. If needed filter chains will be supported in a future SavaPage version.

3.3.2. Document Scope

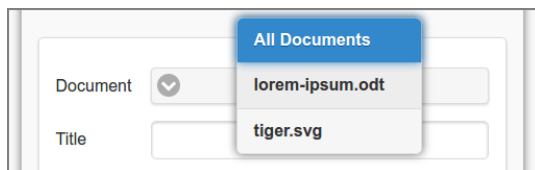


Figure 3.22. User Web App: PDF - Document Scope

- When pressing the **Document** button you get a selection pop-up with titles of acquired SavaPage print jobs. *Tap* a job title to restrict the scope of the PDF output to that job. Select the top item **All Documents** to activate *full* scope.
- When *full* scope is selected the **Pages** ordinals are related to the SafePages total. When a job is selected as scope, the page ordinals are related to the page total of the job.
- Selecting a scope initializes the **Title** text with the job name, or clears it when you choose full scope. You can edit the **Title** if you wish.
- If there is just a *single* acquired SavaPage print job, this job is shown as only option.



Note

When a user **rearranged** or **deleted** any SafePages the scope is confined to *full* scope.

3.3.3. Description

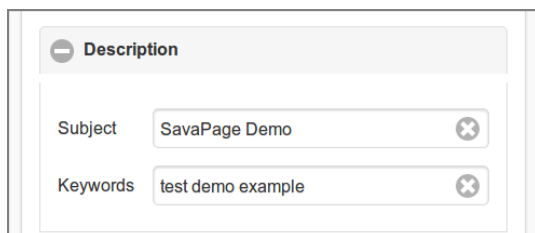


Figure 3.23. User Web App: PDF - Description

- Press the **Description** button to expand the section.
- Enter the **Subject** and (space separated) **Keywords** of the PDF document.



Note

When a Subject or Keywords are entered, the **Description** toggle in the **Apply** section will be enabled. Use this toggle to apply (or deny) the Description to the generated PDF.

3.3.4. Security

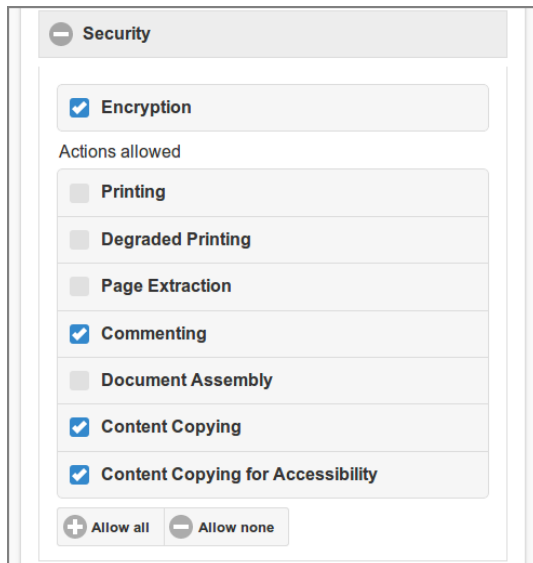


Figure 3.24. User Web App: PDF - Security

- Press the **Security** button to expand the section.
- Select the **Encryption** option to select the allowed actions on the generated PDF.
- Use the **Allow all** or **Allow none** buttons to select the actions in one go. Or select each allowed action separately.
- This is a list with the allowed actions, each with a short description:
 - *Printing*: Printing the document.
 - *Degraded Printing*: same as *Printing*, but with a lower quality.
 - *Page Extraction*: Modifying the contents. For example, changing the contents of a page, or inserting or removing a page.
 - *Commenting*: Adding or modifying text annotations.
 - *Document Assembly*: Inserting, removing, rotating and bookmarking pages. The content can't be changed, unless *Page Extraction* is also selected.
 - *Content Copying*: Copying or otherwise extracting text and graphics from the document, This also applies for screen readers or other accessibility devices.
 - *Content Copying for Accessibility*: Extracting text and graphics for use by accessibility devices.



Note

SavaPage uses 128-bit PDF encryption.



Note

When **Encryption** is selected, the **Security** toggle in the **Apply** section will be enabled. Use this toggle to apply (or deny) the security settings to the generated PDF.

3.3.5. Passwords

Figure 3.25. User Web App: PDF - Passwords

- The *User* password (also known as the *open password*) locks the PDF file for anyone who doesn't know the password.
- The *Owner* password (also known as the *permissions password*) is needed to read the PDF file in order to change the permissions.
- The maximum password length is 32 characters.
- If you don't enter a user password, all users will be able to open the PDF document without being prompted for a password. However, the security settings will remain in place.
- When both PDF user and owner password are entered they must be different.



Important

When a User password is set or Security settings are active, and the Owner password is *not* set, SavaPage will replace it by a random string.



Warning

Security settings *without* a User password aren't really secure, since the encryption key is derived from the User password. When the User password is omitted, the content is encrypted as described in the public PDF reference, so decryption is also known in this case (although illegal to practice).



Note

When a User or Owner password is entered, the **Passwords** toggle in the **Apply** section will be enabled. Use this toggle to apply (or deny) the passwords to the generated PDF.

3.3.6. Letterhead

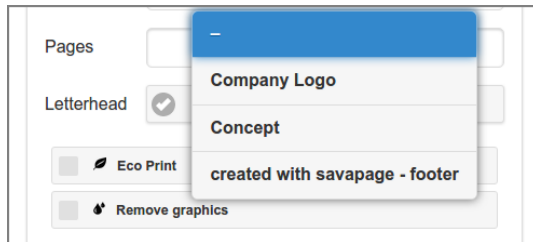


Figure 3.26. User Web App: PDF - Letterhead

- Press the button in the Letterhead section to get a selection pop-up with available Letterheads.
- *Tap* on a letterhead title to select it, or select the top item *dash* to deselect a letterhead.



Note

Depending on [User Privileges](#) the Letterhead section might not be shown.



Note

Letterheads are not subjected to [PDF Filters](#) but are applied to the filtered result.

3.3.7. Download

- Press the **Download** button to download the SafePages as PDF file, with the properties set in this dialog.
- Your browser will present a Save dialog so you can save the PDF file in the location of your choice.
- The default PDF file name will be identical to the **Title** you entered as PDF property.



Note

Depending on [User Privileges](#) the **Download** button might not be shown.

3.3.8. Send

A *tap* on the **Send** button in the PDF dialog, shows this Send dialog. See [Section 3.3, “PDF”](#) [35].

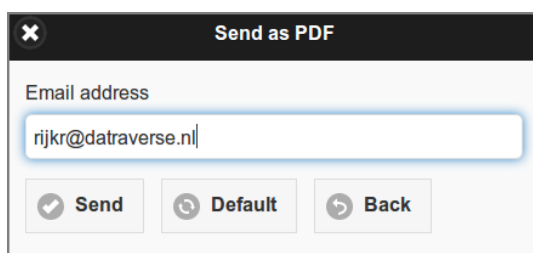


Figure 3.27. User Web App: PDF - Send

- Enter the Email address of the recipient.

- The last used Email address is shown. Press the **Default** button to reset the address to the one that belongs to the logged in user.
- Press the **Send** button to generate the PDF document and send it as attachment to the recipient.



Note

Depending on **User Privileges** the **Send** button might not be shown.

3.4. Print

A *tap* on the **Printer** or **Ticket** button in the main SafePages view shows the Print dialog. See [Section 3.2](#), “SafePages” [26].

*If just one printer is available, it is automatically selected, and its **Printer Settings** dialog is opened.*



Note

The Print dialog enables users to set custom printer and job options. When a single copy with default printer options is required, users can apply **Fast Print Mode** (when this mode is configured for a printer).

3.4.1. Printer Selection

When a printer was not yet selected a Select Printer dialog is displayed.

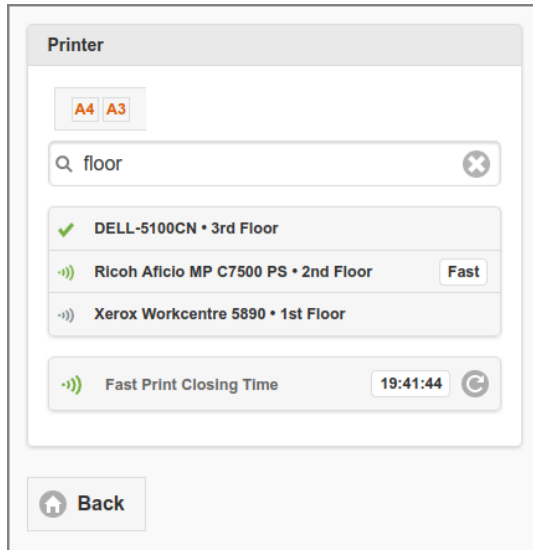





Figure 3.28. User Web App: Print - Select Printer

- A list with a maximum of 5 accessible printers is shown in alphabetic order. Access is according to **User Roles** “Print Job Creator” and “Job Ticket Creator”.
- Printers marked with an  icon are secured with **Hold Print Mode** Release. An  icon means the printer is secured with **Direct Print Mode** Release. Printers marked with a **Fast** label are (additionally) enabled for **Fast Print Mode** Release.
- **Job Ticket Printer** instances are marked with an .

- Printers can be selected by entering part of the printer name or location. In the figure above the text “floor” was entered, resulting in 3 hits.
- You select a printer from the list by tapping (clicking) it. This brings the **Printer Settings** dialog of the selected printer into view.
- The **Fast Print Closing Time** button shows the expiration time of Fast Print Release. The time is reset when the Print Dialog opens or the button is pushed. This button is shown when the user is granted access to at least one proxy printer with Fast Print Release enabled.



Note





When User Role permits access to Job Ticket Printers only, the dialog will show “Ticket” as title.

3.4.2. Selected Printer

The selected **Printer** is shown at the top section of the dialog.



Figure 3.29. User Web App: Print - Selected Printer

- The selected printer is shown as button at the bottom of the **Printer** section. **Settings** of the selected printer can be changed by pushing the button.
- The **Page Size Indicator** is shown at the top with symbols for the main printer options:
 -  : color printing.
 -  : black and white printing.
 -  : duplex printing.
 -  : two or more pages on one sheet.
- Another printer can be selected by reentering the search text (you can clear the quick search first by pushing the “cross” button at the right). *If just one printer is available no search option is offered.*

3.4.3. Delegated Print

The Delegated Print Invoicing option is offered in the **Printer Settings** and **Print Job Settings** dialog when the following conditions are met:

- **Proxy Print Delegation** is enabled.
- The User has **Print Job Delegate** role.

See [Section 3.4.7, “Delegated Print Edit” \[46\]](#).

3.4.4. Print Job Settings

Print job settings can be entered in the **Job** section.

Figure 3.30. User Web App: Print - Job Settings

Select an **Account** to which the costs of the job is charged. The account can either be a Personal (default) or a **Shared Account** the user has access to. See [Section 4.6.2, “Edit Account”](#) [97].

- **Account** selection is *hidden* when **Copies**, collected by **Delegated Print Invoicing**, is switched On. Availability of this switch is dependent on configuration property `webapp.user.proxy-print.delegate-copies-apply-switch`.

Selecting a **Document** scope and **Title** for your print job is identical as in [Section 3.3.2, “Document Scope”](#) [37].

The number of **Copies** can be entered via the slider control. The “maximum number of copies per job” is set as **Proxy Print Option**. The number of copies is automatically reset to one (1) after printing.

- **Copies** selection is *hidden* when **Delegated Print Invoicing**, is switched On.
- When **Delegated Print Invoicing** is switched Off, and a **Job Ticket Printer** is selected, the number of **Copies** is not restricted and can thus be entered as text.

A selection of SafePages to print can be entered as a range of **Pages**. For example: 1–4, 6, 8–10, 15–. The value can be a single page, a range of pages, or a collection of page numbers and ranges separated by commas. The pages will always be printed in ascending order, regardless of the order of the pages in the page-ranges option. The page range is automatically emptied after printing. Be aware that the page ordinals are related to the **Scope**.

Selecting a **Letterhead** for your print job is identical as in [Figure 3.26, “User Web App: PDF - Letterhead”](#) [40].

- Depending on **User Privileges** the Letterhead section might not be shown.

Check one of the **PDF Filters** **Eco Print** or **Remove graphics**.

- The **Eco Print**, including discount percentage, must be enabled in [Section 4.10.11.2, “Eco Print Settings”](#) [145].

- See the Configuration Properties table below on how to disable the **Remove graphics** filter.

The **Collate** option is shown when you print multiple copies and describes how printed material will be organized. For example, if you have a five page document and are printing multiple copies with collate enabled, SavaPage prints pages 1,2,3,4 and 5 in that order and then repeats. However, if collate is disabled and you print three copies of those same five pages, SavaPage prints pages in this order: 111, 222, 333, 444, and then 555. The icons in the checkbox are a mnemonic of the output when the collate option is enabled or disabled.

- See [Section K.1.1.6, “sheet-collate” \[293\]](#) on how to specify the collate option as **PPD Extension**.

When the **Print documents separately** option is checked, each acquired document is printed as a separate job. In this way finishings (like stapling) can be applied per printed document.

- This option is *not* shown when the acquired documents are edited (pages have been deleted or rearranged) , or the **Document** scope is restricted to a single document .
- When this option is *not* checked, one (1) CUPS job is created. To separate documents on paper output, in case print is in duplex, blank filler pages are inserted in between, so the first page of a next document is on the front page of a new sheet. The same strategy is applied for single-sided n-up print jobs.

When the option **Delete pages after printing** is checked, input documents or pages are deleted after the printing command is issued, and an extra option is displayed to select the deletion scope. The option is automatically reset after printing.

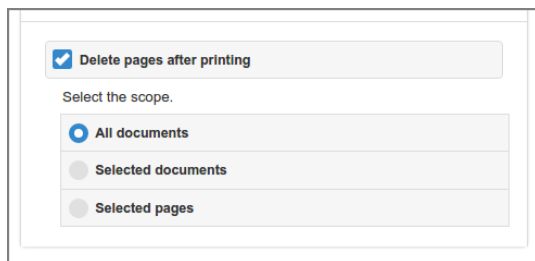


Figure 3.31. User Web App: Print - Delete Pages

- **All documents:** all input documents are deleted.
- **Selected documents:** documents for which pages were printed are deleted.
- **Selected pages:** all pages selected for printing are deleted.

The “Delete pages after printing” option can be preset and disabled for editing in [Section 4.10.10, “Proxy Print” \[141\]](#). In that case a fixed text, denoting the action, is (optionally) displayed instead of the checkbox. Press the **Print & Close** button to print and close the dialog. The job is printed right away, *unless* it is a Job Ticket or when **NFC Authentication** is configured for the printer. In the later case the user must authenticate with an NFC card swipe to release the print job.



Note

In rare situations, when the host system is under heavy load, and connecting to CUPS fails, a warning message is displayed saying “The print service is currently unavailable or too busy. Please try again later.”

In the **Hold Release** scenario the job is held so it can be released by the user at a later time without using the User Web App. In the **Direct Print Release** scenario described below the user is prompted to authenticate immediately.

Pending Job Tickets and Hold Print Jobs can be inspected and removed when needed: see [Section 3.2.2.2, “Hold Print Jobs” \[30\]](#).



Warning

Job Tickets allow unrestricted printing, but printing to “regular” printers may be denied when the number of job pages exceeds a maximum. See the [Proxy Print Options](#).

The following configuration properties apply:

Configuration property	Description
proxy-print.remove-graphics.enable	Set to Y or N (default), to show/hide the Remove graphics option.
webapp.user.proxy-print.clear-printer	If Y, the selected printer is cleared after each proxy print. If N (default), the selected printer and options are preserved, and will be used for the next print job.
webapp.user.proxy-print.clear-delegate	If Y, Delegated Print data is cleared after each proxy print. If N (default), this data is preserved, and will be used for the next print job.
webapp.user.proxy-print.separate	The default Print documents separately option value, when All Documents are selected. If Y, a separate print job is created for each vanilla inbox document. If N (default), one (1) print job is printed of the complete vanilla inbox content.
webapp.user.proxy-print.separate.enable	Enable the Print documents separately option. If Y, the option is enabled (shown), if N (default) it is not.

Table 3.1. Print Job Settings Configuration Properties

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to set these items.

3.4.5. Direct Print Release

When a print job is issued for a printer secured with **Direct Print Release**, a dialog is shown prompting the user to swipe his card to release the print job.

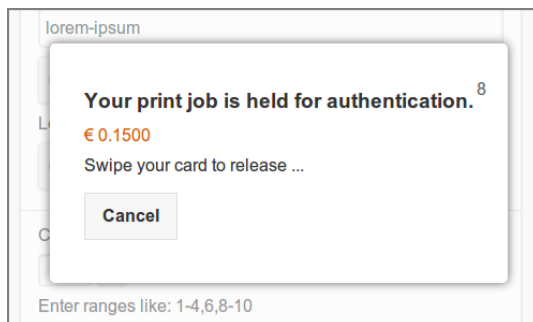


Figure 3.32. User Web App: Printer - Direct Print Release

- The cost of the print job is shown in orange.
- A countdown of the remaining seconds for the card swipe is shown in the top right corner of the pop-up. The time limit (seconds) is contained in configuration key `proxy-print.direct-expiry-secs`. See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change this value.



Note

Since the card reader will be mounted near the printer this implements a *secure pull-print* scenario.

3.4.6. Full Print Scope and Jobs

When a user did *not rearrange or delete* any SafePages and *full* scope is selected without a range of **Pages**, each input job is printed as a separate job when *duplex* printing is selected. As a result the first page of new input job will always start on a new sheet. When the **Title** is left blank, the titles of the print jobs will correspond to the titles of the input jobs. When a **Title** is specified it will be used for *all* print jobs.

When a user *did rearrange or delete* any SafePages the scope is confined to *full* scope and SafePages will always be printed as a single job. When the **Title** is left blank the print job title will be generated according to the timestamp format SavaPage-CCYY-MM-DDTHH:MM:SS.

3.4.7. Delegated Print Edit

This dialog is part of the **Invoicing** track, and is started from **Printer Settings** or **Print Job Settings**. It is used to edit Delegated Print.

In a Delegated Print scenario the user prints as *delegate* on behalf of other users, known as *delegators*. The result is a single job with multiple copies of a document. The total cost of the job is *pro rata* charged to the account of each delegator.

*The **Copies** section at the bottom of the dialog summarizes the selected delegators from the **Invoicing** section at the top.*

















Copies			
Selection	Members	Copies	Invoicing
<input checked="" type="checkbox"/>  Klas.1A1	3	6	 Klas.1A1
<input type="checkbox"/>  Klas.1B1	2	2	 Personal account
<input checked="" type="checkbox"/>  Stella Timmer	1	1	 Personal account
<input type="checkbox"/>  datraverse	4	8	 ICT Department (Staff)
<input type="checkbox"/> 		1	 Marketing (Staff)
<div> <input type="button" value="Select all"/> <input type="button" value="Deselect all"/> <input type="button" value="Remove"/> </div>			
<div> <input type="button" value="Back"/> <input type="button" value="Next"/> </div>			

Figure 3.33. User Web App: Print Delegation Copies

- **Selection** type is marked with one of these icons:
 -  : User Group.
 -  : Individual User.
 -  : Shared Account.
- Each selection line shows the number of **Members** and **Copies**. The latter is the multiplication of members and the number of copies entered in the **Invoicing** section.
 - All *User Group* members are included, *whatever their role*.
 - *Individual Users* of course have just one member and have **Print Job Delegator** role.
 - *Shared Account* counts unspecified users. Just the number of copies is shown.
- The **Account** column shows where printing costs are charged to:
 -  : Group Account.

-  : Personal Account of Group **Members**.
-  : Shared Account.

The screenshots from the **Invoicing** section, presented in the sections below, explain how the lines in the **Copies** section were created.

Remember that for items to appear on the various search list:

- **Groups** must have **Print Job Delegator** role *enabled*.
- **Users** must have **Print Job Delegator** role *enabled*.
- **Accounts** must be accessible by the Delegate as member of one of the User Groups set in **Account Access Control**.
 - When a Delegate has access to a single Shared Account only, the Account search list is **not** displayed, and the single account is implicitly selected.

Availability of the option to enter multiple number of copies is dependent on configuration property `proxy-print.delegate.multiple-member-copies.enable`.

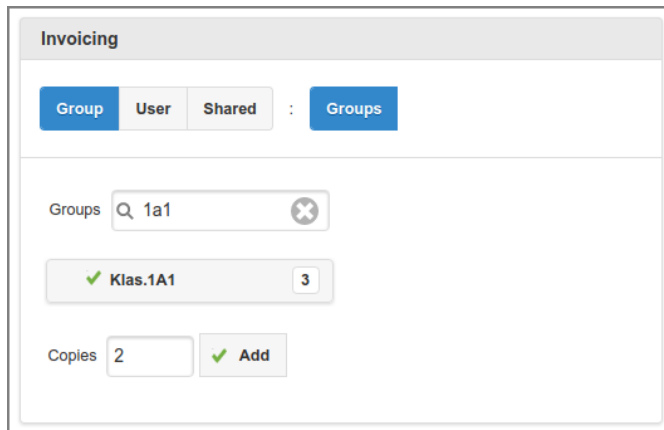


Tip

Delegated Print can be integrated with PaperCut. See [Section N.1, “Delegated Print to PaperCut” \[315\]](#).

3.4.7.1. Group Invoicing

Select **Group** to invoice copies for **Groups** on Group Account.



The screenshot shows the 'Invoicing' section of the User Web App. At the top, there are three tabs: 'Group', 'User', and 'Shared', with a colon and a 'Groups' button to the right. Below the tabs, there is a search field labeled 'Groups' with a magnifying glass icon and a close button (X). The search results show a single entry: '✓ Klas.1A1' with a '3' in a box next to it. At the bottom, there is a 'Copies' field with the value '2' and an 'Add' button with a green checkmark.

Figure 3.34. User Web App: Delegated Print - Group Invoicing

Availability of this option is dependent on configuration property `proxy-print.delegate.account.group.enable`.

- Enter a part of the group name in the **Groups** quick search field and the first chunk with matching groups will show below. Paging buttons are shown when more chunks are available.
- Select one or more **Groups** from the list by tapping them (a green check mark sign will show at the front).
- Enter the number of **Copies** to be printed for each group member.
- Press the **Add** button to add the groups as delegator.
- The Group Account will be invoiced.

3.4.7.2. User Invoicing for Groups

Select **User** to invoice copies for **Groups** on User Account.

Figure 3.35. User Web App: Delegated Print - User Invoicing (Groups)

- Select one or more **Groups** from the list.
- Enter the number of **Copies** to be printed for each group member.
- Press the **Add** button to add the group members as delegator.
- *The Personal Account of each individual group member will be invoiced.*

3.4.7.3. User Invoicing for Users

Select **User** to invoice copies for Individual **Users** on User Account.

Figure 3.36. User Web App: Delegated Print - Personal Invoicing (Users)

- Select one of the **Groups** to narrow down the search scope. When *no* group is selected the **All Users** built-in group is implied.
- Search and select one or more **Users** from the list.
- Enter the number of **Copies** to be printed for the selected users, and press the **Add** button to add them as delegator.
- *The Personal Account of the selected users will be invoiced.*

3.4.7.4. Shared Invoicing for Groups

Select **Shared** to invoice copies for **Groups** on Shared **Account**.

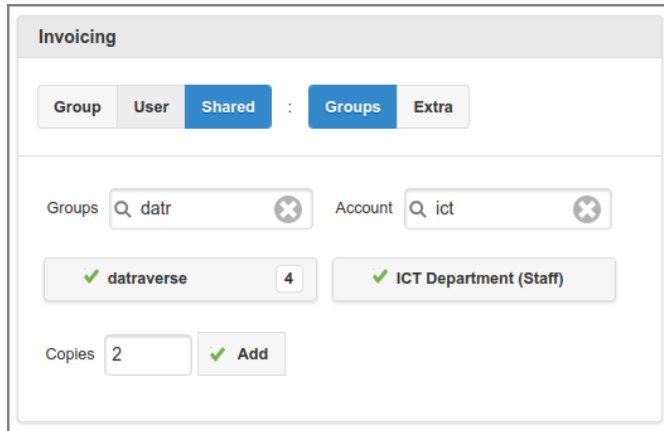


Figure 3.37. User Web App: Delegated Print - Shared Account Invoicing (Groups)

Availability of this option is dependent on configuration properties `proxy-print.delegate.account.shared.enable` and `proxy-print.delegate.account.shared.group.enable`.

- Select one or more **Groups** from the list.
- Select a Shared **Account**.
- Enter the number of **Copies** to be printed, and press the **Add** button.
- *The Shared Account will be invoiced.*

3.4.7.5. Shared Invoicing for Extra Copies

Select **Shared** to invoice Extra copies on Shared **Account**.

As this is the last Invoicing Selection of our example, the complete Delegated Print dialog is shown in the screenshot.

Figure 3.38. User Web App: Print - Delegated Print - Shared Account Invoicing (Extra)

Availability of this option is dependent on configuration property `proxy-print.delegate.account.shared.enable`.

- Select a Shared **Account**.
- Enter the number of **Copies** to be printed, and press the **Add** button.
- *The Shared Account will be invoiced.*

3.4.7.6. Configuration Properties

The following Delegated Print configuration properties apply:


Configuration property	Description
<code>proxy-print.delegate.account.group.enable</code>	Set to Y (default) or N, to enable/disable invoicing with Group Account.
<code>proxy-print.delegate.account.shared.enable</code>	Set to Y (default) or N, to enable/disable invoicing with Shared Account.
<code>proxy-print.delegate.account.shared.group.enable</code>	Set to Y (default) or N, to enable/disable invoicing with Shared Account for Groups copies.
<code>proxy-print.delegate.multiple-member-copies.enable</code>	Set to Y (default) or N, to enable/disable multiple copies for Groups and Users. When disabled, the number of copies is 1.

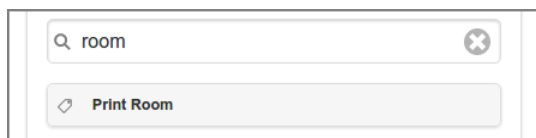
Configuration property	Description
webapp.user.proxy-print.delegator.name-id	Set to Y (default) or N, to append the User ID (or not) to the user name in the Delegated Print User Search List. Set to N when the User ID is already part of the user name.
webapp.user.proxy-print.delegate-copies-apply-switch	Set to Y (default) or N, to show/hide an On/Off switch to apply Delegated Print Copies in the User Web App Print Dialog. When N, Delegated Print print Invoicing is active when the sum of collected delegator Copies is greater then zero.

Table 3.2. Delegated Print Configuration Properties

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to set these items.

3.4.8. Job Ticket Print

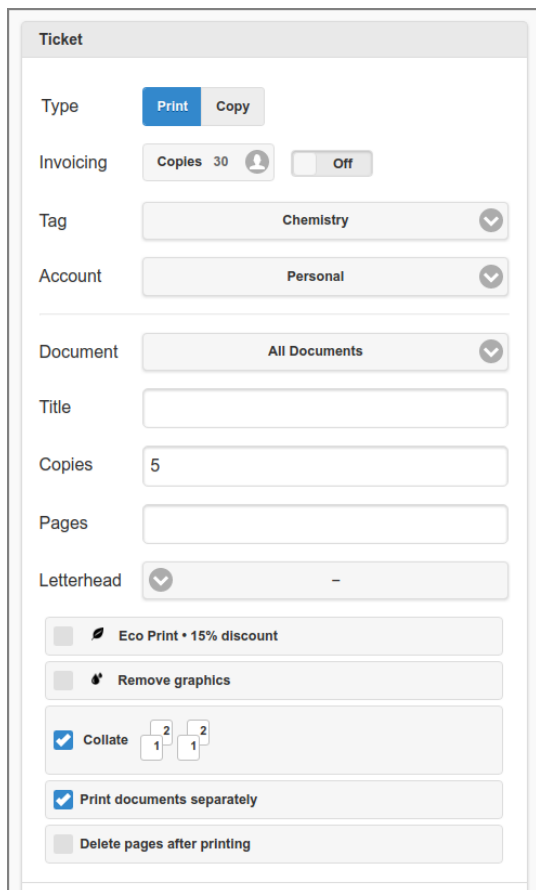
When **Job Ticket Printer** instances are present, users with role **Job Ticket Creator** can select the  marked printers.



The screenshot shows a search bar with the text 'room' and a close button (X). Below the search bar is a button labeled 'Print Room' with a diamond icon on the left.

Figure 3.39. User Web App: Print - Select Job Ticket Printer

By default, Job Tickets are of Type **Print**.



The screenshot shows the 'Ticket' settings dialog for 'Print'. It includes fields for 'Type' (Print/Copy), 'Invoicing' (Copies 30, Off), 'Tag' (Chemistry), 'Account' (Personal), 'Document' (All Documents), 'Title', 'Copies' (5), 'Pages', and 'Letterhead'. There are also checkboxes for 'Eco Print • 15% discount', 'Remove graphics', 'Collate' (checked), 'Print documents separately' (checked), and 'Delete pages after printing'.

Figure 3.40. User Web App: Print - Job Ticket Settings - Print

All options for a regular print job apply. The **Tag** option is unique for Job Tickets: see the `jobticket.tags.*` configuration properties at the end of this section.



Note

When the **Title** of the Ticket is *not* specified, SavaPage will compose one, based on the **Document** title(s). In this way extra identifying data will be available.

When configured as such, Job Tickets can also be used for **Copy** requests. All printer settings of a regular proxy print apply, as well as some job options. However, SafePages are not needed, and no PDF document will be attached to the ticket.

The screenshot shows a 'Ticket' configuration window. At the top, there are two tabs: 'Print' and 'Copy', with 'Copy' being the active tab. Below the tabs, there are several settings:

- Invoicing**: A section with 'Copies' set to 30, a user icon, and a toggle switch labeled 'Off'.
- Tag**: A dropdown menu currently showing 'Chemistry'.
- Account**: A dropdown menu currently showing 'Personal'.
- Title**: An empty text input field.
- Copies**: A text input field containing the number '5'.
- Pages**: A text input field containing the number '1'.
- Collate**: A checked checkbox followed by a sequence of page numbers: 1, 2, 1, 2.

Figure 3.41. User Web App: Print - Job Ticket Settings - Copy

Users can enter a **Date and time of delivery** and **Remarks**.

The screenshot shows a form section for 'Date and time of delivery'. It includes three input fields: a date field with '2018-02-23', a time field with '10', and a minute field with '30'. Each field has a clear (X) button. Below these is a 'Remarks' section with a text area containing the text 'Yellow paper'. At the bottom of the form, there are two buttons: 'Back' (with a home icon) and 'Send' (with a checkmark icon).

Figure 3.42. User Web App: Print - Job Ticket Settings

After the ticket is sent, the issued Ticket Number is displayed, and the user can view the Job Ticket on his **Hold Print Jobs** list.

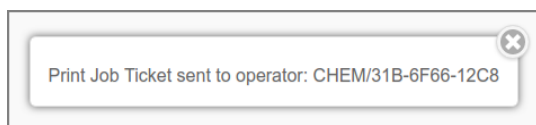


Figure 3.43. User Web App: Print - Job Ticket - Sent

In case of a Copy Job, the user will add a note with the Ticket Number to the hard copy original, before handing it to the Job Ticket operator, who will use it for off-the-glass copying according to the specs in the job ticket.

A Job Ticket is printed on a central queue and is handled and released by users with role **Print Job Operator** in a special Web App. See [Chapter 5, Job Tickets Web App](#) [174]. When the Job Ticket is printed or canceled, the user is optionally notified by email.

The following configuration properties apply:

3.4.8.1. Configuration Properties

Configuration property	Description
jobticket.delivery-datetime.enable	Set to Y (default) or N, to show/hide the Date and time of delivery option. When date/time is hidden its value is set by the system to the date/time of the submit.
jobticket.copier.enable	Set to Y or N (default), to enable/disable the creation of a Copy Job Ticket.
jobticket.notify-email.completed.enable	See Section 5.3, “Ticket Configuration Properties” [181].
jobticket.notify-email.canceled.enable	See Section 5.3, “Ticket Configuration Properties” [181].
jobticket.tags	A comma separated list of Job Ticket tags to be applied as job ticket number prefix. Each tag on the list is formatted as "TAG/word", where "TAG" is a unique N-letter upper-case mnemonic, "/" is a fixed separator, and "word" is a case-sensitive single word used in UI context. E.g. "MATH/Maths, PHYS/Physics, CHEM/Chemistry". When "CHEM" tag is applied, a generated ticket number looks like "CHEM/EE1-FA3E-6596".
jobticket.tags.enable	Set to Y or N (default) to enable <code>jobticket.tags</code> .
jobticket.tags.required	Set to Y (default) or N, to make usage of a Job Ticket Tag compulsory or optional.

Table 3.3. Job Ticket Print Configuration Properties

See [Section 4.10.14.10, “Config Editor”](#) [155] on how to set these items.

3.5. Printer Settings

Ricoh Aficio MP C7500 PS

A4 A3 1

Default

Landscape

Media Source

Automatic

Double-sided printing

Two-sided (long edge)

Color Mode

Color

Resolution

600x600 DPI

Rotate by 180 degrees

Off

Pages per side

1

Back Invoicing Job

Figure 3.44. User Web App: Printer - Settings

Set one or more printing options by pushing the pick-list buttons. The options are initialized with the CUPS printer defaults at the start of a user session. Changes made in this dialog are held during a user session, unless they are cleared after a proxy print: see [Table 3.1, “Print Job Settings Configuration Properties”](#) [45].

Options are printer specific, and are automatically identified by SavaPage. See [Section 3.5.3, “Printer Setting Options”](#) [57].

A preview of the Finished Page is shown at the top. A checkbox shows the orientation (Portrait, Landscape) of the first page of the selected document or page range. See [Section 3.5.2, “Print Preview”](#) [56] for more information.

The **Invoicing** button is visible when [Proxy Print Delegation](#) is enabled and the user has [Print Job Delegate](#) role. Pressing it opens the [Print Delegation Dialog](#).

Pressing the **Job** button opens the [Print Job Settings](#).



Note

Options are validated with [SPConstraint](#) rules, when leaving the Printer Settings dialog with **Invoicing** or **Job** button.

If options present in the CUPS PPD file are missing you can add them, as explained in [Appendix K, PPD Extensions](#) [292]. In this way you can add finishing options:

Figure 3.45. User Web App: Printer - Settings - Finishings

Or **Job Ticket** Options:

Figure 3.46. User Web App: Printer - Settings - Job Ticket

When all paper sizes of SafePages jobs are supported by the printer, each paper size is indicated in green at the top of the dialog, and the **Media Source** defaults to **Automatic**. In this example A4 and A3 are supported. See [Section 4.8.2.3, “Media Sources” \[106\]](#) on how media size is assigned to a printer's media sources.

- When a user selects Media Source **Automatic**, and the target printer supports automatic media source selection, SavaPage will use IPP `media-source` attribute value `auto` when sending the job to the printer. In this way, even when the job is redirected to a another (compatible) printer, the right media source will be selected automatically, based on media size.



Tip

If you always want to force a print job to automatic media source selection, irrespective from the **Media Source** choice, you can use a **PPD Extension** to map all `media-source` values to the same PPD `auto` value.

3.5.1. Page Scaling

When you choose a specific Media Source (holding a specific paper size) that does *not* match all paper sizes of SafePages jobs, a **Page Scaling** option will appear with an extra page size indicator (with light orange background) at the top of the dialog.

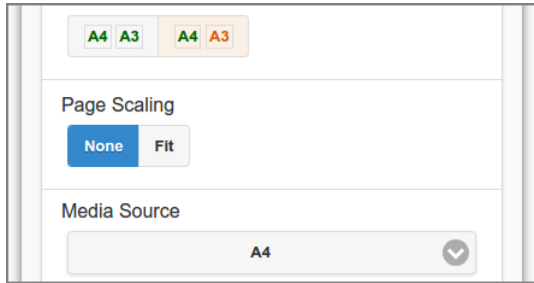


Figure 3.47. User Web App: Print - Page Scaling (None)

For example, as the orange A3 indicator shows, the selection of the A4 media source does not fit available A3 page sizes.

The **None** option (indicated as solid white square) will scale the deviant pages according to the print-scaling option “none” in the PPD Extension file.

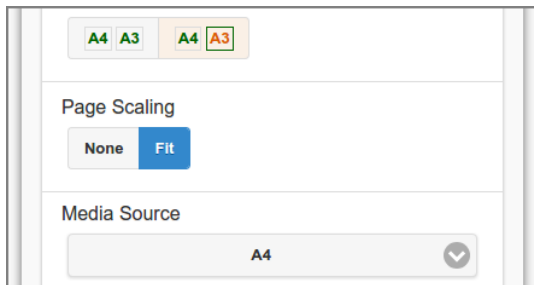


Figure 3.48. User Web App: Print - Page Scaling (Fit)

The **Fit** option (indicated as solid square) will scale the deviant pages according to the print-scaling option “fit” in the PPD Extension file.

When the print-scaling option is not defined in the PPD Extension file, fit will be the only available option, and deviant pages will always be scaled up or down to fit the target media source.

See [Section K.1.1.5, “print-scaling” \[293\]](#).

3.5.2. Print Preview

Print Preview helps the user to envision the effect of **Pages per side**, **Rotate by 180 degrees**, **Staple** and **Punch** on the Finished-Page.

A checkbox at the top right shows the orientation (Portrait, Landscape) of the first page of the selected document or page range. This is the right default in most cases, but the value can be toggled if the user knows for sure that the first page has a different orientation. For example, because he selects a specific page range from SafePages document(s) that have mixed page orientation. Whatever the choice is, SavaPage always checks if the assumption was right. When the actual print orientation turns out different as the one selected in the preview, and orientation dependent options, like staple and punch, were chosen, the user will be prompted to adapt the print settings and try again.



Figure 3.49. User Web App: Print - Preview (4-up, landscape, staple, punch)



Note

Print Preview can be enabled/disabled by setting `webapp.number-up-preview.enable` to Y (default) or N.

3.5.3. Printer Setting Options

The options presented in the Printer Settings dialog are a collection from:

- Basic IPP Printer Attributes as shown in the table below.
- `org.savapage.int-page-rotate180`
- Internal IPP - PPD Mapping Extensions.
- Internal IPP Job Ticket Extensions.

Attribute	Value source
media	CUPS (a subset of common media sizes is used).
media-source	CUPS or PPD Extension.
media-type	PPD Extension.
number-up	CUPS (values 1, 2, 4, 6, 9).
output-bin	CUPS or PPD Extension.
print-color-mode	CUPS or PPD Extension.
printer-resolution	CUPS.
sides	CUPS or PPD Extension.

Table 3.4. Basic IPP Printer Attributes



Important

User input is validated with `SPConstraint` rules.

3.6. Letterheads

A *tap* on the `Letterhead` button in the main SafePages view shows the Letterhead dialog. See [Section 3.2](#), “SafePages” [26].

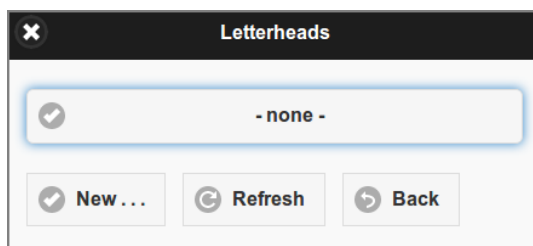


Figure 3.50. User Web App: Letterheads

- Press the top button to get a pop-up list of letterheads. When a letterhead is selected from the list it can be previewed and edited. The selected letterhead acts as default in the Print and PDF dialog: see [Figure 3.26](#), “User Web App: PDF - Letterhead” [40] and [Figure 3.28](#), “User Web App: Print - Select Printer” [41].

- Press the **Refresh** button to rebuild the list if you suspect public letterheads were added or removed.
- Press the **New** button to create a new letterhead from the current SafePages. All the SafePages are used for the letterhead. [Figure 3.51, “User Web App: Letterhead - New” \[58\]](#) is an example of a fresh created letterhead.



Note

Depending on [User Privileges](#) buttons and sections for creating and editing letterheads might not be shown.



Note

If the SafePages contain DRM-restricted content the **New** button is not visible.

Figure 3.51. User Web App: Letterhead - New

- If the letterhead document consists of one page, this page is applied to every page of the document. The letterhead page is scaled and rotated as needed to fit the input page.
- If the letterhead document has more than one page, each page of the letterhead is applied to the corresponding page of the output document. If the output document has more pages than the letterhead, then the final letterhead page is repeated across these remaining pages of the output document.
- Letterheads can be applied as foreground or background.
- Users who are administrator can mark letterheads as *public* by ticking the "Public" checkbox. Public letterheads are available for all users, but can be edited and deleted by administrators only.
- The title of a private letterhead can be edited. The title of a public letterhead can be edited by an administrator only.
- Tap on a letterhead thumbnail to get a detailed pop-up image. See [Figure 3.52, “User Web App: Letterhead - Detail” \[59\]](#).
- Press the **Apply** button to commit changes or push the **Delete** button to remove the letterhead.



Note

For a background letterhead to be effective, SafePages should be transparent. HTML pages printed from browsers like Internet Explorer and Firefox might pose a problem here. The white background on HTML pages might act as a solid background.

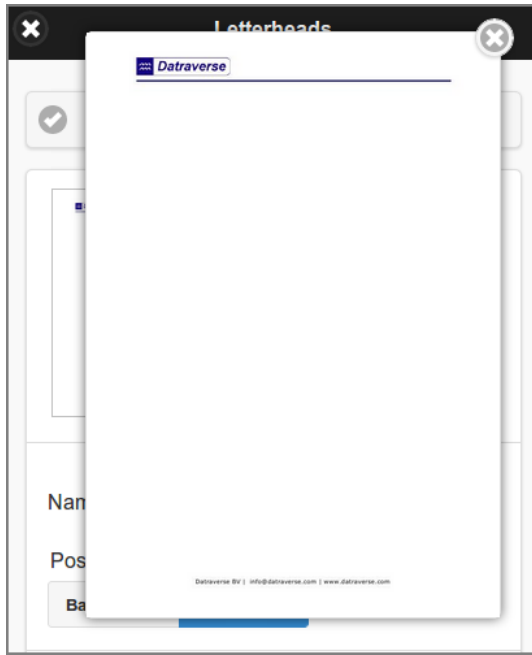


Figure 3.52. User Web App: Letterhead - Detail

- Press the **x** button in the upper right corner, or *tap* outside the pop-up, to close the pop-up image.

3.7. Delete

A *tap* on the **Delete** button in the main SafePages view shows the Delete dialog. See [Section 3.2, “SafePages” \[26\]](#).

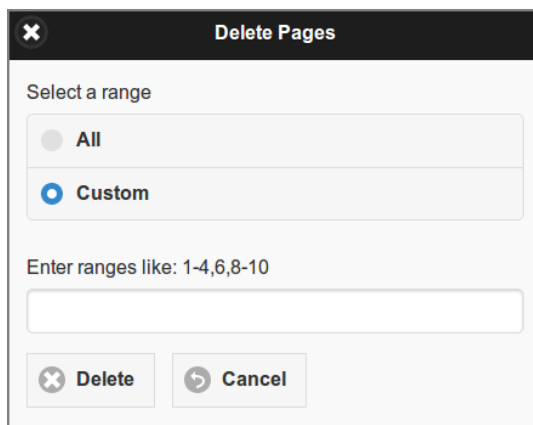


Figure 3.53. User Web App: Delete SafePages

- Select a range of SafePages to delete. Press the **All** button to select all pages. Or, push the **Custom** button to enter a custom range of SafePages: the value can be a single page, a range of pages, or a collection of page numbers and ranges separated by commas.

- Tap the **Delete** button to perform the delete action.



Caution

After SafePages are deleted any **Fast Print** will fail and all SafePages will be cleared as a result. Please use a **Hold Print** instead.



Tip

You can delete all SafePages of a SavaPage print job in the Document Details dialog, as described in [Section 3.2.3.1, “Delete and Undo” \[32\]](#).

3.8. Log

A tap on the **Log** button in the main SafePages view shows the Document Log. See [Section 3.2, “SafePages” \[26\]](#).

3.8.1. Documents

The Document Log shows all documents the user printed to SavaPage, and which were subsequently exported or Proxy-Printed. For a detailed description of this screen see [Section 4.11, “Documents” \[157\]](#) in the Admin Web App chapter.

Some Document **Type** selections are *hidden*, depending on user **role**, **privileges** and resource (Queue, Printer) availability.

- Type **PDF** is hidden when user has no privileges to Send and Download SafePages.
- Type **Ticket** is hidden when user is not a Job Ticket Creator.
- Type **Print** is hidden when user is not a Print Job Creator, or has no access to a printer.

The following configuration property applies:

Configuration property	Description
webapp.user.doclog.select.type.default-order	<p>A comma separated priority list of enum values ALL, IN, OUT, PDF, PRINT, TICKET, that correspond to Document Type selection options. When this list is empty, ALL is assumed.</p> <p>The default Type selection is the first value from the list that corresponds to a visible select option.</p> <p>To comply to EU GDPR ("Right for a data subject to receive personal data concerning them"), IN and OUT types are always available, so full history can be queried regardless of current status.</p>

Table 3.5. Document Type Select: Configuration Property

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to set these items.

The **Letterhead** selection is hidden when user has no Letterhead privileges. The **Queue** selection does not hold disabled or inactive (reserved) queues (/gcp, /mailprint, /webprint). The **Printer** selection holds accessible Printers only (Job Ticket "Printers" are not shown).

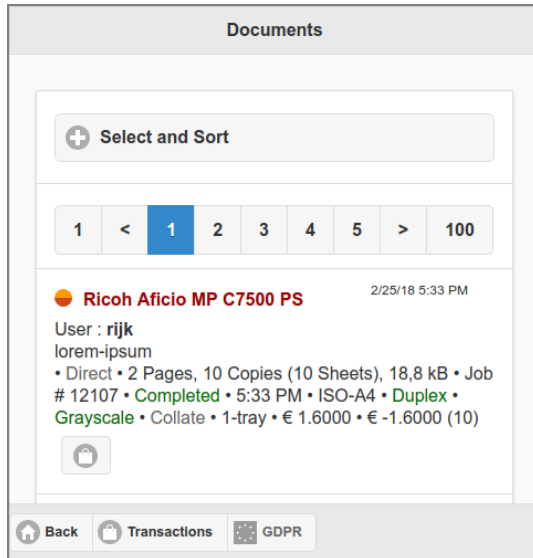


Figure 3.54. User Web App: Log - Documents

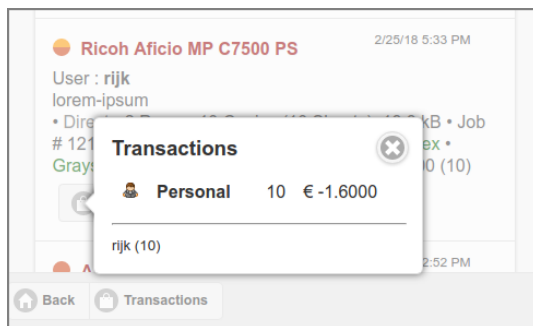


Figure 3.55. User Web App: Log - Document Transactions

Press the transaction button in the document item to see invoicing details. Press the **Transactions** button in the footer bar to view the **Transaction List**, and the **GDPR** button to view the **GDPR Dialog**.



Note

Depending on **User Privileges** the **Transactions** button might not be present.

3.8.2. Transactions

The Transaction Log shows the financial transactions on the user's account.

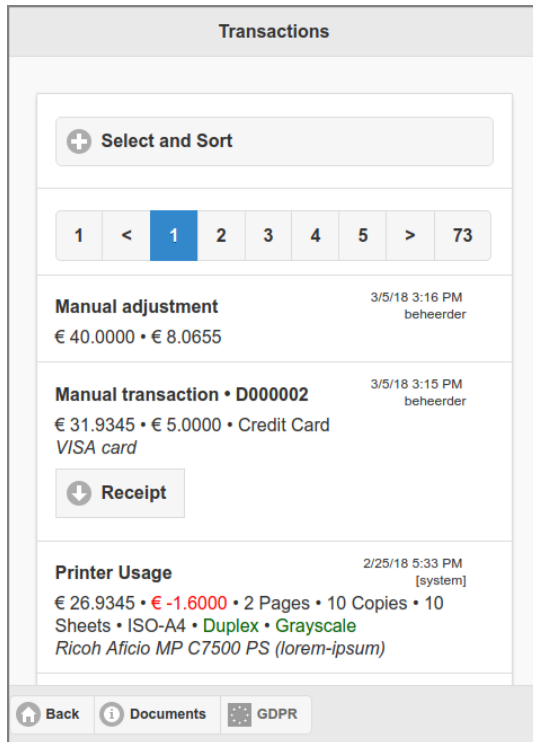


Figure 3.56. User Web App: Log - Transactions

Each entry in the list has the following lines.

- A header with the transaction type.
- The resulting account balance with the transaction amount.
- Extra information is added depending on the transaction type.
- A Manual Transaction denotes the Receipt Number in the header and has a **Receipt** button to download the receipt as PDF.

Some samples of other entries...



 **mistercash**

5/26/15 12:26 PM
[system]

EUR 56.08 • EUR 7.61 • 6787 • tr_GrSpp748AK
SavaPage credit for rijk.
• EUR 8.00 -/- 0.39

Transfer

6/19/15 12:18 PM
rijik

EUR 5.00 • **EUR -5.00**
Transfer from user rijk to user "ellen" - cash betaling

Transfer

6/19/15 12:18 PM
rijik

EUR 4.90 • EUR 5.00
Transfer from user rijk to user "ellen" - cash betaling

 **bitcoin**

6/15/15 4:57 PM
[system]

EUR 4.86 • EUR 0.83 • [96212...82710](#) • B 0.00400000

Bitcoin Payment entries show a hyperlink with a shortened transaction hash. The hyperlink opens the transaction details in a new browser tab. The hyperlink URL is held in configuration property `financial.bitcoin.user-page.url-pattern.trx` and can be changed with the [Configuration Editor](#). The value must contain the `{0}` placeholder for the transaction hash. Sample values are `https://blockchain.info/tx-index/{0}` and `https://blockexplorer.com/tx/{0}`.

Push the **Documents** button in the footer bar to view the [Document List](#), and the **GDPR** button to view the [GDPR Dialog](#).

Figure 3.57. User Web App: Log - Transactions

Transactions can be filtered and sorted as follows:

- **Comment containing text:** selects transactions with comments containing a text snippet.
- **Type:** selects – (all) or a single one of the transaction types...
 - **Initial:** Balance allocated when account was created.
 - **Adjustment:** Manual adjustment by an administrator. See [Section 4.4.4.6, “Financial” \[87\]](#).
 - **Deposit:** Adjustment of balance at a [Point-of-Sale](#).
 - **External:** Increment of balance by transferring funds from an external account. See [Section 3.10.6, “Transfer Money” \[69\]](#).
 - **Transfer:** Increment or decrement of balance by transferring credit to another user. See [Section 3.10.5, “Transfer Credit” \[68\]](#).
 - **Voucher:** Increment of balance by [redeeming a voucher](#).
 - **Queue:** A delegator transaction as part of a [Delegated Print](#) where the document and delegate info is retrieved from an *External Supplier* and printed to a Proxy Printer managed by a *Third Party Print Management System* (TPPMS). The transaction appears as **Queue Usage** item in the list. Print status and cost are retrieved from the TPPMS. The cost will be 0.00 when the status is “Pending (external)”, “Expired” or “Canceled”. When status is “Completed” the cost will be known. Currently External Supplier TPPMS [PaperCut](#) are supported.
 - **Printer:** A transaction for proxy printing. When the Proxy Printer is solely managed by SavaPage the costs are according to the specified [Printer Costs](#). When the Proxy Printer is additionally managed by a *Third Party Print Management System* status and cost are retrieved from that system and displayed just like the previously discussed **Queue Usage** item.
- Select a creation **Period** by entering a **From** and **To** date. Tap the **x** button after a date to clear it. See this example [Data Selection Dialog](#).
- Transactions can be sorted **Ascending** or **Descending** by creation **Date** or **Type**.
- The list is refreshed, and the selection applied, after you push the **Apply** button.
- The **Default** button resets the selection items to their default values.

- The PDF and CSV buttons downloads the User List in their respective formats, using the selection item values.
- The minus icon collapses the Select and Sort section.

3.9. Sort

A **tap** on the **Sort** button in the main SafePages view switches to Sort mode. See [Section 3.2, “SafePages” \[26\]](#).

In sort mode pages can be rearranged and deleted.



Figure 3.58. User Web App: Sort

- This screen shows the result after some editing.
- One *cut* page is shown with a red border.
- Notice the mini scissor icon at the bottom of the screen, showing the page number of the cut page.
- One *selected* page is shown with an orange border.



- The **footer bar** shows a scissor icon with a page number of the cut, and a pin icon with the page number of the selected page.

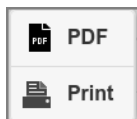
These are the actions that can be performed on page images:

- A **tap** on a *single* page will (un)select it.
- A **tap** on an *aggregated* page will expand it. Aggregated pages are described at [Figure 3.11, “User Web App: SafePages - Aggregated” \[28\]](#).

Here is a summary of the buttons and their function:



More actions : shows a pop-up with more actions (see below).



More actions pop-up : pop-up with more actions. Selected pages are transferred to the **PDF** and **Print** dialog.



Unselect all : unselects all selected pages. Selected pages that are in view are marked with a orange border. The mini pin icon at the bottom of the screen shows all selected pages.



Cut : cuts the selected pages into the clipboard. Cut pages that are in view are marked with a red border. The mini scissor icon at the bottom of the screen shows the cut page ranges in the clipboard.



Undo : reverts all cut actions and empties the clipboard.



Left Paste : pastes the cut pages *before* the first selected page.



Right Paste : pastes the cut pages *after* the first selected page.



Delete : deletes the selected pages.



Inbox : returns to the Main view.



Note

By default a fixed button text is shown on mobile devices only: on desktops a hover text is shown. See [Section 3.2, “SafePages” \[26\]](#) on how to change this behavior.



Caution

After SafePages are sorted any **Fast Print** will fail and all SafePages will be cleared as a result. Please use a **Hold Print** instead.

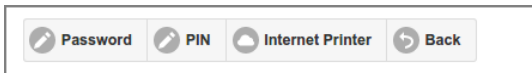
3.10. User Details

This dialog shows the **Pagometers** and **Financial** status of the user, and is shown after a tap on the **User** button in the **footer** of the main panel.



Note

Depending on **User Privileges** the **Financial** section might not be shown.



- For an **Internal User** a **Password** button is shown, when a password has been set (i.e. not erased): see [Section 4.4.4.7, “Password” \[88\]](#). A tap on the button will show the **Password Reset Dialog**.
- When users are allowed to change their PIN a **PIN** button is shown. A tap on the button will show a PIN Reset Dialog. See [Section 4.10.3, “User Authentication” \[126\]](#).
- When an **Internet Print** protocol://authority is present the **Internet Printer** button is shown. A tap on the button will show the **Internet Printer Device URI**.

3.10.1. Internet Printer

Users can copy/paste their private **Internet Print** Device URI from this dialog, and the SAVAPAGE .ppd file can be downloaded.

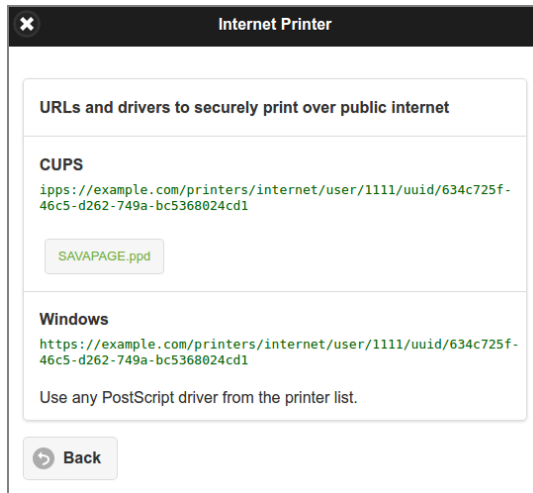


Figure 3.59. User Web App: User Details - Internet Printer Device URI

3.10.2. Pagometers

This section shows the *personal* **Pagometers** of the user, and are analogous to the ones in the Admin Dashboard as shown in Figure 4.8, “Admin Web App: Dashboard - Pagometer” [79] and Figure 4.10, “Admin Web App: Dashboard - Environmental Impact” [80].

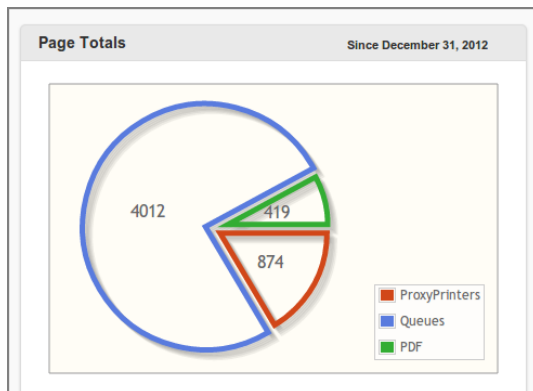


Figure 3.60. User Web App: User Details - pagometer

Environmental Impact	
Paper	900 SU
Trees	10.8%
CO ₂	4.59 kg
Energy	0.011 kW-h

Figure 3.61. User Web App: User Details - Environmental Impact

3.10.3. Financial

This section shows the financial status of the user account and ways to increment the account balance from an external account.



Note

Depending on [User Privileges](#) the section for account transactions might not be shown.

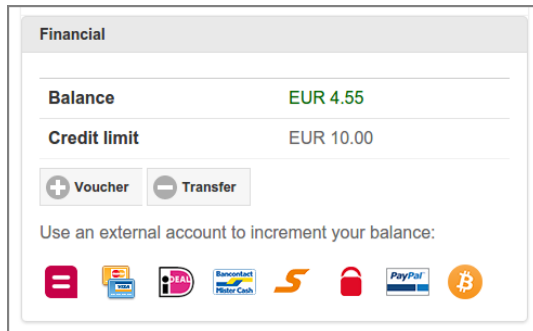


Figure 3.62. User Web App: User Details - Financial

- **Balance:** the user's account balance.
- **Credit limit:** see [Section 4.4.4.6, “Financial”](#) [87].
- A push on the **Voucher** button opens the [Redeem Voucher](#) dialog. The visibility of this button is dependent on an [application setting](#).
- A push on the **Transfer** button opens the [Transfer Credit](#) dialog. The visibility of this button is dependent on an [application setting](#).

When a [Generic](#) and/or [Bitcoin](#) Payment Gateway Plug-in is enabled, an icon is shown for each active payment method. Pushing (clicking) the payment method icon will pop-up the dialog to [Transfer Money](#) or to [Transfer Bitcoins](#).

3.10.4. Redeem Voucher

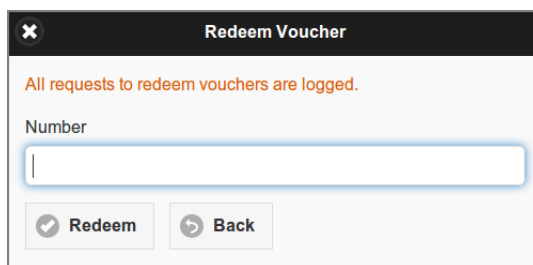
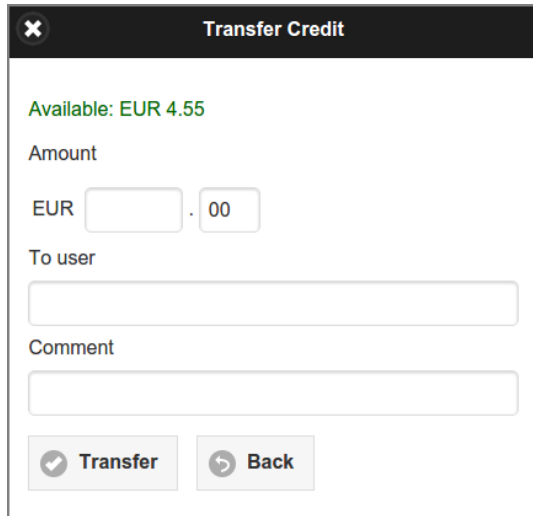


Figure 3.63. User Web App: Redeem Voucher

- Enter the voucher **Number** in the dialog box and press **Redeem**. Make sure to enter the number exactly as listed on the voucher including any dashes (-).
- If you entered the number correctly, the value as shown on the voucher will be transferred to your account and a new entry will list in your [transaction log](#).

3.10.5. Transfer Credit

This dialog is used to transfer funds to another user.



A dialog box titled "Transfer Credit" with a close button (X) in the top left corner. The dialog contains the following elements: a green text label "Available: EUR 4.55"; a label "Amount" above a currency input field showing "EUR" followed by two empty boxes and ". 00"; a label "To user" above a text input field; a label "Comment" above a text input field; and two buttons at the bottom: "Transfer" with a checkmark icon and "Back" with a circular arrow icon.

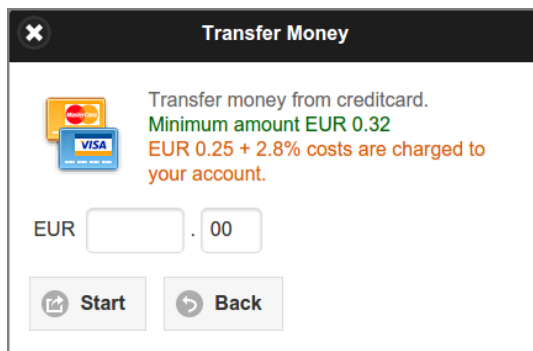
Figure 3.64. User Web App: Transfer Credit

- Enter the **Amount** in currency units and cents. The available amount is shown in green at the top.
- Enter the **To user** id and an optional **Comment**.
- When you press the **Transfer** button the amount will be transferred from your account to the account of the target user. New entries will list in your **transaction log** and the log of the target user.

To configure this dialog see [Section 4.10.12.5, “Transfer Funds” \[147\]](#).

3.10.6. Transfer Money

This dialog is used to transfer money from an external account. The figure below shows a dialog in preparation for a credit card transaction. Other payment methods are available as defined by the active **Generic Payment Gateway Plug-in**. See for example [Section M.1.1.1.1, “Mollie Payment Plug-in” \[312\]](#).



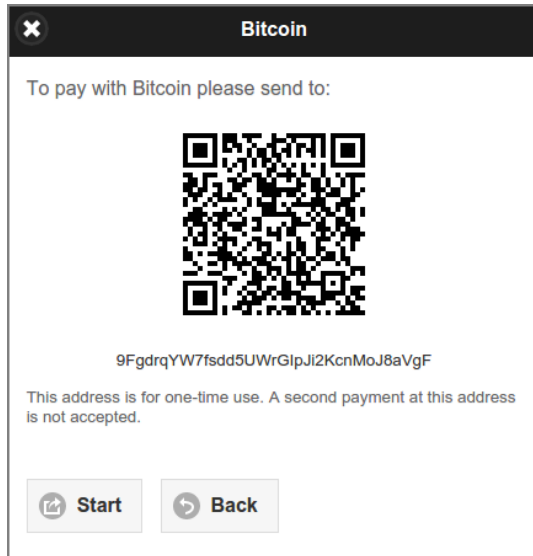
A dialog box titled "Transfer Money" with a close button (X) in the top left corner. The dialog contains the following elements: a Visa logo; a text label "Transfer money from creditcard."; a green text label "Minimum amount EUR 0.32"; an orange text label "EUR 0.25 + 2.8% costs are charged to your account."; a currency input field showing "EUR" followed by two empty boxes and ". 00"; and two buttons at the bottom: "Start" with a right-pointing arrow icon and "Back" with a circular arrow icon.

Figure 3.65. User Web App: Transfer Money from Credit Card

- Enter the **Amount** in currency units and cents.
- Press the **Start** button to start the payment transaction.

3.10.7. Send Bitcoins

This dialog is used as a start to send Bitcoins to the account balance.



Note: the Bitcoin address in the screenshot is intentionally made invalid.

Figure 3.66. User Web App: Send Bitcoins

Start sending Bitcoins by performing one of the following actions:

- Press the **Start** button to automatically open a send transaction in the default Bitcoin wallet on your system.
- Open a send transaction manually in a Bitcoin wallet application on your computer or device (Android, iOS, ...) and either scan the QR code or copy/paste the Bitcoin address (below the QR code).

... and enter the amount to send from your Bitcoin wallet.



Note

The BTC amount is converted to the system currency according to the exchange rate of the Bitcoin service back-end of the **Bitcoin Payment Plug-in**.

3.11. Upload

3.11.1. Upload Dialog

This dialog implement the SavaPage **Web Print** function, and is shown after a tap on the **Upload** button in the **footer** of the main panel.

Figure 3.67. Web Print: Upload File

- **Font for plain text:** change the font when you upload a plain text file (TXT) that contains characters not supported by the **Default** font, like **CJK**. Use **Unifont** when the source text has a real broad Unicode coverage.
- **Choose Files** to be uploaded. Beware that the actual file selection button differs from browser to browser.
- Press the **Upload** button to start the upload.
- The status of the upload will be displayed below the selected file name.
- The **Reset** button clears the status messages and selected file.
- After the upload, use the **Back**, **Print** or **PDF** buttons to navigate to the next step.

Document types can be restricted with the following configuration properties:

Configuration property	Description
web-print.graphics.enable	Enable graphics files for Web Print: Y (default) N
web-print.file-ext.exclude	A comma or space separated list of file extensions (without leading point), that are excluded for Web Print. For example: "rtf,html,ps,txt". When TXT file type is excluded, the Font for plain text selection in Upload File dialog is removed.

Table 3.6. Configuration Properties for Web Print Document Types

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to set these items.



Note

For uploaded file types that do not have a page size defined (HTML, TXT) the **default paper size** is used. For image files the graphic involved is a best fit on the **default paper size**.

3.11.2. Upload Drop Zone

The **Upload Dialog** and thumbnail area of the **main view** can act as **Web Print Drop Zone**. You can (multiple) select files in any desktop application and drag & drop them into the zone, after which they are immediately uploaded. When files are dragged into the zone, it lights up with a green border.

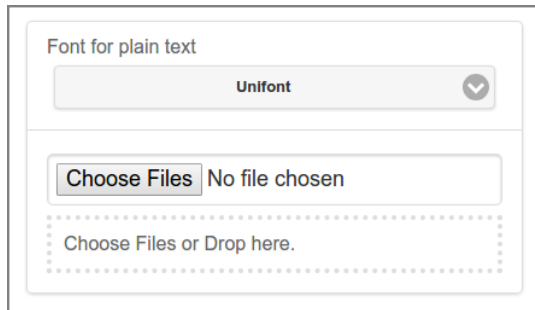


Figure 3.68. Web Print: Drop Zone - Upload Dialog

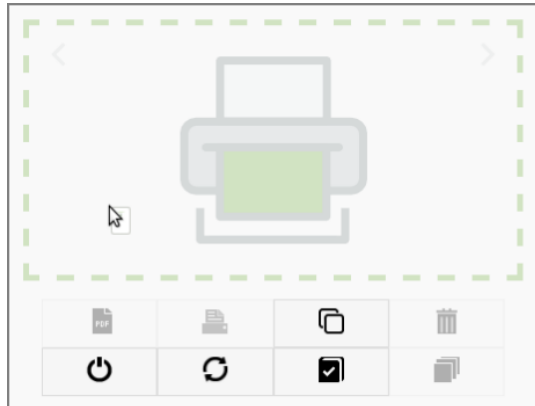


Figure 3.69. Web Print: Drop Zone - Main

The Drop Zone can be enabled at [Section 4.10.8, “Web Print”](#) [139].



Note

For plain text files dropped in the Main Drop Zone, the selected font in the Upload Dialog is used.

3.12. GDPR Dialog

This dialog is shown when the GDPR button is pressed from the **SafePages** top bar action menu, the **Transaction List** or the **Document List**.

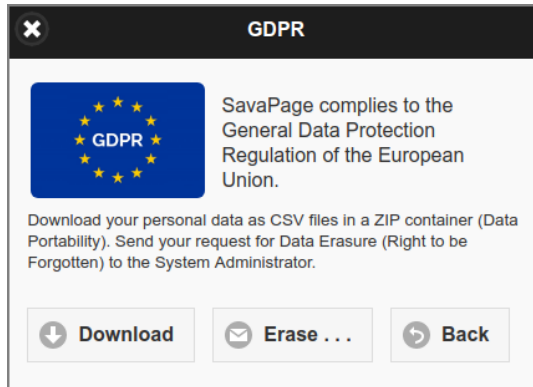


Figure 3.70. User Web App - GDPR Dialog

See [Section 14.2.1, “Data Portability” \[223\]](#) and [Section 14.2.2, “Data Erasure” \[223\]](#).

The following configuration properties apply:

Configuration property	Description
webapp.user.gdpr.enable	Enable the User Web App GDPR Dialog: Y (default) or N.
webapp.user.gdpr.contact.email	Optional email address for GDPR Data Erasure requests.

Table 3.7. Configuration Properties for User Web App GDPR

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to set these items.

Chapter 4. Admin Web App



The Admin Web App can be reached at <https://savapage:8632/admin>. See [Appendix E, URL Cheat Sheet \[278\]](#).

4.1. Login

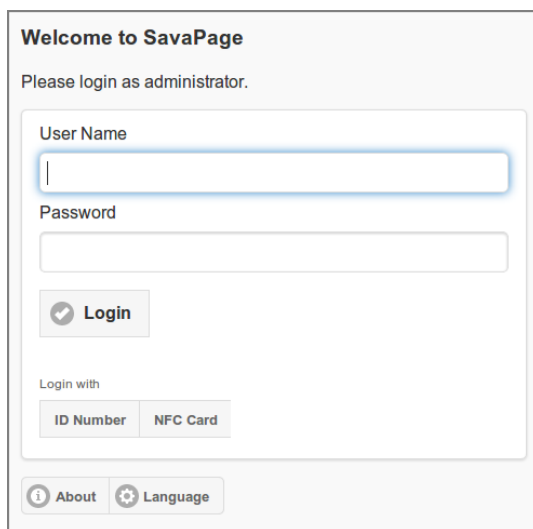
A screenshot of the Admin Web App login screen. The title is "Welcome to SavaPage". Below it, it says "Please login as administrator." There are two input fields: "User Name" and "Password". Below the "User Name" field is a "Login" button with a checkmark icon. Below the "Login" button are two buttons: "ID Number" and "NFC Card". At the bottom, there are two buttons: "About" and "Language".

Figure 4.1. Admin Web App: Login

This login screen is a variant of the User Login screen described at [Section 3.1, “Login” \[22\]](#), with the following exception:

- The internal admin user and Persons with admin rights are allowed to log in. See [Section 4.4.4, “Edit User” \[84\]](#) how to assign admin rights to users.
- After a successful login [Figure 4.2, “Admin Web App: Menu” \[75\]](#) is shown.



Note

Initially, just after installation, only the internal admin user can login. See [Section 4.4.7, “Administrator Role” \[90\]](#).

4.2. Menu

After a successful login this main Admin screen is shown. If this is a first time login, a message will show, telling you that SavaPage needs to be set up and is not ready to use yet. The message will prompt you to go to the Options section and to check the settings. A long as setup is not completed this message will keep appearing after login. When setup is completed, a similar message will appear when the password of the internal admin account still has the default value.

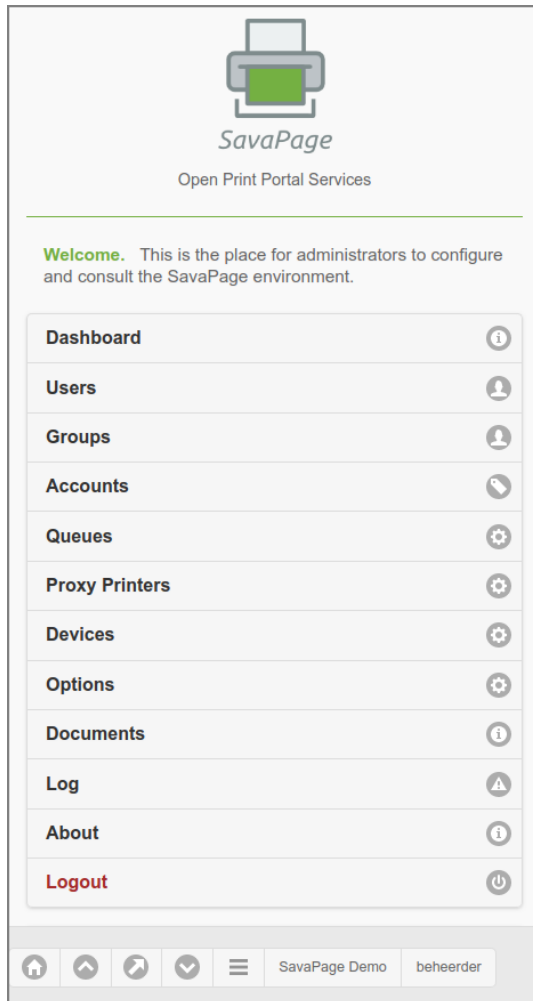


Figure 4.2. Admin Web App: Menu

A *tap* on a the **Logout** button brings you back to the Login screen. A *tap* on any other menu option brings a detailed screen into view. Please see the sections below for a description of each menu option:

- Section 4.3, “Dashboard” [76].
- Section 4.4, “Users” [81].
- Section 4.5, “Groups” [90].
- Section 4.6, “Accounts” [96].
- Section 4.7, “Queues” [99].
- Section 4.8, “Proxy Printers” [101].
- Section 4.9, “Devices” [110].
- Section 4.10, “Options” [120].
- Section 4.11, “Documents” [157].
- Section 4.12, “Log” [161].
- Section 4.13, “About” [163].

These are the buttons in the footer and their function:



Navigate to the top of the page.



Navigate to the top of the menu.



Navigate to the top of the detail panel.



Navigate to the bottom of the page.



Show pop-up menu with additional actions as shown in the figure below.

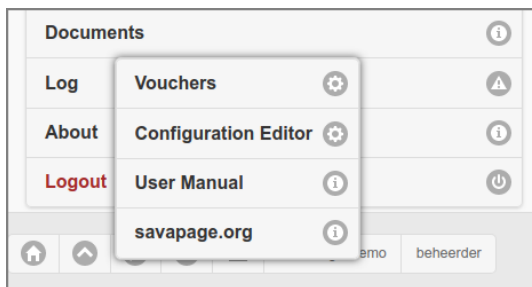


Figure 4.3. Admin Web App: Action Pop-up Menu

Please see the sections below for a description of menu options:

- [Section 4.14, “Vouchers” \[168\]](#).
- [Section 4.10.14.10, “Config Editor” \[155\]](#).



Note

Due to [Admin Privileges](#) certain menu options might *not* be visible.

The User Manual and savapage.org menu items open in a new browser tab.

At the leftmost of the footer is a button with the Community member name. When pushed it opens the [About](#) dialog. A label with the logged-on user id closes the ranks.

4.3. Dashboard

After a *tap* on the **Dashboard** button in the main menu this panel is shown. See [Section 4.2, “Menu” \[74\]](#).



Note

The Dashboard section is automatically refreshed every 60 seconds.

4.3.1. Status

Dashboard	
Community member	SavaPage Demo
Membership	Cardholder
Participants	500
Valid till	February 11, 2019
Days remaining	321
Mode	Production
Status	Ready to use
Uptime	21m
Users	36
Client sessions	0
Web sessions	1
Recent errors	0
Recent warnings	0
SSL valid till	Apr 26, 2018

Figure 4.4. Admin Web App: Dashboard - Status

The head section displays indicators about **Community Membership** status and application runtime:

- *Community member*: The name of the community member organization (empty when no Member Card was imported).
- *Membership* :
 - *Cardholder* : a community member.
 - *Visitor* : a visitor of the community.
 - *Exceeded* : the number of users in the database exceeds the number of Member Card participants.
 - *Expired* : the Member Card reached end-date.
 - *Visitor Expired* : the visitor period expired.
 - *Visitor Edition* : a permanent visitor with 5 users or less in the database.
 - *Invalid version* : the Member Card is incompatible with this SavaPage version.
 - *Invalid* : the Member Card is incompatible with this community.
- *Participants* : the number of community participants.
- *Valid till* : expiration date of Member Card.
- *Days remaining* : number of days till Member Card expires.
- *Mode* : the system mode. Press the button to change the mode to Maintenance or Production.

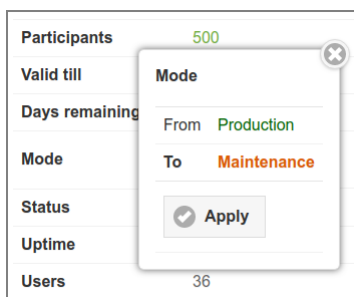


Figure 4.5. Admin Web App: Dashboard - Change System Mode

When system is in *Maintenance Mode*:

- Access to all Web Apps is restricted to users with **Administrator** role.
- Regular users are shown a message explaining the situation, at the Login dialog or in the authorized session.
- *Status*
 - *Ready to use* : SavaPage can be used without impediments.
 - *Setup is needed* : There are one or more options that need to be set up. Access to the User Web App is denied till setup is finished. In the Admin Web App, editing of user details and adding internal users, user groups and shared accounts will not be available.
 - *Not available* : Database access is blocked because a batch job is in progress (database backup, user synchronization, database cleanup). To prevent that user actions requiring database access will block, resulting in long waits, access to the **User Web App** is temporarily denied. Users are informed after a login attempt or, when already logged in, are warned when actions requiring database access are not permitted.
 - *Member Card Missing* : Signals a missing **Community Member Card**.
- *Uptime* : the time the application has been working and available.
- *Users* : the number of users in the database. Deleted users are not part of the total: see [Section 4.4.6, “Deleted Users”](#) [90].
- *Client sessions* : the number of active **User Client** sessions.
- *Web sessions* : the number of active **User Web App** sessions. Multiple sessions on one IP address are counted as one.
- *Recent errors* : the number of errors that occurred in the **Application Log** during the last hour.
- *Recent warnings* : the number of warnings that occurred in the **Application Log** during the last hour.
- *SSL valid till* : when the SSL certificate expires within a year, its expiration date is shown . When expiration is due within 30 days it is shown in orange.

Technical information about the server process can be added to, or removed from, the Dashboard by setting the value of configuration key `webapp.admin.dashboard.show-tech-info` to Y or N. See [Section 4.10.14.10, “Config Editor”](#) [155] on how to change this value. When added, the following extra information appears:

JVM memory	919.1 MB Max • 316.7 MB Total • 224.0 MB Free	?
Threads	62	
Connections	4 (1) • services (database)	
Proxy Print Queue	0	
Job Ticket Queue	0	

Figure 4.6. Admin Web App: Dashboard - Technical Information

- *JVM memory* : “Max” is the maximum memory the JVM will attempt to use. “Total” is the total memory in the JVM (varies over time). “Free” is the free memory in the JVM (increases after a garbage collect).
- *Threads* : An estimate of the number of active process threads.
- *Connections* : The number of active connections to internal services and the database.
- *Proxy Print Queue* : The number of pending **Proxy Print** jobs.
- *Job Ticket Queue* : The number of pending **Job Tickets**.

4.3.2. Services

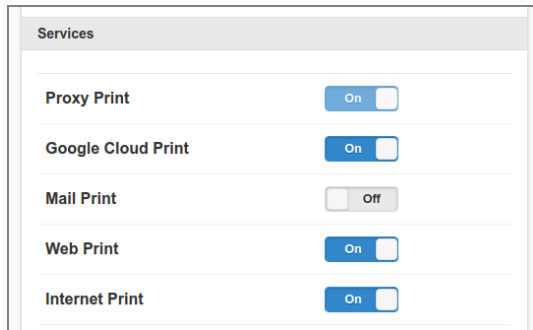


Figure 4.7. Admin Web App: Dashboard - Services

This section lists the status of services.

- The Proxy Print indicator is read-only, and shows the CUPS connection status. When connectivity is broken, the switch shows “Off”, with the reason displayed at the right.
- Core services like **Google Cloud Print** and **Mail Print** must be *enabled* to be on the list.
- **Web Print** and **Internet Print** services are a fixed part of the list.
- **Plug-in** services like **Mollie** and **Blockchain.info** Payment Gateways are part of the list if they are enabled in their property file.

With **Dashboard Editor Privileges**, you can turn a service **On** or **Off** by flipping the status switch. *With Reader Privileges service switching is disabled.*

When the SavaPage server restarts enabled *core* services are turned **On** by default. The initial state of enabled *plug-in* services is governed by the online setting in their property file. The on/off state of **Internet Print** translates to the enabled/disabled state of the reserved / internet **Queue**.

4.3.3. News

The News section shows the currently installed versus the latest published SavaPage version. A push on the button brings you to the **Downloads and Release Notes** Internet page ¹.

4.3.4. Pagometers

The *Pagometers*² counting the pages printed-out with Proxy Printers, printed-in from SavaPage Queues, and exported as PDF are displayed in a Pie-Chart. Pagometers are explained at [Section 4.10.14.9, “Pagometers” \[155\]](#).

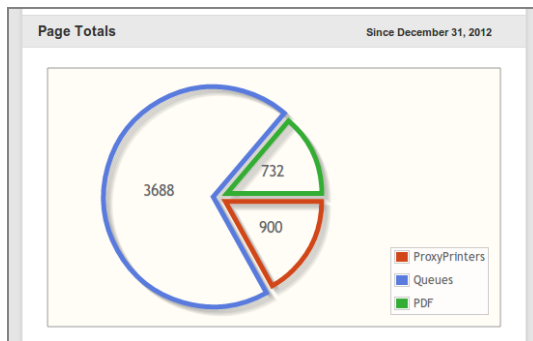


Figure 4.8. Admin Web App: Dashboard - Pagometer

¹ The latest published SavaPage version number is cached on the server and retrieved from the Internet every 12 hours.

² In analogy with the term *Odometer*, the term *Pagometer* is introduced as an instrument to count the number of processed pages.

A Line-Graph shows the day pagometers for the three sources over the last 30 days.

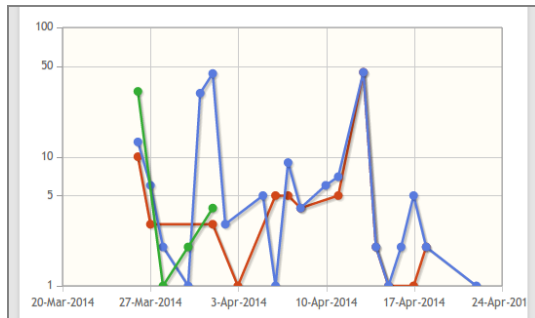


Figure 4.9. Admin Web App: Dashboard - Pagometer Trend

4.3.5. Environmental Impact

The Environmental Impact for the Proxy Printer pagometer are displayed in a separate section. The metrics and units used are discussed at [Section 12.2, “Environmental Impact” \[214\]](#).

Environmental Impact	
Paper	1,495 SU
Trees	17.941%
CO ₂	7.62 kg
Energy	0.019 kW-h

Figure 4.10. Admin Web App: Dashboard - Environmental Impact

4.3.6. Financial Summary

A Financial Summary of User Accounts and Bitcoin Wallet is displayed in a separate section.

Financial			
Accounts	Debit	Credit	
EUR	9.57		
Min	2.82		
Max	3.71		
Avg	3.19		
Count	3		
Bitcoin Wallet	Debit	Date	
EUR	12.23	2015-07-07T14:55:44	
BTC	0.05042385		
Addresses	Total	Payments	Open
Refresh	6	4	1

Figure 4.11. Admin Web App: Dashboard - Financial Summary

The User **Accounts** total and statistics like **Min**, **Max** and **Avg** are shown as **Debit** or **Credit** amount over **Count** number of accounts.

When a **Bitcoin Payment Gateway** is enabled the **Bitcoin Wallet** balance (**Debit**) is shown in the **system currency** and BTC. The **Total** number of Bitcoin **Addresses** in the wallet are split into addresses that received **Payments**, and **Open** addresses waiting for payments. Note that other addresses, not created by our **Bitcoin Payment Plug-in**, may be part of the wallet (in our example there is one such address).

The **Bitcoin Wallet** hyperlink opens the Web Wallet in a new browser tab.

The Accounts summary is updated as the dashboard is (auto) refreshed. However, the Bitcoin Wallet summary is cached by SavaPage and lazy refreshed after a configurable time period (defaulting to 3600 seconds).³ The date/time of the last refresh is shown in the **Date** column. Press the **Refresh** button to force a refresh of the cache.

4.3.7. Activity

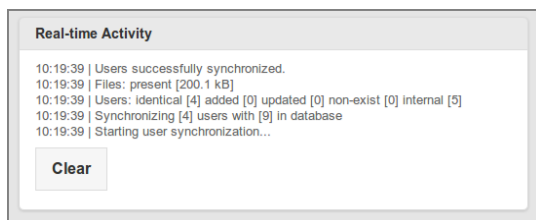


Figure 4.12. Admin Web App: Dashboard - Activity

Relevant system events are real-time displayed in this section. A maximum of 20 event messages remain in view, with the most recent one at the top.

System events are persisted in the rotating log file:

```
/opt/savapage/server/logs/adminpublisher.log
```

This file has a tab separated value (TSV) format for easy import and manipulation into spreadsheet programs. See `/opt/savapage/server/lib/log4j.properties.template` for more information.

4.4. Users

4.4.1. User List

After a *tap* on the **Users** button in the main menu this panel is shown. See [Section 4.2, “Menu” \[74\]](#).

³ Edit the `webapp.admin.bitcoin.wallet.cache-expiry-secs` configuration property with the **Configuration Editor** to set the number of seconds after which the cached Bitcoin Wallet summary is refreshed.

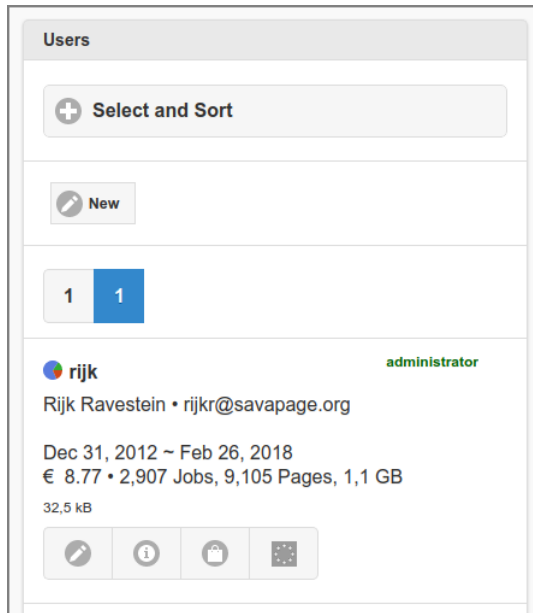


Figure 4.13. Admin Web App: User - List

- All non-deleted users are listed alphabetically by default. A different selection and sorting can be entered: see Figure 4.14, “Admin Web App: User - Select and Sort” [83].
- Press the **New** button to **create** and edit a new **Internal User**.
- The list can be traversed by tapping one of the buttons at the *pager* at the top or bottom of the page.
- An entry is displayed for each user, with identifying data and some usage statistics. From top to bottom:
 - The user's role or status (at the top right corner).
 - An inline pagometer Pie-Chart followed by the user id. The *blue* color in the chart represents the number of pages printed to SavaPage. The *green* color represents the number of pages exported to PDF. The *red* color depicts the pages printed to Proxy Printers.
 - The user id of an **Internal User** is shown with an *orange* color.
 - The full name and email address.
 - The period in which user activity was accumulated on the pagometer.
 - The account balance and the pagometer including the number of jobs and bytes printed to any SavaPage printer.
 - The size of the user's SafePages home.
- Tap the **Edit** button to change or delete the user. See Section 4.4.4, “Edit User” [84].



Note

Deleted Users cannot be edited.

- The **Documents** button brings you to the list of documents the user processed. See Figure 4.108, “Admin Web App: Documents - List” [157]
- The **Transactions** button brings you to the list of financial transactions on the user's account. For a detailed description of this list see Section 3.8.2, “Transactions” [61] in the User Web App chapter.
- The rightmost **GDPR** button opens a pop-up where personal user data can be downloaded. See Section 4.4.2, “Download Personal Data” [83].



Note

Due to **Admin Privileges** certain buttons might *not* be visible.



Tip

The pagometers of all users can be reset at [Options](#) → [Advanced](#) → [Reset Pagometers](#)

The screenshot shows a 'Select and Sort' panel with the following controls:

- Group:** A dropdown menu currently set to 'All Users'.
- ID:** A text input field.
- Email:** A text input field.
- Type:** A button with a minus sign, and two buttons labeled 'Person' and 'Abstract'.
- Role:** A button with a minus sign, and two buttons labeled 'Administrator' and 'User'.
- Status:** A button with a minus sign, and two buttons labeled 'Enabled' and 'Disabled'.
- Deleted:** A button with a minus sign, and two buttons labeled 'Deleted' and 'Active'.
- Sort:** A set of buttons including 'ID', 'Email', 'Ascending' (highlighted), and 'Descending'.
- Bottom Bar:** A row of buttons: 'Apply' (with a checkmark icon), 'Default' (with a circular arrow icon), a PDF icon, a CSV icon, and a minus icon.

Figure 4.14. Admin Web App: User - Select and Sort

- Users can be selected by **Group** and by entering a part (fragment) of their ID or Email. So entering "son" as ID will select both "jason" and "sonja".
- Select the Type, Role and (Deleted) Status. The **-** button will select both.
- The list can be sorted Ascending or Descending on ID or Email.
- Tap the **Apply** button to (re)display the list.
- A tap on the **Default** button resets the selection and sort fields to their default values.
- The PDF and CSV buttons downloads the User List in their respective formats, using the selection item values.
- The minus icon collapses the Select and Sort section.

4.4.2. Download Personal Data

This pop-up opens when the rightmost **GDPR** button is pressed in an entry from the **User List**.

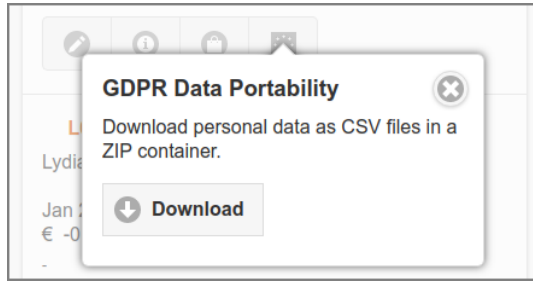


Figure 4.15. Admin Web App: User Data Portability

See [Section 14.2.1, “Data Portability” \[223\]](#).

4.4.3. Erased Users

Erased Users have their personal data cleared. Since their ID is empty, an “anonymous” placeholder with date/time of erasure is used for display.

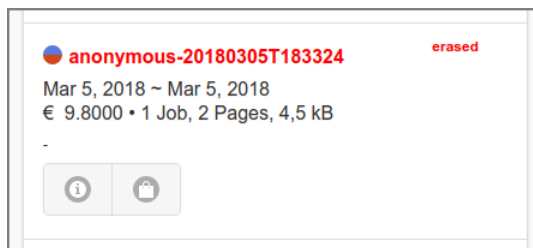


Figure 4.16. Admin Web App: Erased User

See [Section 14.2.2, “Data Erasure” \[223\]](#).

4.4.4. Edit User

This chapter describes the editable sections of the User entity.



Caution

Some data you edit, like the **Name**, **Primary email**, **Card Number** and **ID Number** might be overwritten by values from the user source during synchronization. See [Section 4.10.1.2, “LDAP” \[121\]](#) and [Section 4.10.2, “User Creation” \[124\]](#).



Note

Users can also be edited and deleted with the Server Command Tool. See [Section C.1.19, “setUserProperties” \[263\]](#) and [Section C.1.5, “deleteUser” \[258\]](#).

4.4.4.1. Identity

The screenshot shows a modal window titled 'User' with a close button. Inside, there's a section for 'rijck' (likely a typo for 'Rijk'). Below that, the 'Name' field contains 'Rijk Ravestein'. At the bottom, there are three checkboxes: 'Administrator' (checked), 'Person' (checked), and 'Disabled' (unchecked).

Figure 4.17. Admin Web App: Edit External User - Identity

- The user's full **Name** can be edited. Remember this name can be overwritten for an *external* User as a result of user synchronisation. See [Section 4.10.2, “User Creation” \[124\]](#).
- Assign the **Administrator** role by ticking the checkbox.
- Users are regarded as **Person** by default. Un-tick the Person checkbox if this user represents a generic **functional account** [209]. This will make the user **Abstract**.
- Tick the **Disabled** checkbox to deny the user access to the SavaPage application.



Warning

When a User becomes **Abstract** its SafePages are removed.

4.4.4.2. User Roles

The screenshot shows a modal window titled 'Roles' with a minus sign icon. It contains a list of roles, each with a checkbox: 'Job Ticket Creator' (unchecked), 'Job Ticket Operator' (unchecked), 'Web Cashier' (unchecked), 'Print Job Creator' (unchecked), 'Print Job Delegator' (unchecked), and 'Print Job Delegate' (checked).

Figure 4.18. Admin Web App: Edit User - Roles

User Roles are needed to access certain application objects, as shown in the table below.

Role	Access
Job Ticket Creator	Job Ticket Printer

Role	Access
Job Ticket Operator	Job Tickets Web App
Web Cashier	Point-of-Sale Web App
Print Job Creator	A Proxy Printer that is <i>not</i> a Job Ticket Printer.
Print Job Delegate	Delegated Print and Users and Groups with role “Print Job Delegator” for Delegated Print.
Print Job Delegator	This is a passive role. Delegators can be accessed by users with role “Print Job Delegate”.

Table 4.1. User Roles

Each role is set with a checkbox that has three states:

- *Checked* : The role is *enabled*.
- *Unchecked* : The role is *disabled*.
- *Unchecked and grayed out*: The role is *indeterminate*.

If a User Role is needed to access an application function, SavaPage will check if this role is *enabled* for the authenticated user.

When the role is *indeterminate* at the user level, **Group Roles** are checked of the groups the user belongs to. **Added Groups** are checked first, then the **Built-in Groups**, with the “All Users” group as last.

- Access is *granted* if there is at least one group where the role is *enabled*.
- Access is *denied* when the role is *indeterminate* or *disabled* in all groups.
- Print Job Creator role is special: an *indeterminate* state at “All User” top level is interpreted as *granted*.



Caution

The 3-tier group hierarchy (User Groups > Internal/External Users > All Users) is traversed bottom up, to resolve the role of individual Users only. Group hierarchy is *not* used to resolve roles for User Groups: roles defined at group level are fixed, and are not interpreted in the context of other groups, or individual members.

4.4.4.3. Email

Figure 4.19. Admin Web App: Edit User - Email

- The **Primary email** and **Other emails** addresses are editable and must each be *unique*: they can be associated to just one User. Multiple emails must be separated by any of the characters *space*, *comma*, *semicolon*, or by carriage return or line feed.

4.4.4.4. Card and ID

Figure 4.20. Admin Web App: Edit User - Card

- The **Card Number** and **ID Number** are editable and must each be *unique*: they can be associated to just one User.
- The **ID Number** is used as authentication token for **Internet Print**.
- The **Card Number** must be entered in HEX/LSB format. See [Section B.1, “Card Number Format” \[251\]](#).
- The **PIN** must be digits only.
- The minimum length of **ID Number** is contained in configuration key `user.id-number-length-min`. The minimum and maximum length of a **PIN** are contained in the configuration keys `user.pin-length-min` and `user.pin-length-max`. A maximum value 0 (zero) indicates the maximum is unspecified. See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change these values.
- The **YubiKey Public ID** is used for YubiKey Authentication.
- Press the **OK** button to commit the changes and return to the User List.
- The **Cancel** button brings you back to the User List without changing anything.

4.4.4.5. UUID

Figure 4.21. Admin Web App: Edit User - UUID

The UUID⁴ is used as authentication token for **Internet Print**. It is automatically created when a user successfully logs in for the first time. A new UUID can be created by pushing the **Generate** button.

4.4.4.6. Financial

This section shows the personal **User Account**. Initialization of this account is based on Group Membership as explained in the **Edit Group** section.

⁴ A universally unique identifier (UUID) is an identifier standard used in software construction. See https://en.wikipedia.org/wiki/Universally_unique_identifier

Figure 4.22. Admin Web App: Edit User - Financial

- A new value for the user's account **Balance** results in a financial transaction that corrects the previous account balance. See [Section 3.8.2, “Transactions” \[61\]](#). The user is notified by a pop-up message in his active User Web App when his balance is adapted.
- Set the **Credit limit** with one of these buttons:
 - **None** : user has no credit limit, and is not restricted.
 - **Default** : user has default credit limit. See [Section 4.10.12.2, “General Financial Options” \[146\]](#).
 - **Individual** : when selected a custom credit limit can be entered.

4.4.4.7. Password

Figure 4.23. Admin Web App: Internal User - Password Actions

For an **Internal User Password** actions are shown.

The **Erase** button is shown when a password is set. When pressed, it erases the password and makes itself disappear again. Without an initial password, users cannot reset their password in the **User Web App**. *This gives administrators a means to disable login by user name/password, in favor of other authentication methods.*

A tap on the **Reset** button shows the **Password Reset Dialog**. Use this dialog to initially set or change a password.

A screenshot of a web application dialog titled "guest-peter" with a close button (X). The dialog contains a "Reset password" section. It has two input fields: "New password" and "Confirm password". At the bottom, there are two buttons: "Apply" (with a checkmark icon) and "Cancel" (with a circular arrow icon).

Figure 4.24. Admin Web App: Internal User - Password Reset

4.4.4.8. User Delete

A screenshot of a "Delete" dialog. It has a title bar "Delete" and a message: "Deleting a user is a destructive action which cannot be undone." At the bottom, there are two buttons: "Delete" (with an X icon) and "Cancel" (with a circular arrow icon).

Figure 4.25. Admin Web App: Edit User - Delete

- Press the **Delete** button to delete the user and return to the User List. The [next](#) section describes the effect of this action.
- The **Cancel** button bring you back to the User List without changing anything.

4.4.5. Create Internal User

A tap on the **New ...** button at the top of the **User List** gives this dialog to create a new **Internal User**. Apart from the regular User data, the attributes **ID** and **Password** can be entered.

- The prefix of **ID** is contained in the configuration key `internal-users.username-prefix`.
- The minimum length of the **Password** is contained in the configuration key `internal-users.password-length-min`.
- See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change these configuration values.
- The **Financial** data are initialized with the **New User Settings** of the **Built-in Internal Users Group**. If these new user settings are disabled the **Balance** is set to zero with an **Individual** Credit limit of zero.



Tip

Internal Users can also be added with the Server Command Tool. See [Section C.1.2, “addInternalUser” \[256\]](#).

4.4.6. Deleted Users

Deleting a User makes sense if he is not part of the user source anymore and was not deleted as part of a bulk delete during a manual synchronization. As long as job history or account transactions for a User are present ⁵, SavaPage applies a logical delete. Any logical deleted User will be physically deleted from the database when no related job history and account transactions are present anymore. This situation will automatically occur when you enabled **auto-backup** in combination with the delete of old document and transaction logs.



Important

If SavaPage synchronizes a new User from the user source, a new user instance will be created in the database, despite the fact that a logical deleted User exists with the same identifying name, i.e. the logical delete status of the "identical" user will remain as it is.

4.4.7. Administrator Role

SavaPage sets up a dedicated account called `admin`. This is the master administrator account, with access to all application functions, whose password is assigned during configuration. In large organizations it is likely that the administrator role needs to be granted to more than one person. One solution is to give all those persons the master password; however a better approach is to assign the administrator role to the network user accounts of these individual's. The advantages of this approach are:

- Administrators can access the Admin Web App with their own username and password.
- Since most administrative activity is logged in an audit trace, changes can easily be tracked back to an individual.



Note

Access to certain parts of the Admin Web App can be set on User Group level with **Admin Privileges**.



Tip

Administrative users should login via `https://savapage:8632/admin` rather than `https://savapage:8632/` or `https://savapage:8632/user` so that they are directed to the correct interface.

4.5. Groups

Groups are collections of users. You can **Add and Remove** groups as present in the external **User Source** or **Internal Group** definition.



Note

SavaPage caches group members for performance reasons. Therefore, when group membership changes at the source, it may not be immediately known in SavaPage. The membership cache is updated automatically according to the "Import new users overnight" option in the **User Creation** section, but can be also be refreshed manually at any time by a push on a button in the same section.

4.5.1. Built-in Groups

There are three built-in groups:

⁵ When a users does not print on his own, but is printed for via Delegated Print, no job history is present for that user, but (pending) transactions are.

- *All Users* : all users in the system.
- *External Users* : all users synchronized from the external [User Source](#).
- *Internal Users* : all users created inside SavaPage. See [Section 4.10.1.3, “Internal Users”](#) [123].

4.5.2. Group List

After a *tap* on the **Groups** button in the main menu this panel is shown. See [Section 4.2, “Menu”](#) [74].

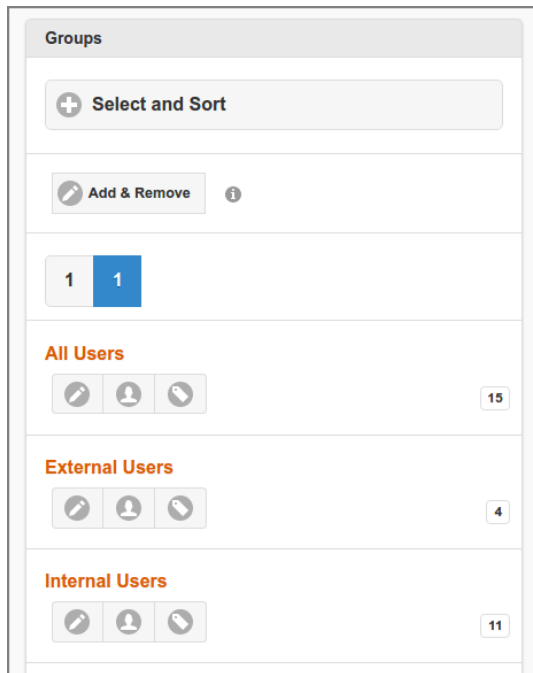


Figure 4.26. Admin Web App: User Group - List

Built-in groups are depicted in orange. Press the **Add & Remove** button to [add additional groups](#).

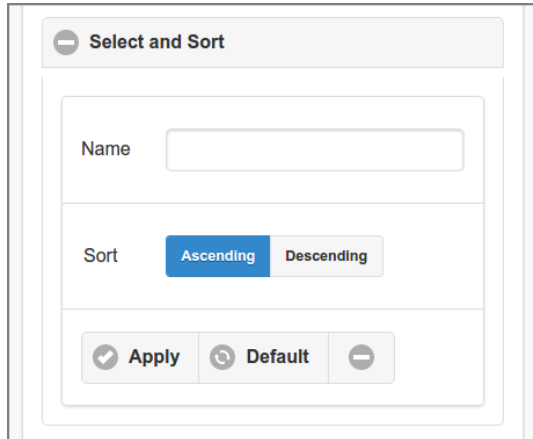
Each item in the list shows the number of members and has buttons to jump to other dialogs. From left to right, these buttons bring you to:

- The [Edit Group](#) dialog.
- The [User List](#) with the group preselected.
- The [Account List](#) with the Group Type and Name preselected. *Note: since the Group Account is lazy created upon first use it might be that the list is empty.*



Note

Due to [Admin Privileges](#) certain buttons might *not* be visible.

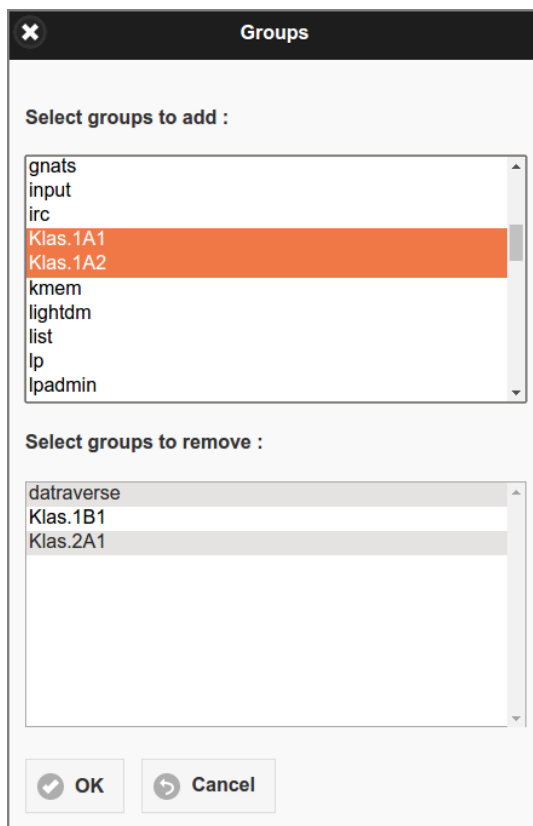


The 'Select and Sort' dialog is a light gray box with a title bar containing a minus icon and the text 'Select and Sort'. Inside, there is a 'Name' input field. Below it, the 'Sort' section has two buttons: 'Ascending' (highlighted in blue) and 'Descending'. At the bottom, there are three buttons: 'Apply' (with a checkmark icon), 'Default' (with a circular arrow icon), and a minus icon for collapsing the dialog.

Figure 4.27. Admin Web App: Group - Select and Sort

- Groups can be selected by entering a part (fragment) of their name.
- The list can be sorted Ascending or Descending on group name.
- Tap the **Apply** button to (re)display the list.
- A tap on the **Default** button resets the selection and sort fields to their default values.
- Use the “minus” button to collapse the Select and Sort section.

4.5.3. Add & Remove Groups



The 'Groups' dialog has a dark header with a close icon and the title 'Groups'. It contains two sections: 'Select groups to add :' and 'Select groups to remove :'. The first section has a list of group names: gnats, input, irc, Klas.1A1, Klas.1A2, kmem, lightdm, list, lp, and lpadmin. The second section has a list of group names: dataverse, Klas.1B1, and Klas.2A1. At the bottom, there are two buttons: 'OK' (with a checkmark icon) and 'Cancel' (with a circular arrow icon).

Figure 4.28. Admin Web App: User Groups - Add & Remove

Select the groups to add and to remove and press the **OK** button to commit the selection.



Note

The group list is a mix from the ones present in the external **User Source** and the ones defined as **Internal Group**. When adding a user group from Microsoft Active Directory, members from nested groups are included.

4.5.4. Edit Group

The Group Edit Dialog has several sections. Press the **OK** button at the bottom to commit all changes.

4.5.4.1. Group Roles

Figure 4.29. Admin Web App: User Group - Edit - Roles

In the **Roles** section you can set the user roles for group members. See [Section 4.4.4.2, “User Roles” \[85\]](#) for an explanation of the roles and how role based user access works.

4.5.4.2. User Privileges

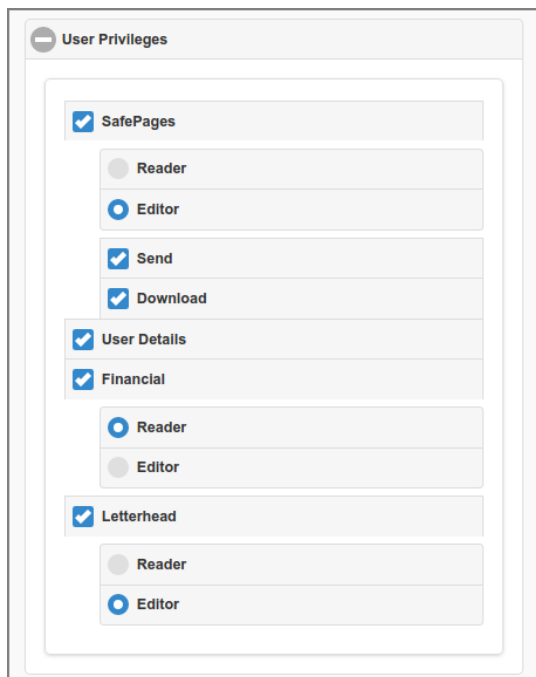


Figure 4.30. Admin Web App: User Group - Edit - User Privileges

In the **User Privileges** section you can set group member access to **User Web App** domain objects. Privileges are set by means three-state buttons. An unselected *grayed out* button means “indeterminate”, plain unselected means “non-privileged” and selected means “privileged”. When a privilege on a domain object is selected a role like **Reader** and **Editor** might be selected, as well as extra actions like **Send** and **Download**. The type of Roles and Actions offered depend on the type of domain object. This is how choices work out:

- When **User Details** is non-privileged the footer button for the **User Details** dialog is replaced with a simple indicator holding the id of the authenticated user.
- When **SafePages** is non-privileged, the ☐ PDF and ☐ Sort buttons are not displayed in the **Main Page**. When selected, the Reader role will display the PDF but not the Sort button, and the Editor role will display both. The Send and Download options activate the respective buttons in the **PDF** dialog.
- When **Financial** is non-privileged, the account balance will not show in the **footer**, the Transactions button will not show in the **Log** page, and Financial data will not show in the **User Details** dialog. When selected, the Reader and Editor role will display all. However, only the Editor role is allowed account transactions in the **User Details** dialog.
- When **Letterhead** is non-privileged, the ☐ Letterhead button is not displayed. When privileged the Reader and Editor role allows user to choose a Letterhead in the **PDF** and **Print** dialog. The Editor role allows users to add letterheads themselves. See [Section 3.6, “Letterheads” \[57\]](#).
- The open spots left by buttons that are not displayed are taken by: the Upload button (moved from the footer), a Browse button pointing to the **Browser**, and the Info button (moved from the footer), in that order. See [Section 3.2.2, “Footer” \[29\]](#).

This is how a privilege is evaluated on runtime:

- To be compatible with existing installations the “indeterminate” state for top level group “All Users” is interpreted as fully “privileged”. Of course, privileges can also be set at “lower” group levels. When determining privileges for a domain object, SavaPage looks at the lowest group first, and bubbles up to higher groups till a “non-indeterminate” privilege for the domain object is found.
- A denial of access due to a privilege takes precedence over any other configuration property.

4.5.4.3. Admin Privileges

In the **Administrator Privileges** section you can set group member access to **Admin Web App** domain objects. The objects correspond to the choices in the **main menu**. Any user with **Administrator Role** is assigned privileges by group membership.

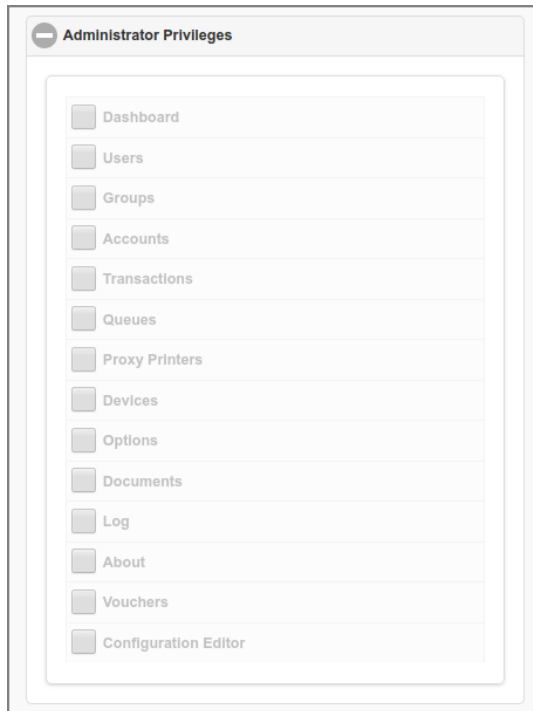


Figure 4.31. Admin Web App: User Group - Edit - Admin Privileges

Privileges are set and evaluated by means three-state buttons, just as **User Privileges**. For most domain objects a **Reader** and **Editor** role can be selected. Access to domain objects will be shown or hidden according to the privileges.

4.5.4.4. New User Settings

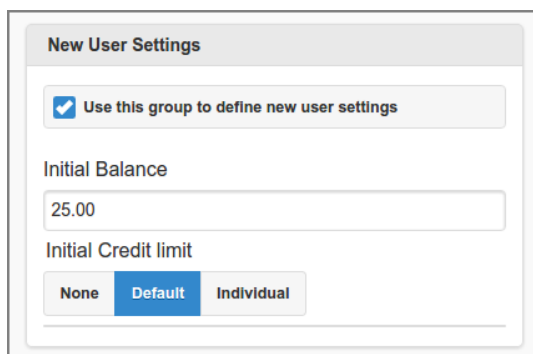


Figure 4.32. Admin Web App: User Group - Edit - New User Settings

When **New User Settings** are *enabled* they are automatically applied upon **User Creation** for members of this group. Note that these settings do **not** affect existing user members. See the **Financial** section of the **Edit User** dialog for a description of the Balance and Credit Limit fields.

When a user belongs to **multiple** groups, the New User Settings of these groups is applied as follows:

- The user is assigned an initial Balance that is the *sum* of the Initial Balances of *all* matching groups (with the exception of the **Built-in Groups**).
- If *any* of the matching groups has Initial Credit Limit “None” the user is assigned this status.
- Since the New User Settings are applied in alphabetical group name order, the Initial Credit Limit “Default” and “Individual” are assigned from the *last* group.

When a user does **not** belong to any group with New User Settings enabled, he is assigned the settings of the “External Users” or “Internal Users” **Built-in Group** (depending on the type of **User Source**).



Note

New User Settings are not shown for Built-in Group “All Users” because they are never used.

4.6. Accounts

This section is about *Shared Accounts* as explained in [Chapter 8, SavaPage Financial \[189\]](#). The accounts are utilized in Delegated Print and **PaperCut** print scenarios.

4.6.1. Account List

After a tap on the **Accounts** in the main menu this panel is shown. See [Section 4.2, “Menu” \[74\]](#).

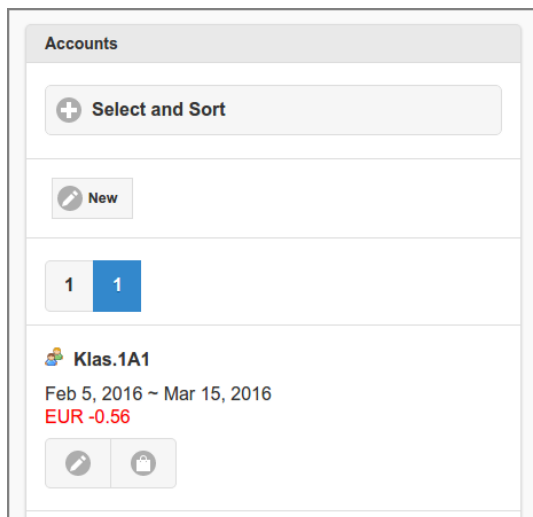


Figure 4.33. Admin Web App: Account - List

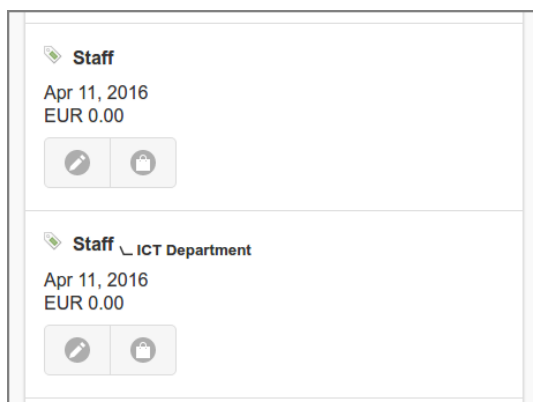




Figure 4.34. Admin Web App: Account - List - Sub Accounts

- All accounts are listed alphabetically by default. A different selection and sorting can be entered: see [Figure 4.35](#), “Admin Web App: Account - List - Select and Sort” [97].
- The  icon signifies a plain Shared Account and  a Group Account.
- Press the **New** button to create and **Edit** a new Account.
- The list can be traversed by tapping one of the buttons at the pager at the top or bottom of the page.
- An entry is displayed for each Account, with the account type icon and name, the transaction period and the balance amount.
- Parent _ child accounts are depicted with a subscripted smaller font child name.
- Tap the **Edit** button to change or delete the account. See [Section 4.6.2](#), “Edit Account” [97].
- The **Transactions** button brings you to the list of transactions. This list has an identical setup as the one for User Account transactions: see [Section 3.8.2](#), “Transactions” [61].



Note

Due to [Admin Privileges](#) certain buttons might *not* be visible.

Figure 4.35. Admin Web App: Account - List - Select and Sort

- Accounts can be selected by entering the containing text (fragment) of their **Name**.
- Select the **Type** and **Deleted** Status. The **-** button will select both.
- The list can be sorted Ascending or Descending on Name.
- Tap the **Apply** button to (re)display the list. A tap on the **Default** button resets the selection and sort to their default values. The minus button collapses the section.

4.6.2. Edit Account

Group Accounts are ad-hoc created in the [Delegated Print](#) scenario, and have names identical to their User Group. Editing of Groups Accounts is limited to **Balance**, **Notes** and the **Delete** option. **Shared Accounts** can be freely created, and can fully be edited as explained below.

Figure 4.36. Admin Web App: Account - Edit

- The **Name** must be case insensitive unique for parents and for children within a parent. For instance, the following Parent\Child name combinations can coexist: A\C, B\C, C\C. Shared and Group Accounts have different name spaces, so account name “A” can exist as Shared *and* Group Account. A “.” (point) character can not be part of an account name.
- *Group Accounts cannot act as Parent Account.* Therefore the **Parent Account** *must* be a Shared Account.
- An Account that acts as Parent Account for other accounts can have no Parent Account itself.
- In the **Access control** section names of **User Groups** can be entered, whose members can use this account to charge (printing) costs on. See [Section 3.4.4, “Print Job Settings” \[42\]](#) and [Section 3.4.7, “Delegated Print Edit” \[46\]](#).
- A new **Balance** value results in a financial transaction that corrects the previous account balance. See [Section 3.8.2, “Transactions” \[61\]](#).
- Tick the **Delete** checkbox to delete the Account. This will be a logical delete as long as transactions are present. Any logical deleted Account will be physically deleted from the database when no related transactions are present anymore. This situation will automatically occur when you enabled automatic backup in combination with the delete of old account transactions. See [Section 4.10.13, “Backups” \[148\]](#).
 - A delete of a parent account cascades to all of its child accounts. An un-delete *does not*.
- Press the **OK** button to commit the changes and return to the Account List.
- The **Cancel** button brings you back to the Account List without changing anything.

4.7. Queues

Queues are print-in channels for acquiring [SafePages](#).

4.7.1. Reserved Queues

Dedicated queues are reserved and pre-installed for:

- The default IPP “/” queue.
- The Raw IP Printer Port (which defaults to 9100)
- [Google Cloud Print](#)
- [AirPrint](#)
- [Mail Print](#)
- [Web Print](#)

4.7.2. Queue List

After a *tap* on the **Queues** button in the main menu this panel is shown. See [Section 4.2, “Menu”](#) [74].

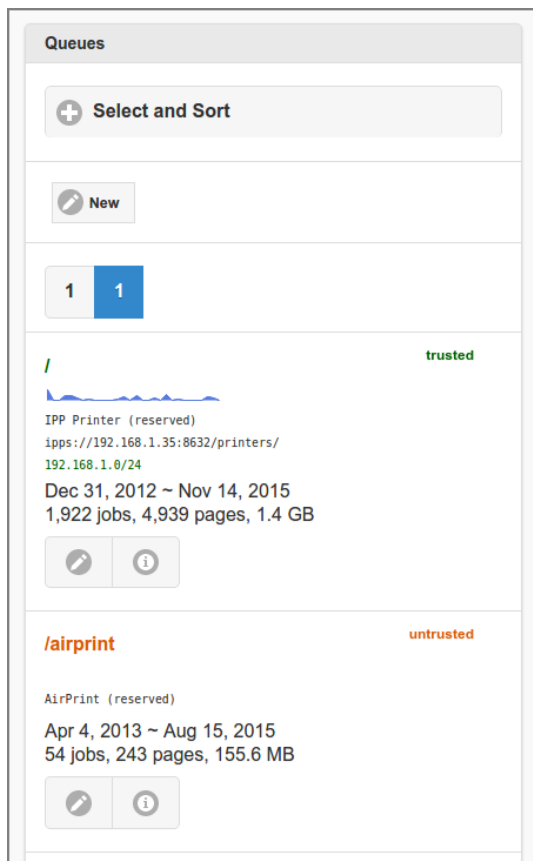


Figure 4.37. Admin Web App: Queue - List

- All non-deleted queues are listed alphabetically by default. A different selection and sorting can be entered: see [Figure 4.38, “Admin Web App: Queue - Select and Sort”](#) [100].
- Press the **New** button to create and **edit** a new queue.
- The list can be traversed by tapping one of the buttons at the *pager* at the top or bottom of the page.

- An entry is displayed for each queue, with identifying data and some usage statistics. From top to bottom:
 - The queue's trust or status (at the top right corner).
 - The URL Path of the queue. The path is relative to the `/printers` URL base.
 - An inline Line-Graph showing the day pagometers of the printed pages over the last 30 days.
 - The full IPPS [URL variant](#) of the queue.
 - Optionally, the allowed client IPv4 addresses as a [CIDR Set](#).
 - The period in which activity was accumulated on the pagometer.
 - The pagometer of the queue including the number of jobs and bytes printed.
- Tap the **Edit** button to change or delete the queue. See [Section 4.7.3, “Edit Queue” \[101\]](#)
- The **Log** button brings you to the list of documents printed to the queue. See [Figure 4.108, “Admin Web App: Documents - List” \[157\]](#)



Note

Due to [Admin Privileges](#) certain buttons might *not* be visible.



Tip

The pagometers of all queues can be reset at [Options → Advanced → Reset Pagometers](#)

The screenshot shows the 'Queues' management interface. A 'Select and Sort' panel is open, allowing users to filter and sort the queue list. The 'Containing text' field is empty. Under 'Trust', the 'Trusted' button is selected. Under 'Status', the 'Enabled' button is selected. Under 'Deleted', the 'Active' button is selected. The 'Sort' section has 'Ascending' selected. The 'Apply' button is highlighted with a checkmark, and the 'Default' button has a refresh icon.

Figure 4.38. Admin Web App: Queue - Select and Sort

- Queues can be selected by entering the containing text (fragment) of their URL Path.
- Select the queue's Trust and (Deleted) Status. The **-** button will select both.
- The list can be sorted Ascending or Descending on URL Path.

- Tap the **Apply** button to (re)display the list.
- A tap on the **Default** button resets the selection and sort fields to their default values.

4.7.3. Edit Queue

Figure 4.39. Admin Web App: Queue - Edit

- The **URL Path** is editable. Renaming the URL path name will permanently overwrite the old name in all related job history records with the new name.
- Enter IPv4 address ranges as a **CIDR Set** at **IP addresses allowed** to restrict access to the queue based on requesting IP address. If the field is empty all requesting IP addresses are allowed to print to the queue.
- Tick the **Trusted** checkbox to make this a **Trusted Queue**. When this option is not selected the queue will be a **Public Queue**.
- Tick the **Disabled** checkbox to disable access to the queue for all users.
- Tick the **Delete** checkbox to delete the Queue. This will be a logical delete as long as related job history is present. Any logical deleted Queue will be physically deleted from the database when no related job history is present anymore. This situation will automatically occur when you enabled automatic backup in combination with the delete of old document logs. See [Section 4.10.13, “Backups” \[148\]](#).
- Press the **OK** button to commit the changes and return to the Queue List.
- The **Cancel** button brings you back to the Queue List without changing anything.



Important

Some reserved queues like **Google Cloud Print**, **Web Print** and **Mail Print** can not be edited. Other reserved queues like **AirPrint** and **Internet Print** are untrusted by nature, hence the field **Trusted** cannot be edited.

4.8. Proxy Printers

4.8.1. Proxy Printer List

After a tap on the **Proxy Printers** button in the main menu this panel is shown. See [Section 4.2, “Menu” \[74\]](#).

SavaPage automatically detects CUPS printer queues, and uses their unique name to replicate corresponding Proxy Printers in its database. When a printer queue is deleted in CUPS it is real-time detected by SavaPage, and as a result the corresponding Proxy Printer is marked as *not present*. Proxy Printers which are *not present* are hidden in the User Web App, so they cannot be used for printing. When a printer queue is renamed in CUPS, two events occur in SavaPage. First, the new name is detected as a new Proxy Printer, and second, the Proxy Printer with the old name is detected as a deleted CUPS queue. Proxy Printers which are not present anymore can either be *deleted* or *renamed*.

SavaPage selects *local* CUPS printer queues by default. This is a sensible policy since local CUPS queues are able to connect to locally attached printers (USB, LPT) or network enabled printers. However, in some odd cases you might want to proxy print to a *remote* CUPS queue, i.e. a queue shared by another machine. In that case you can set the value of the `cups.ipp.remote-enabled` config item to `Y`. See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change this value.

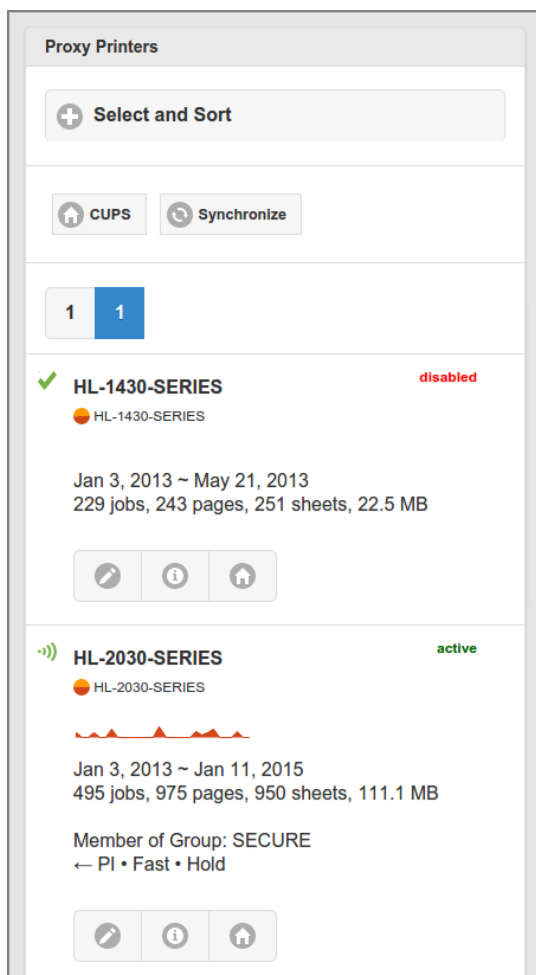


Figure 4.40. Admin Web App: Proxy Printer - List

- All non-deleted Proxy Printers are listed alphabetically by default. A different selection and sorting can be entered: see [Figure 4.41, “Admin Web App: Proxy Printer - Select and Sort” \[104\]](#).
- The **CUPS** button is shown when the browser points to “localhost”, “127.0.0.1”, a “.local” address, or to the intranet IP address of the SavaPage server. Click the button to open the [CUPS Administration web page](#) in a new browser tab.
- Click the **Synchronize** button to synchronize all CUPS printer options to SavaPage. Since SavaPage does not automatically detect changed printer defaults (yet), you need to perform this action after you change CUPS printer properties, so users will see the right defaults in the Common Printer Dialog of their Web App.
- The list can be traversed by tapping one of the buttons at the *pager* at the top or bottom of the page.

- An entry is displayed for each Proxy Printer, with identifying data and some usage statistics. From top to bottom:
 - The Proxy Printer status (at the top right corner).
 - The display name (alias) of the printer as shown to the user.
 - An inline pagometer Pie-Chart followed by the CUPS Name of the printer. The *red* color in the chart represents the number of *pages* printed. The *orange* color represents the number of printed *sheets*.
 - An inline Line-Graph showing the day pagometers of the printed pages over the last 30 days.
 - The make and model of the printer.
 - The period in which activity was accumulated on the pagometer.
 - The pagometer of the Proxy Printer including the number of jobs, sheets and bytes printed.
 - The **Printer Groups** this Proxy Printer is member of.

Tap the **Edit** button to edit the entry. Proxy Printers which are present in CUPS can be edited: see [Figure 4.42, “Admin Web App: Proxy Printer - Edit - Identity” \[105\]](#). Proxy Printers which are *not* present in CUPS can also be *deleted* or *renamed*: see [Figure 4.47, “Admin Web App: Proxy Printer - Rename” \[110\]](#)

The **Log** button brings you to the list of documents printed to the Proxy Printer. See [Figure 4.108, “Admin Web App: Documents - List” \[157\]](#)

The **Home** button is shown under the same condition as the **CUPS** button. Click to open the [CUPS Administration web page](#) for the printer in a new browser tab.



Note

Due to [Admin Privileges](#) certain buttons might *not* be visible.



Important

The CUPS Administration web page must be accessible as explained in [Section 2.3, “Step 3 - Configure CUPS and Samba” \[12\]](#). When CUPS authentication is required you can log in with user name `root` or with a user name that belongs to the admin group.



Tip

The pagometers of all Proxy Printers can be reset at [Options → Advanced → Reset Pagometers](#)

Each Proxy Printer in the list is marked with an icon:

- ✓ A non-secure Proxy Printer which can be used by any device.
 - See [Section 4.10.10, “Proxy Print” \[141\]](#).
- ✗ A non-secure Proxy Printer which can *not* be used by any device.
 - See [Section 4.10.10, “Proxy Print” \[141\]](#).
- 🔑 A secured Proxy Printer whose jobs needs to be authorized with a NFC Card swipe on a Network Card Reader.
 - See [Section 4.9.1.2, “Proxy Print Authentication” \[114\]](#).
 - A referral to the Reader and the enabled release functions are shown in the list item.
- ✓ A Proxy Printer which can only be used from certain Terminals.

- See [Section 4.9.2.1, “Custom Proxy Print” \[118\]](#).

🔑 A Proxy Printer which can only be used from certain Terminals and whose jobs needs to be authorized with a NFC Card swipe on a Network Card Reader on other Terminals.

🎫 A Job Ticket Printer.

- See [Section 4.8.2.1, “Proxy Printer Identity” \[105\]](#).

4.8.1.1. CUPS Printer Class

A CUPS Class is a group of printers. When printing to it, CUPS will redirect the job to one of the printer members. Which printer would depend on user rights or which printer currently is available.

A CUPS Class appears as a regular Proxy Printer in the list: the number of member printers is enclosed in parenthesis, just after its CUPS name.



Important

SavaPage requires that CUPS Class members all have the same PPD.

4.8.1.2. Select and Sort

The screenshot shows the 'Proxy Printers' management interface. A sidebar on the left contains a 'Select and Sort' section with the following controls:

- Containing text:** A text input field with 'HL-' entered.
- Status:** Two buttons, 'Enabled' (highlighted in blue) and 'Disabled'.
- Deleted:** Two buttons, 'Deleted' (highlighted in blue) and 'Active'.
- Sort:** Two buttons, 'Ascending' (highlighted in blue) and 'Descending'.
- Buttons:** At the bottom of the sidebar are 'Apply' (with a checkmark icon) and 'Default' (with a circular arrow icon) buttons.

Figure 4.41. Admin Web App: Proxy Printer - Select and Sort

- Proxy Printers can be selected by entering the containing text (fragment) of their display name. So, entering "HL-" will select both "HL-1430-SERIES" and "HL-2030-SERIES".
- Select the (Deleted) Status. The **-** button will select both.
- The list can be sorted Ascending or Descending on display name.
- Tap the **Apply** button to (re)display the list.

- A *tap* on the **Default** button resets the selection and sort fields to their default values.

4.8.2. Edit Proxy Printer

4.8.2.1. Proxy Printer Identity

RICOH-AFICIO-MPC-7500PS

Display as

Location

Proxy Printer Groups

☒ **Disabled**
☐ **Internal use**

PPD Extension File

☐ **Job Ticket Printer**

Proxy Printer Group with compatible redirect printers.

Figure 4.42. Admin Web App: Proxy Printer - Edit - Identity

- The title bar shows the CUPS Printer Name.
- The **Display** name and **Location** are editable.
- Multiple **Proxy Printer Groups** can be entered separated by space, comma, colon or semicolon.
- Tick the **Disabled** checkbox to disable access to the Proxy Printer for all use.
- Tick the **Internal use** checkbox to mark the printer for internal use only. This will disable access to the Proxy Printer from the User Web App, but the printer can still be used for internal print work flow scenarios.
- The **PPD Extension File** is optional and refers to a .ppde text file with extensions to the assigned PPD in CUPS. This is the way to map vendor specific PPD options to IPP attributes. See [Appendix K, PPD Extensions \[292\]](#).
- By ticking the **Job Ticket Printer** checkbox the printer will act as Job Ticket Printer and its display name will be shown to users with role **Job Ticket Creator** for printer selection. You must enter the **Proxy Printer Group** containing the printers the Job Ticket print job can be redirected to. See [Section 4.8.3, “Printer Groups” \[109\]](#) and [Chapter 5, Job Tickets Web App \[174\]](#).

- This option can *not* be selected for PaperCut managed printers (a warning message is displayed).
- Press the **Apply** button to commit the changes and return to the Proxy Printer List.
- The **Cancel** button brings you back to the Proxy Printer List without changing anything.

See [Section 3.4.8, “Job Ticket Print” \[51\]](#) for an explanation of Job Tickets and how users can create them. Also see:

- [Chapter 5, *Job Tickets Web App* \[174\]](#).
- [Section A.2.1, “Delegated Print - \(Non\) Secure & Job Ticket Scenarios” \[247\]](#).

4.8.2.2. Printer Costs

Printer costs are specified per media *side* and can be set for One-sided and Two-sided prints and differentiated for B/W (Black and White) and Color printing.

When SavaPage calculates the cost of a Proxy Print job, the two-sided (duplex) page cost is only applied to pages that are part of a sheet that is printed on both sides. So, for a document with an odd number of pages, the two-sided cost is not applied to the last page. For example, when a 7 page document is printed as two-sided, the two-sided cost is applied to the first 6 pages, and the one-sided cost to the last.

Costs are displayed, and can be entered, with a precision (number of decimals) as defined in [Section 4.10.12.2, “General Financial Options” \[146\]](#).

Printer costs are irrelevant and ignored in the following cases:

- When this is a **Job Ticket Printer** and **Job Ticket Media Costs** are defined in the **PPD Extension File**.
- When a printer is selected from the **Job Ticket Print** dialog (Job Ticket cost is leading).
- When PaperCut cost is leading. This is the case for a PaperCut Managed Printer, and when all conditions below are met:
 - This is a Non-Secure Printer (Secure Printing must be enforced by PaperCut). See [Section 4.10.10, “Proxy Print” \[141\]](#).
 - **PaperCut Integration** is enabled.
 - **PaperCut Delegated Print Integration** is enabled.
 - A **Print Job** is *not* charged to a Personal Account, but to a Shared Account or via the **Delegated Print - PaperCut Scenario**.

4.8.2.3. Media Sources

In this section you can assign Media Size and Costs to Media Sources.

Media Source

Cost per media side in EUR

☒ Use B/W as default

☐ Perform B/W conversion locally

Source	One-sided	* Two-sided		
*auto	B/W	Color	B/W	* Color
<input checked="" type="checkbox"/> tray-1	0.0500	0.1000	0.0400	0.0800
A4 - white	A4			
<input checked="" type="checkbox"/> tray-2	0.0600	0.1200	0.0500	0.1000
A4 - yellow	A4			
<input checked="" type="checkbox"/> tray-3	0.1000	0.2000	0.0800	0.1600
A3 - white	A3			

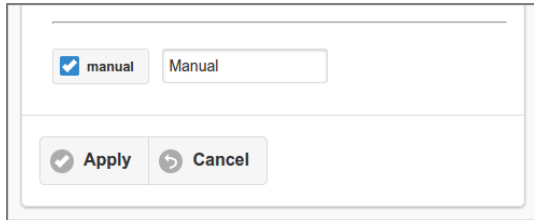
Figure 4.43. Admin Web App: Proxy Printer - Edit - Media Source

- CUPS printer defaults are indicated in the header with an asterisk (*). In some odd cases the CUPS printer color mode default “grayscale” is not correctly transferred as IPP attribute. You can easily correct this behavior by setting the **Use B/W as default** option at the top of the **Media Source** section. Instead, you can also use a **PPD to IPP mapping** to set the default for the IPP **sides** attribute: this correction is more basic, and immediately be reflected in the default * indicator.
- When the **Perform B/W conversion locally** option is set grayscale print jobs are converted to B/W before sending them to a color capable proxy printer. This setting is needed when you observe that print jobs with color content are printed in color despite the Grayscale Color Mode setting in the **Printer Settings** dialog. Indeed, some PPDs expect grayscale conversion is done client-side.
- The ***auto** text below the Source header indicates that the printer supports automatic tray selection based on media options.
- Each entry in the list has a checkbox with the IPP attribute keyword of the Media Source.
- Tick the checkbox to enable the media source, and enter a user-friendly name as will show up in the Media Source picklist of the **Printer Settings** dialog.
- Select the Media Size that is present in the Media Source.
- Specify the costs. Notice that only those cost cells are enabled that are applicable for the printer.



Warning

Depending on the PPD file used for the CUPS printer, some media sources might not be applicable. You are advised to do some tests to make sure that media sizes are indeed applicable to the media sources as you intended.



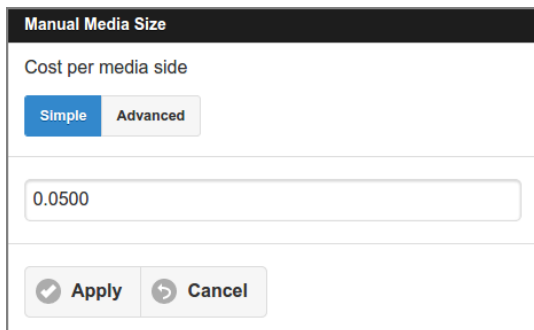
The screenshot shows a web form titled "Manual Media Source". It contains a checked checkbox labeled "manual" next to a text input field containing the word "Manual". Below this, there are two buttons: "Apply" with a checkmark icon and "Cancel" with a circular arrow icon.

Figure 4.44. Admin Web App: Proxy Printer - Edit - Manual Media Source

- Costs for the *manual* media source can not be entered here, but must be specified as described in the [next section](#).

4.8.2.4. Manual Media Sizes

In this section you can specify the Proxy Printer media costs for the *manual* media source. You can either use a **Simple** or **Advanced** definition.



The screenshot shows a web form titled "Manual Media Size". It has a section labeled "Cost per media side" with two tabs: "Simple" (selected) and "Advanced". Below the tabs is a text input field containing the value "0.0500". At the bottom, there are two buttons: "Apply" with a checkmark icon and "Cancel" with a circular arrow icon.

Figure 4.45. Admin Web App: Proxy Printer - Edit - Manual Media Size (Simple)

The **Simple** definition allows for a single cost per media side. This is appropriate for a non-duplex monochrome Proxy Printer that can handle a single media size (Letter or A4) only.

Manual Media Size

Cost per media side

Simple Advanced

	One-sided		Two-sided	
	B/W	Color	B/W	Color
<input checked="" type="checkbox"/> Default	0.0500	0.1000	0.0300	0.0700
<input type="checkbox"/> US Letter				
<input type="checkbox"/> A4				
<input checked="" type="checkbox"/> A3	0.1000	0.2000	0.0600	0.1400
<input type="checkbox"/> A5				
<input type="checkbox"/> C5				
<input type="checkbox"/> US Legal				

Show selected

Apply Cancel

Figure 4.46. Admin Web App: Proxy Printer - Edit - Manual Media Size (Advanced)

Advanced mode is best suited for a duplex color Proxy Printer that can handle multiple media sizes.

- The list of supported media sizes is dependent on the Proxy Printer type.
- Use the check-box at a media size to enable its custom cost specification.
- Costs for unspecified (disabled) media sizes fall back to the **Default** specification.

4.8.3. Printer Groups

Printer Groups allow administrators to combine Proxy Printer instances so they can be addressed as group by a single *tag*. A Proxy Printer can have one or more groups tags. See [Section 4.8.2, “Edit Proxy Printer” \[105\]](#).

Printer Groups are used to customize access to Proxy Printers. See:

- [Section 4.9.1.2, “Proxy Print Authentication” \[114\]](#)
- [Section 4.9.2.1, “Custom Proxy Print” \[118\]](#)
- [Section 4.10.10, “Proxy Print” \[141\]](#)
- [Section 4.8.2.1, “Proxy Printer Identity” \[105\]](#).



Note

Printer Group tags are added to the database on first use. Tags without Proxy Printer members are removed from the database at the start of the application and thereafter at a daily schedule.

4.8.4. Rename Proxy Printer

When a Proxy Printer is removed from the host system it is marked in the list as *not present*. When editing new options appear, as is shown in the screenshot below.

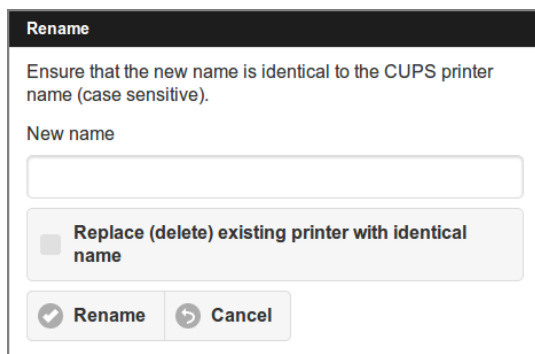


Figure 4.47. Admin Web App: Proxy Printer - Rename

- See [Figure 4.42, “Admin Web App: Proxy Printer - Edit - Identity” \[105\]](#) for a description of the basic edit options.
- Tick the **Delete when job history is cleaned** button to logically delete the Proxy Printer. It will be physically deleted from the database when no related job history is present anymore. This situation will automatically occur when you enabled automatic backup in combination with the delete of old document logs. *Deleting makes sense if the queue is permanently removed from CUPS*, and you don't want the Proxy Printer list in the Admin Web App to be cluttered with out-of-date "not present" Proxy Printers.
- Press the **OK** button to commit the changes and return to the Proxy Printer List.
- You can change the associated CUPS printer by entering a **New name**. *Renaming makes sense as a mirroring action of renaming a CUPS queue*. After renaming a printer in CUPS, the Proxy Printer associated with the old CUPS name will be identified by SavaPage as "not present", and a new Proxy Printer for the new CUPS queue will be created. At this point you can re-associate (rename) the old CUPS name of the Proxy Printer to the new one. This will work as long as no job history is already accumulated on the Proxy Printer associated with this new CUPS name. To overrule this constraint you can tick the **Replace (delete) existing printer with identical name** checkbox, so an existing Proxy Printer associated with the same (new) CUPS name will be deleted and replaced.
- Press the **Rename** button to commit the renaming action and return to the Proxy Printer List.
- Both **Cancel** buttons bring you back to the Proxy Printer List without changing anything.



Caution

If SavaPage detects a CUPS queue whose name is identical to a logical deleted Proxy Printer, the logical delete mark will be removed and the Proxy Printer will be re-activated.

4.9. Devices

A Device is a hardware component with a dedicated function.

SavaPage defines devices of type **Terminal** and **Network Card Reader**. They are identified by IP address and, in case of the reader, a port number. The combination of IP-address and Type must be unique.



Note

Although most of the time an IP address will harbor one device type, this constraint does allow that Terminal and Network Card Reader are combined on a single physical device.

- A Network Card Reader acts as *User Authenticator*, either at *Login* or *Proxy Print* time.
- A Terminal device runs customized SavaPage Web App Sessions overriding *global User Authentication* and *Proxy Print* defaults.

After a *tap* on the **Devices** button in the main menu this panel is shown. See [Section 4.2, “Menu” \[74\]](#).

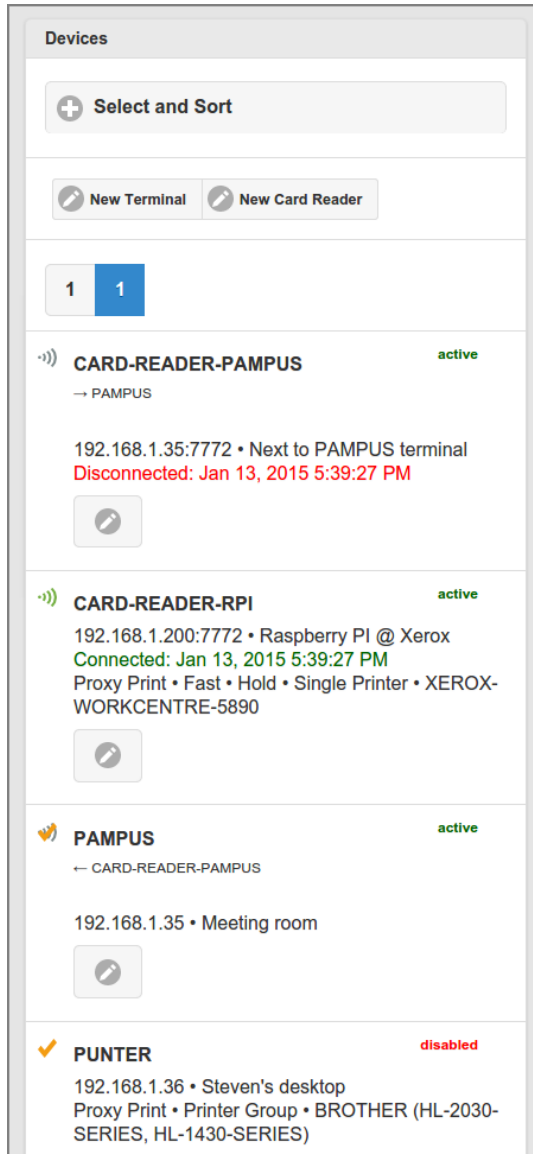


Figure 4.48. Admin Web App: Device - List

All Devices are listed alphabetically by default. A different selection and sorting can be entered: see [Figure 4.49, “Admin Web App: Device - Select and Sort” \[112\]](#). Press the **New Terminal...** or **New Card Reader...** button to create a new Device.

Devices are marked with an icon and hold the following items:

·) A Network Card Reader used for NFC Card Login at a Terminal.

- The name of the Reader.
- A referral to the Terminal.

- The IP address, port, and location of the Reader.
- A time-stamp of the last connection attempt with the status *connected* (green) or *disconnected* (orange).

📶 A Network Card Reader used for NFC Card *Proxy Print Authentication*.

- The name, IP address, port, and location of the Reader.
- A time-stamp of the last connection attempt with the status *connected* (green) or *disconnected* (orange).
- The **Proxy Printer** or **Proxy Printer Group** for which this reader acts as authenticator.
- The Print Mode of the proxy printer(s): Fast, Hold, Direct.

✓ A Terminal with custom settings.

- The name, IP address and location of the Terminal.
- The **Proxy Print** or **Proxy Printer Group** for which this Terminal is entitled to print to.

👤 A Terminal with custom settings and a Network Card Reader used for NFC Card Login.

- The name of the Terminal.
- A referral to the Reader.
- The IP address and location of the Terminal.
- The **Proxy Printer** or **Proxy Printer Group** for which this Terminal is entitled to print to.



Note

Due to **Admin Privileges** certain buttons might *not* be visible.

The screenshot shows the 'Devices' management interface. A 'Select and Sort' panel is open, containing the following controls:

- Containing text:** A text input field for filtering devices by name.
- Status:** A dropdown menu (indicated by a '-' icon) with 'Enabled' and 'Disabled' options.
- Type:** A dropdown menu (indicated by a '-' icon) with 'Reader' and 'Terminal' options.
- Sort:** Two buttons, 'Ascending' (highlighted in blue) and 'Descending'.
- Buttons:** 'Apply' (with a checkmark icon) and 'Default' (with a refresh icon) buttons at the bottom.

Figure 4.49. Admin Web App: Device - Select and Sort

- Devices can be selected by entering the containing text (fragment) of their display name. So, entering "CARD-" will select both "CARD-READER-PAMPUS" and "CARD-READER-RPI".
- Select the Status. The - button will select both Enabled and Disabled devices.

- Select the Type. The ☐ button will select both Reader and Terminal devices.
- The list can be sorted Ascending or Descending on device name.
- Tap the **Apply** button to (re)display the list.
- A tap on the **Default** button resets the selection and sort fields to their default values.

4.9.1. Network Card Reader

A Network Card Reader device runs the SavaPage **Network Card Reader Service**.

4.9.1.1. Custom User Login

Figure 4.50. Admin Web App: Devices - Network Card Reader - Custom User Login

- A reader is associated to a Terminal in the Terminal dialog. See [Section 4.9.3, “Custom User Login” \[118\]](#). The associated **Terminal** is shown here with icon and name.
- Tick **Disabled** to disable the Device definition.
- The format of the **NFC Card Number** must be specified. See [Section B.1, “Card Number Format” \[251\]](#).



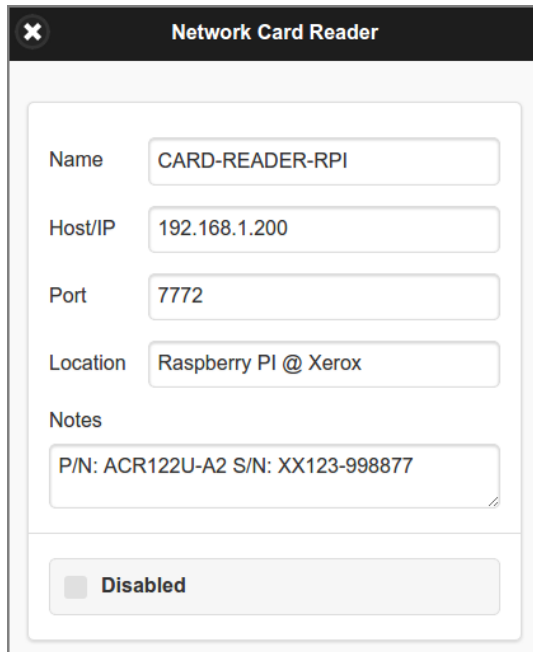
Note

A Network Card Reader can be used as NFC Card Login authenticator by just one Terminal.

4.9.1.2. Proxy Print Authentication

A Network Card Reader can act as print job authenticator for a single **Proxy Printer** or a **Proxy Printer Group**.

When the reader device is placed next to the printer device this setup implements **Follow-me Printing**⁶.



The screenshot shows a configuration window titled "Network Card Reader" with a close button (X) in the top left corner. The form contains the following fields:

- Name:** CARD-READER-RPI
- Host/IP:** 192.168.1.200
- Port:** 7772
- Location:** Raspberry PI @ Xerox
- Notes:** P/N: ACR122U-A2 S/N: XX123-998877

At the bottom of the form, there is a toggle switch labeled "Disabled", which is currently turned off.

⁶ [https://en.wikipedia.org/wiki/Follow_Me_\(printing\)](https://en.wikipedia.org/wiki/Follow_Me_(printing))

The screenshot shows a web form for configuring Proxy Print Authentication. At the top, there is a checkbox labeled "Proxy Print Authentication" which is checked. Below this, the "Print Mode" section contains five radio button options: "Fast", "Hold", "Direct", "Fast + Hold" (which is selected), and "Fast + Direct". The "Target" section has two text input fields: "Single Printer" containing "XEROX-WORKCENTRE-5890" and "Printer Group" containing "XEROX". The "NFC Card Number" section features four buttons: "HEX", "DEC", "LSB" (which is selected), and "MSB". At the bottom, there are three buttons: "OK", "Delete", and "Cancel".

Figure 4.51. Admin Web App: Devices - Network Card Reader - Proxy Print Authentication

Checking the **Proxy Print Authentication** option shows all the detail options. The **Print Mode** determines the authentication scenario. Basically there are three modes: **Fast**, **Hold** and **Direct**. Fast Mode can be combined with Hold and Direct Mode. See [Section 4.9.1.2.4, “Combining Print Modes” \[117\]](#).

- Depending on the Print Mode, the name of a **Single Printer** and/or **Printer Group Target** must be entered. Which targets are needed is explained in the Print Mode subsections below.
- The format of the **NFC Card Number** must be specified. See [Section B.1, “Card Number Format” \[251\]](#).



Important

When using Proxy Print Authentication concurrently with the [User Web App](#) and [User Client](#) you are strongly advised to use an external database like PostgreSQL. See [Chapter 17, Using an External Database \[236\]](#).

4.9.1.2.1. Fast Print Mode

Fast Print Mode applies to a **Single Printer** and supports the following scenario:

1. User prints one or more jobs to SavaPage. See [Chapter 10, SavaPage as Printer \[195\]](#).
2. User walks to the printer.
3. User swipes his NFC token along the reader.
4. As a result he gets one (1) printed copy of all SafePages jobs according to the default printer settings.

Expired SafePages jobs are *skipped* and *cleared*. The expiry period is set in [Section 4.10.10, “Proxy Print” \[141\]](#). A user can extend his fast job expiry in the User Web App. See [Section 3.4.1, “Printer Selection” \[41\]](#).

- A Fast Print clears *all* SafePages after printing.
- This scenario does not need any action in the User Web App. Therefore, it is the shortest way to proxy print with SavaPage.



Caution

Fast Print Mode is meant for straightforward proxy printing without intervening editing. So, when a user opens the Web App and performs a **Delete** or **Sort** on his SafePages he is supposed to Hold Print instead. When a Fast Print is tried on edited content, nothing is printed and all SafePages will be cleared as a result.



Important

When Fast Mode is combined with **Hold** or **Direct** Mode, and a **Printer Group** is specified, one of the printers from the group must be specified as **Single Printer**.



Tip

For Fast Mode printers with a single tray holding A4 media, **PPD Extensions** can be used to configure an extra virtual tray for Letter media. See [Section K.5.1, “Fast Print A4 and Letter to Single Tray” \[303\]](#).

4.9.1.2.2. Hold Print Mode

Hold Print Mode applies to a **Single Printer** or **Printer Group** and supports the following scenario:

1. User prints one or more jobs to SavaPage. See [Chapter 10, SavaPage as Printer \[195\]](#).
2. User opens the User Web App and **proxy prints** to either the Single Printer or one of the printers from the Printer Group.
3. As a result the proxy print job is held.
4. User walks to the chosen printer.
5. User swipes his NFC token along the reader.
6. As a result all hold jobs for the printer are printed.

Expired hold jobs are *skipped* and *cleared*. The expiry period is set in [Section 4.10.10, “Proxy Print” \[141\]](#). A user can delete hold jobs or extend expiry in the User Web App. See [Section 3.2.2.2, “Hold Print Jobs” \[30\]](#).

4.9.1.2.3. Direct Print Mode

Direct Print Mode applies to a **Single Printer** or **Printer Group** and supports the following scenario:

1. User prints one or more jobs to SavaPage. See [Chapter 10, SavaPage as Printer \[195\]](#).
2. User opens the User Web App in his mobile device.
3. User selects either the Single Printer or one of the printers from the Printer Group. See [Section 3.4.1, “Printer Selection” \[41\]](#).
4. User selects printer and job settings.
5. User walks to the chosen printer.
6. User presses the Print button in the Web App.
7. As a result a dialog pops up prompting the user to swipe his NFC Card for authentication. See [Section 3.4.5, “Direct Print Release” \[45\]](#). User has a 10 second time limit to swipe his card. The time limit is set in [Section 4.10.10, “Proxy Print” \[141\]](#).
8. User swipes his NFC token along the reader.
9. As a result the job is printed.

4.9.1.2.4. Combining Print Modes

Combining **Fast** and **Hold** Print Mode either in *one* proxy printer, or over *different* printers, needs special attention. Namely, if a user creates a Hold Print Job and does *not* delete the origin SafePages, the same SafePages will remain the target of a Fast Print, leading to duplicate prints. As this may be intended in rare cases, in the majority of cases this will not be intended by the user. To protect the user from unintended duplicate printing the following rules are applied:

- If Hold jobs exist for **any** printer, a Fast Print is only done when Proxy Printing is configured to “Always remove SafePages after printing”. This way, because all SafePages are cleared after creating a Hold Job, we know for sure that present SafePages are not part of any Hold Print Job. See [Section 4.10.10, “Proxy Print” \[141\]](#).
- As a consequence, when Proxy Printing is **not** configured to “Always remove SafePages after printing”, *all SafePages will be ad-hoc cleared after a Hold Print release*. This will prevent that, after all Hold jobs are released at Hold-only printers, and no Hold jobs exist for **any** printer, remaining SafePages will be Fast Printed anyway.

The Fast + Hold Print Mode supports the following scenario:

1. User creates a Hold Proxy Print Job.
 - All SafePages are removed after printing.
2. User prints one or more jobs to SavaPage.
3. User walks up to the Fast + Hold target printer of the Hold Print Job.
4. User swipes his NFC token along the reader.
5. As a result:
 - a. All Hold jobs for the printer are printed.
 - b. One (1) copy of all SafePages jobs is printed with the default printer settings.

When a user created Hold Print jobs for different printers, each supporting Fast + Hold Print Mode, obviously the Fast Print occurs at the first printer visited.

4.9.2. Terminal

A Terminal runs customized SavaPage Web App Sessions on a specific device.

The screenshot shows a web form titled "Terminal". It includes the following fields and controls:

- Name:** A text input field containing "PUNTER".
- Host/IP:** A text input field containing "192.168.1.36".
- Location:** A text input field containing "Steven's desktop".
- Notes:** A large, empty text area for additional information.
- Idle Seconds before auto logout:** A text input field containing "0".
- Disabled:** A checkbox that is currently unchecked, with the label "Disabled" next to it.

Figure 4.52. Admin Web App: Devices - Terminal - Custom Proxy Print

- The **Idle Seconds before auto logout** is targeted at public Terminals and is meant to protect users who forget to logout when done. Enter a number of seconds: the Web App will automatically logout after this period of inactivity. Enter 0 (zero) when no auto logout is desirable.

4.9.2.1. Custom Proxy Print ---

A Terminal can allow printing to a single **Proxy Printer** or **Proxy Printer Group**. This is usually done for printers that need to be secured according to *global Proxy Print* defaults.

When the Terminal device is placed next to a (group of) printer(s), this setup implements *Pull Printing*⁷.

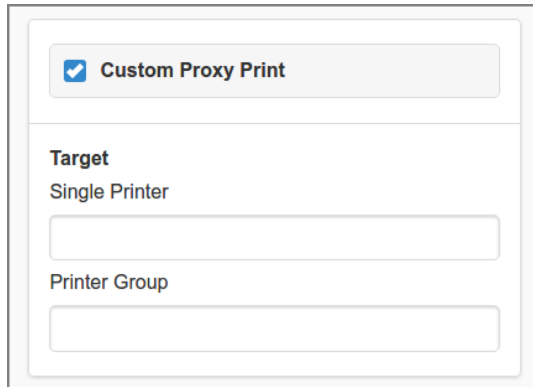
The image shows a web form for configuring a terminal's printing settings. At the top, there is a checkbox labeled "Custom Proxy Print" which is checked. Below this, under the heading "Target", there are two input fields. The first is labeled "Single Printer" and the second is labeled "Printer Group". Both fields are currently empty.

Figure 4.53. Admin Web App: Devices - Terminal - Custom Proxy Print

Check the **Custom Proxy Print** option and enter the name of a **Single Printer** or **Printer Group Target**.

4.9.3. Custom User Login ---

Check the **Custom User Login** option to override *global User Authentication* defaults just for this Terminal.

⁷ [https://en.wikipedia.org/wiki/Follow_Me_\(printing\)](https://en.wikipedia.org/wiki/Follow_Me_(printing))

The screenshot shows a configuration interface for 'Custom User Login'. It is divided into several sections:

- Custom User Login:** A single row with a checked checkbox.
- Authentication Methods:** A list of five options, each with a checked checkbox: Username, ID Number, Local NFC Card, Network NFC Card, and YubiKey.
- ID Number:** A sub-section containing two options: 'Mask input' (unchecked) and 'Require PIN' (checked).
- Network NFC Card Reader:** A section with a label and a text input field containing the value 'CARD-READER-PAMPUS'.
- NFC Card:** A sub-section containing two options: 'Require PIN' (unchecked) and 'Self Association' (checked).

Figure 4.54. Admin Web App: Devices - Terminal - Custom User Login

The screenshot shows a configuration interface for 'Default Login'. It contains a single section with five radio button options:

- Username
- ID Number
- Local NFC Card
- Network NFC Card** (This option is selected, indicated by a blue circle around the radio button.)
- YubiKey

Figure 4.55. Admin Web App: Devices - Terminal - Custom User Login - Default

- The options in these section are identical to the ones in [Section 4.10.3, “User Authentication” \[126\]](#) with the addition of the **Network NFC Card** option.
- Enter the name of the Network Card Reader below the **Network NFC Card Reader** label.

4.10. Options

A *tap* on the **Options** button in the **main menu** shows a panel options organized in sections. A *tap* on one of the sections expands (or collapsed) the underlying details. Please see the sections below for a detailed description:

- Section 4.10.1, “User Source” [120].
- Section 4.10.2, “User Creation” [124].
- Section 4.10.3, “User Authentication” [126]
- Section 4.10.4, “Mail” [130].
- Section 4.10.6, “Google Cloud Printer” [133].
- Section 4.10.7, “Mail Print” [137].
- Section 4.10.8, “Web Print” [139].
- Section 4.10.9, “Internet Print” [140].
- Section 4.10.10, “Proxy Print” [141].
- Section 4.10.11, “Eco Print” [143].
- Section 4.10.12, “Financial” [145].
- Section 4.10.13, “Backups” [148].
- Section 4.10.14, “Advanced” [149].



Note

Due to **Admin Privileges** the Option panel might be *disabled* and certain buttons *not* visible.

4.10.1. User Source

This section is about the configuration of external and internal user sources.

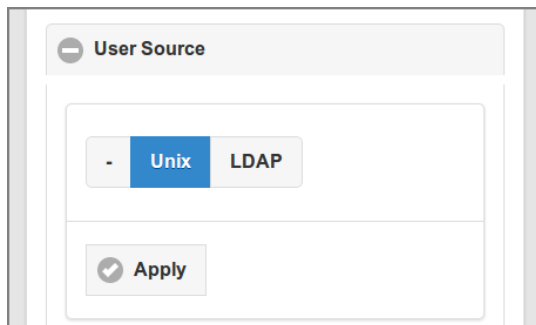


Figure 4.56. Admin Web App: Options - User Source

- *Tap* the **-** button if you do *not* want to import users from an external source. Remember to enable the **Internal Users** feature if you want to acquire any user into the system.
- *Tap* the **Unix** button if you want to import **Unix user accounts** defined on the SavaPage host.
- *Tap* the **LDAP** button to import users from an existing LDAP domain. This includes OpenLDAP, Apple Open Directory, Novell eDirectory and Microsoft Active Directory. When this option is selected the **LDAP connection data** are shown.
- Press the **Apply** button to commit the changes.

4.10.1.1. Unix

This option imports user accounts that are setup and defined on the local system as standard Unix accounts or mapped into the system from a central directory service such as LDAP via `nsswitch.conf` and PAM. Most large established Unix networks will support this option.

4.10.1.2. LDAP

LDAP (Lightweight Directory Access Protocol) directories usually store information about user and groups in an organization. One of the most common uses of LDAP is to provide **single sign-on** on a network that comprises multiple platforms and applications. When a network consists of Windows computers only, then an Active Directory domain can be used. But when there is a mix of Windows, Apple and GNU/Linux machines then LDAP can provide the single source of user, group and authentication information. (It is worth noting that both Active Directory and Novell eDirectory implement the LDAP protocol).

SavaPage can use an LDAP directory for user authentication and as a source of user and group information. LDAP can either be enabled at installation time, or by changing the user source at this point. When enabling LDAP, a number of configuration properties must be specified to allow the application to connect to the LDAP server. Please ask your LDAP administrator what values to use for the various options.

The screenshot shows the 'Options - User Source - LDAP' configuration page. At the top, there are three tabs: '-', 'Unix', and 'LDAP', with 'LDAP' being the active tab. Below this, there are four sub-tabs: 'OpenLDAP', 'Novell', 'Apple', and 'Windows', with 'OpenLDAP' being the active sub-tab. The configuration fields are as follows:

- Host:** A text input field containing 'zuideree'.
- Port:** A text input field containing '636'.
- Use SSL:** A checkbox that is checked.
- Trust Self-Signed Certificate:** A checkbox that is checked.
- Base DN:** A text input field containing 'DC=dataverse,DC=nl'.
- Admin DN:** A text input field containing 'CN=admin,DC=dataverse,DC=nl'.
- Admin password:** A password input field with masked characters (dots).

Figure 4.57. Admin Web App: Options - User Source - LDAP

- The Server Type determines which LDAP fields are used to get user and group information. Select one of the following server types that SavaPage supports out-of-the-box:
 - **OpenLDAP** : **OpenLDAP**.
 - **Apple** : **Apple Open Directory**.
 - **Novell** : **Novell eDirectory**.
 - **Windows** : **Microsoft Active Directory**.

However, it is easy to support other server types by adjusting the fields SavaPage uses for LDAP searches. This is discussed in [Appendix J, Advanced LDAP Configuration \[288\]](#)

- Enter the hostname or IP address of the LDAP server at the **Host** prompt.
- Enter the IP port of the LDAP server at the **Port** prompt. The value defaults to 389.
- Tick the **Use SSL** checkbox to use encrypted SSL connection to connect to the LDAP server. The LDAP server requires SSL support to be enabled and should accept connections on the standard LDAPS port 636.

- Check if you want to **Trust Self-signed certificate** of the LDAP server.
- Enter the Base DN of the LDAP server at the **Base DN** prompt. This is the equivalent of the “suffix” config setting of the OpenLDAP server. For example, if the domain hosted by the LDAP server is “domain.com” then the Base DN might be: `DC=domain,DC=com`

The format of the Base DN can differ significantly depending on the configuration. Some older Novell eDirectory installations may require a blank Base DN to operate. Some examples:

```
DC=myorganization,DC=com
DC=mycompany,DC=co,DC=uk
OU=OrgUnit,DC=domain,DC=com
DC=local
```

- The **Admin DN** is the DN of the user who has permission to connect to and query the LDAP server. This is typically an administrative user, although it can be a user that only has read-only access to the LDAP server. An example of the DN of the Administrator user on a Windows AD domain "domain.com", would be `CN=Administrator,CN=Users,DC=domain,DC=com`. The exact format of the DN depends on the LDAP server. Some examples:

- *Microsoft Active Directory (in organizational unit)*

```
CN=adminiator,OU=OrgUnit,DC=domain,DC=com
```

- *Apple Open Directory*

```
uid=diradmin,CN=users,DC=domain,DC=com
```

- *OpenLDAP*

```
uid=root,DC=domain,DC=com
uid=ldadmin,DC=domain,DC=com
```

- *Microsoft Active Directory*

```
CN=Administrator,CN=Users,DC=domain,DC=com
```

- *Novell eDirectory*

```
CN=root,DC=domain,DC=com
CN=ldadmin,OU=users,DC=domain,DC=com
```

- The **Admin password** is the password for the administrator specified in the Admin DN above.



Tip

Some LDAP servers are configured to allow “anonymous” LDAP query access. In these situations, the **Admin DN** and **Admin password** may be left blank.

The screenshot shows a configuration window titled "LDAP fields for alternative authentication". It contains three input fields: "ID Number" with the value "employeeNumber", "Card Number" with the value "pager", and "Card Format" with four buttons: "HEX", "DEC", "LSB" (which is selected), and "MSB".

Figure 4.58. Admin Web App: Options - User Source - LDAP

At the **LDAP fields for alternative authentication** section LDAP field names can be entered for the two alternative user authentication methods **ID Number** and **Card Number**, as described in [Section 4.10.3, “User Authentication”](#) [126]. Field names are optional and can be left empty. The **Card Format** is relevant when the **Card Number** is specified. See [Section B.1, “Card Number Format”](#) [251].



Important

The ID and Card Number must each uniquely identify a user, so make sure that no two users have the same number. This means that the numbers defined in LDAP should be unique. If SavaPage encounters a non-unique ID or Card Number that user will not be updated.

4.10.1.3. Internal Users

With the internal users feature you can directly manage users inside SavaPage. Enabling this feature removes the obligation to define an external User Source to create and manage Users. Of course you can enable this feature as an addition to an external source.

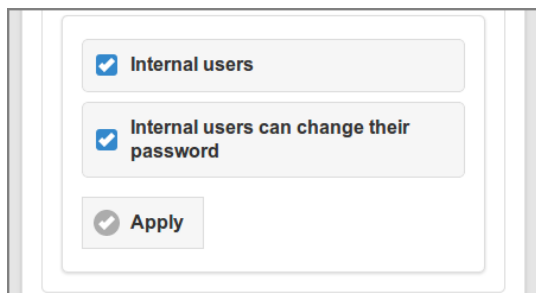


Figure 4.59. Admin Web App: Options - Internal Users

When **Internal Users** is selected an extra option appears where you can allow internal users to change their password. See [Section 3.10, “User Details”](#) [66].



Tip

Use the [Server Command Tool](#) to batch import internal users. See [Section C.1.2, “addInternalUser”](#) [256]

4.10.1.4. Internal Groups

SavaPage has the ability to define internal user groups. Just like [internal users](#) these groups are internal to the SavaPage system. Internal groups are created in addition to groups already provided by the external [user source](#) and are useful in the following situations:

- You have configured the system to import users from a source that does not support groups.
- You do not have permission to create new groups in the user source.
- You'd like to create small groups just for use within SavaPage and it's not appropriate to create a new group in the user source.

Internal Groups are defined in a plain text file and composed of members who are either synchronized from the external user source or who were created as internal user. A fully annotated template text file is present here:

```
/opt/savapage/server/data/conf/internal-groups.txt.tmpl
```

... copy this file to ...

```
/opt/savapage/server/data/conf/internal-groups.txt
```

... and start defining your groups.

Internal Groups are fully emancipated to their external fellows and can be moved in and out of scope. See [Section 4.5.3, “Add & Remove Groups” \[92\]](#).



Warning

Internal Groups should have a name distinctive to any groups defined in your external user source. In case of a name clash, the internal group takes precedence.

4.10.2. User Creation

This section is about the creation and synchronization of *external* users. *Internal* Users are created in the User Web App or with the Server Command Tool. See [Section 4.4.5, “Create Internal User” \[89\]](#) and [Section C.1.2, “addInternalUser” \[256\]](#).

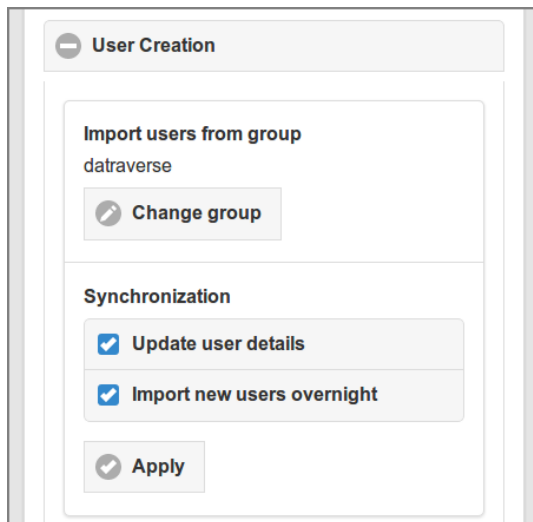


Figure 4.60. Admin Web App: Options - User Creation - Import

The **Import users from group** section holds an option to import a subset of users from the source by selecting a group. This option is relevant if you want to restrict access to SavaPage for a subset of *external* users.

- A *tap* on the **Change group** button shows a list of available groups as seen in [Figure 4.61, “Admin Web App: Options - User Creation - From Group” \[124\]](#).
- Select a group from the list and press the **Apply** button to commit the change, or press the **Cancel** button to roll it back.
- Use the **[All Users]** button to cancel the group restriction.

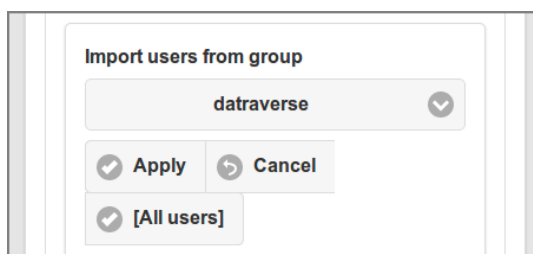


Figure 4.61. Admin Web App: Options - User Creation - From Group



Caution

In Active Directory, user group membership comes in two flavors. It can either explicitly be assigned, or be implied by the user's Primary Group ID, i.e. the RID of the primary group the user is member of. Because primary group membership is *implicit*, the Active Directory API will return an empty `member` attribute for this group. When users are *explicitly* assigned as member to groups the API will return group members as expected.

For example, because Active Directory sets the Primary Group ID of all users to the built-in “Domain Users” group, the Active Directory API will *not* report any members for the “Domain Users” group.

This issue is discussed in the following Microsoft Knowledge Base article: <https://support.microsoft.com/kb/275523>



Note

Active Directory supports an advanced feature called “Nested Groups”. This allows a group to have other groups as member. Since a sub-group can again have sub-group members, nesting can be several levels deep. When importing users from a group, SavaPage traverses the nested group tree to collect all containing users.

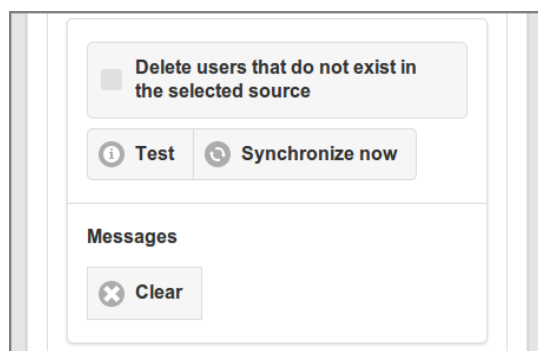


Figure 4.62. Admin Web App: Options - User Creation - Synchronize

The **Synchronization** section holds options for the *external* user synchronization process.

- Tick the **Update user details** checkbox if you want to overwrite user details in the database with details from the source.



Caution

An *external* User will overwrite an *internal* User with the same user id: as a result the User will become *external*.

- Select **Import new users overnight** to automatically synchronize daily at 10 minutes past midnight⁸.
- Press the **Apply** button to commit the changes.

Press the **Synchronize now** button to manually start a synchronization.

- Tick the **Delete users that do not exist in the selected source** checkbox to (logically) delete users in the database that were removed from the source. Note that this checkbox is deselected again after each synchronization.

⁸ Overnight user synchronization takes place according to the default **Cron Trigger Expression** "0 10 0 * * ?" contained in configuration key `schedule.daily`. See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change this value.

- Feedback messages from the synchronization process are real-time displayed in the **Messages** section. Press the **Clear** button to remove them.
- Use the **Test** button to check the effect of the synchronization without updating the database. Messages are shown with a "test" prefix.



Note

Disabled Active Directory users will *not* be imported by default. If you want to change this behavior you can set the value of configuration key `ldap.disabled-users.allow` to Y (or enter an empty value to fall back to the default). See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change this value.



Tip

To delete all external users, select **-** as User Source and use **Synchronize now** with the **Delete users** option.



Caution

The SafePages of *external* users not present in the source are deleted.

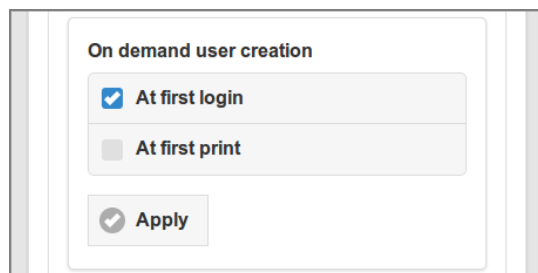


Figure 4.63. Admin Web App: Options - User Creation - On Demand

On demand user creation specifies which events, apart from regular user synchronization, will ad-hoc create new *external* users in the database.

- If the user associated with these events is not in the database, SavaPage will check if the user is part of the **User Source** and thereby a sure **Synchronized User** candidate. If so, it will ad-hoc synchronize the user into the database.
- Select **At first login** to ad-hoc create a user when he successfully passed the SavaPage **Login** for the first time.
- Select **At first print** to ad-hoc create a user when he prints to a SavaPage queue for the first time.
- Press the **Apply** button to commit the selection.

4.10.3. User Authentication

This sections describes the global defaults for User Authentication.

User Authentication

Persistence

☒ Browser Local Storage

PIN

☒ Users can change their PIN

Trust

☒ Trust User Client

NFC Card

☐ Require PIN

☒ Self Association

Figure 4.64. Admin Web App: Options - User Authentication

The **Persistence** section holds a options to persist authentication in **Browser Local Storage**. When *enabled*, a successful login to the SavaPage Web App will store an authentication token in the “Local Storage”⁹ of the browser. When the user closes the browser after a successful login and opens it again, login will be automatic, without the need to authenticate again. Separate authentication tokens are held for the User and Admin Web App context. See [Section 13.1.3, “Authentication Tokens” \[216\]](#).



Note

The presence of an authentication token is essential for the **iOS Web Clip** to function, and is pure convenience in other environments.

When **Browser Local Storage** is *disabled*, authentication persistence is implemented with **Web Sessions**.

The **PIN** section holds the defaults for PIN usage.

- When **User can change their PIN** is enabled users are granted the option to change their PIN. See [Section 3.10, “User Details” \[66\]](#).

When **Trust User Client** is *enabled* User Web App authentication is automatic when:

- An authenticated **User Client** session is present at the same IP address.
- The **User Web App** is opened with an URL parameter identifying the user from the User Client session. See [Appendix E, *URL Cheat Sheet* \[278\]](#).

The **NFC Card** section holds the defaults for card swipe logins using a **Local Card Reader** or **Network Card Reader**.

- With **Require PIN** enabled the user must also provide their associated PIN. This provides additional security for swipe card logins.
- When the **Self Association** option is selected, users are allowed to swipe cards previously not used or registered. When swiping such an unregistered card, SavaPage will ask the user if he wants to associate the new card to his

⁹ Local Storage is a W3C standard and stores data in the browser with no expiration date. The data will not be deleted when the browser is closed, and will be available the next day, week, or year.

account. When the user agrees SavaPage will switch to User Name authentication. After successful authentication the new card will be registered, thereby replacing any previously associated card. This feature is available for User Web App Login only. See [Section 3.1.5, “Card Self Association Dialog” \[26\]](#).

Figure 4.65. Admin Web App: Options - User Authentication - Login Methods

In the **Login Methods** section several login methods can be activated. When a method is activated a detailed section is shown. Detailed sections are explained in:

- [Section 4.10.3.1, “Username Login” \[128\]](#)
- [Section 4.10.3.2, “ID Number Login” \[128\]](#)
- [Section 4.10.3.3, “Local NFC Card Login” \[129\]](#)
- [Section 4.10.3.4, “YubiKey Login” \[129\]](#)
- [Section 4.10.3.5, “Default Login” \[130\]](#)



Note

The globally active Login Methods can be overridden for a specific [Terminal](#) by defining *Custom User Login* settings.



Note

ID Numbers and NFC Card Numbers can be synchronized with an external source like [LDAP](#), or imported from file.

4.10.3.1. Username Login

The **Username** login method allows a Person to use his regular username and password to login.

Figure 4.66. Admin Web App: Options - User Authentication - Username Login

If the **Show in Dialog** option is selected, the Username login method is part of the Login dialog. When this option is disabled this login can only be achieved by use of the `login` URL parameter. See [Appendix E, *URL Cheat Sheet* \[278\]](#).

4.10.3.2. ID Number Login

The **ID Number** login method allows a Person to use his identity number. Identity numbers are convenient when usernames are too long or cumbersome to enter. For example, rather than entering a username like “steven.brown-002”, it is more convenient to enter the employee or student ID Number “3624”.

The screenshot shows a settings panel for 'ID Number' login. It has a title 'ID Number' and three options: 'Show in Dialog' (checked), 'Mask input' (unchecked), and 'Require PIN' (unchecked).

Figure 4.67. Admin Web App: Options - User Authentication - ID Number Login

If the **Show in Dialog** option is selected, the ID Number login method is part of the Login dialog. When this option is disabled this login can only be achieved by use of the `login` URL parameter. See [Appendix E, *URL Cheat Sheet* \[278\]](#).

When **Mask input** is set the ID Number will be masked when entered (like a password).

With **Require PIN** set the user must also provide their associated PIN. This provides additional security for ID Number logins.

4.10.3.3. Local NFC Card Login

The **Local NFC Card** login method allows a Person to login by swiping an NFC Card across a [Local Card Reader](#).

The screenshot shows a settings panel for 'Local NFC Card' login. It has a title 'Local NFC Card', a 'Show in Dialog' checkbox (checked), and a 'Format' section with four buttons: 'HEX' (selected), 'DEC', 'LSB', and 'MSB'.

Figure 4.68. Admin Web App: Options - User Authentication - Local NFC Card Login

If the **Show in Dialog** option is selected, the Local NFC Card login method is part of the Login dialog. When this option is disabled this login can only be achieved by use of the `login` URL parameter. See [Appendix E, *URL Cheat Sheet* \[278\]](#).

The **Format** of the card number must be specified. See [Section B.1, “Card Number Format” \[251\]](#).



Warning

This setting applies to any [Local Card Reader](#) hooked up to any device. If a card reader is used that produces a different format a [Terminal](#) definition with a *Custom User Login* needs to be created for the device the reader is hooked up to.

4.10.3.4. YubiKey Login

The **YubiKey** login method allows a Person to login with a [YubiKey](#)¹⁰ Token.

The screenshot shows a settings panel for 'YubiKey' login. It has a title 'YubiKey' and a 'Show in Dialog' checkbox (checked).

Figure 4.69. Admin Web App: Options - User Authentication - YubiKey Login

¹⁰ <https://www.yubico.com/>

If the **Show in Dialog** option is selected, the YubiKey login method is part of the Login dialog. When this option is disabled this login can only be achieved by use of the `login` URL parameter. See [Appendix E, URL Cheat Sheet \[278\]](#).

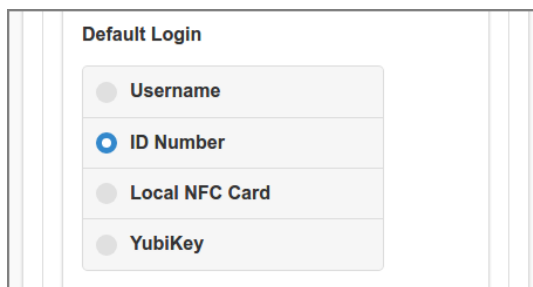
Get the YubiKey API credentials from [yubico.com](https://upgrade.yubico.com/getapikey/)¹¹, and enter them as configuration property.

Configuration property	Description
<code>auth-mode.yubikey.api.client-id</code>	The YubiKey API client ID.
<code>auth-mode.yubikey.api.secret-key</code>	The YubiKey API secret key.

Table 4.2. YubiKey Configuration Properties

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to set these items.

4.10.3.5. Default Login



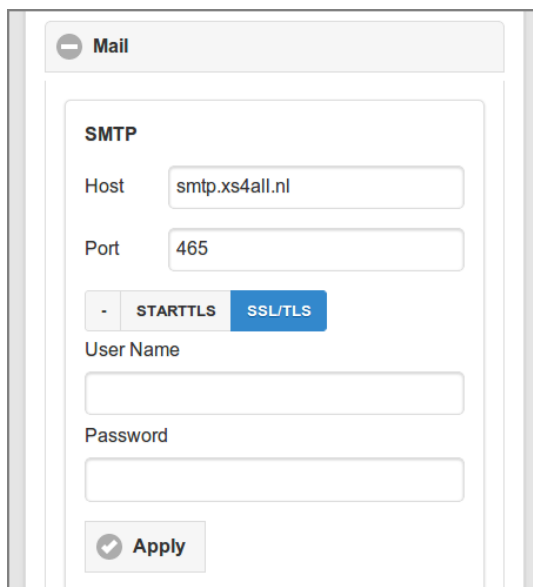
The screenshot shows a 'Default Login' section with four radio button options. The 'ID Number' option is selected, indicated by a blue dot. The other options are 'Username', 'Local NFC Card', and 'YubiKey', each with a grey dot.

Figure 4.70. Admin Web App: Options - User Authentication - Default Login

Select the Login method that is displayed as default in the Login dialog.

4.10.4. Mail

This section holds the settings for outgoing mail.



The screenshot shows the 'Mail' configuration section. Under the 'SMTP' heading, there are input fields for 'Host' (containing 'smtp.xs4all.nl') and 'Port' (containing '465'). Below these are two checkboxes: 'STARTTLS' (unchecked) and 'SSL/TLS' (checked). There are also input fields for 'User Name' and 'Password'. At the bottom, there is an 'Apply' button with a checkmark icon.

Figure 4.71. Admin Web App: Options - Mail - SMTP

¹¹ <https://upgrade.yubico.com/getapikey/>

Enter the **SMTP** connection parameters:

- The host name or IP address of the **Host**.
- The IP port at **Port**. The standard SMTP ports are: 25 (insecure), 465 (SSL/TLS) and 587 (STARTTLS). The value defaults to 465 (SSL/TLS).
- Select the connection security: - for an insecure connection, and **STARTTLS**¹² or **SSL/TLS**¹³ for a secure connection. TLSv1, TLSv1.1 and TLSv1.2 are supported.
- Enter the **User Name** and **Password** if authentication is required.

Messages

Sender address
admin@dataverse.nl

Sender name
Dataverse B.V.

Reply to address
no-reply@dataverse.nl

Reply to name
DO NOT REPLY

☒ Apply

Figure 4.72. Admin Web App: Options - Mail - Messages

The **Messages** section holds the sender and reply parameters used for email messages send by the system:

- **Sender address** : enter a valid email address representing the sender of the message.
- **Sender name** : the name default to SavaPage.
- **Reply address** : enter a valid email address the recipient can reply to.
- **Reply to name** : the name to reply to.

Press the **Apply** button to commit the changes.

Send a test message. Press the Test button after you applied the changes.

To

Figure 4.73. Admin Web App: Options - Mail - Test

Check if all mail parameters are valid by sending a test email.

- Enter a valid email address to send a message **To** and press **Test**. Check the mailbox of the recipient to see if the message arrived.

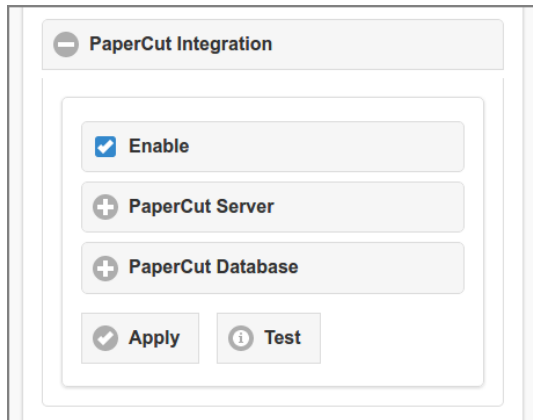
¹² STARTTLS is a way to take an existing insecure connection, and upgrade it to a secure connection using SSL/TLS.

¹³ SSL and TLS both provide a way to encrypt the communication between a client and a server computer. TLS is the successor to SSL and the terms SSL and TLS are used interchangeably.

4.10.5. PaperCut Integration

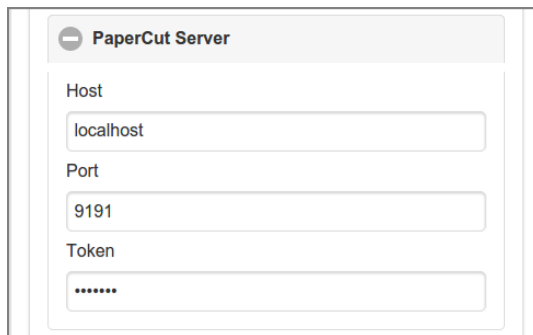
PaperCut is a popular print and copy management software product developed by [PaperCut Software](https://www.papercut.com/)¹⁴ based in Melbourne, Australia. Some functions not present in PaperCut can be implemented with SavaPage as pre-processor and integrator. See [Appendix N, PaperCut Integration \[315\]](#).

When PaperCut Integration is enabled, connectivity parameters for the **PaperCut Server** (XML-RPC API) and **PaperCut Database** (JDBC) can be entered. Press the **Apply** button to commit the changes. Press the **Test** button to test the PaperCut connectivity: a message confirming the connection status is displayed.



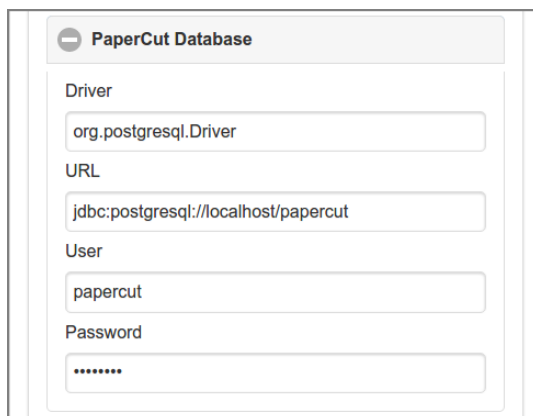
The screenshot shows the 'PaperCut Integration' section of the Admin Web App. It features a toggle switch labeled 'Enable' which is currently turned on. Below this are two expandable sections: 'PaperCut Server' and 'PaperCut Database', each with a plus icon. At the bottom of the section are two buttons: 'Apply' (with a checkmark icon) and 'Test' (with an information icon).

Figure 4.74. Admin Web App: Options - PaperCut Integration



The screenshot shows the 'PaperCut Server' configuration section. It contains three input fields: 'Host' with the value 'localhost', 'Port' with the value '9191', and 'Token' with a masked value represented by seven dots.

Figure 4.75. Admin Web App: Options - PaperCut Server



The screenshot shows the 'PaperCut Database' configuration section. It contains five input fields: 'Driver' with the value 'org.postgresql.Driver', 'URL' with the value 'jdbc:postgresql://localhost/papercut', 'User' with the value 'papercut', and 'Password' with a masked value represented by seven dots.

Figure 4.76. Admin Web App: Options - PaperCut Database

¹⁴ <https://www.papercut.com/>

4.10.6. Google Cloud Printer

Google Cloud Print¹⁵™ (GCP) is a Google service by which any Cloud Print aware application (web, desktop, mobile), on any device connected to the cloud, can print to any remote printer connected to that cloud.

The service is agnostic about the abundant combinations of client devices and target printers, and clients do *not* need to install device drivers to get things going. However, documents need to be fully transmitted to the Google cloud first, before they can be printed.

GCP is part of **Android and Chrome OS** and is, apart from that, available on all mobile devices and desktops via Google Cloud Print enabled Web Apps¹⁶.

Several hardware vendors have already integrated their solution with Google Cloud Print services so their printers can receive jobs from the Google cloud.

SavaPage closes the ranks with its own GCP integration so it truly qualifies as *Google Cloud Ready Printer*.



Note

Google Cloud Print maps to the reserved **Queue** /gcp.

4.10.6.1. Google Cloud Printer Registration

This section describes how to register the Google Cloud Printer just after you installed SavaPage.



Tip

During registration additional browser tabs and windows are opened. Therefore, it is more convenient to use a desktop browser during registration.

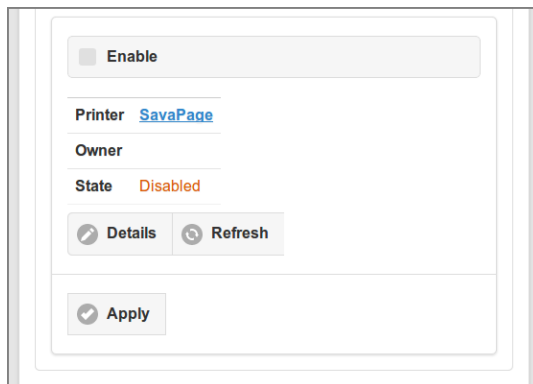


Figure 4.77. Admin Web App: Options - Google Cloud Print - Status

The top panel in this section shows the printer status with the following items:

- **Enable.** A check-box indicating whether the Cloud Printer is enabled or not.
- **Printer.** The name of the Cloud Printer.

¹⁵ <https://www.google.com/cloudprint/learn/>

¹⁶ A list of Web Apps that work with Google Cloud Print can be found at <https://www.google.com/cloudprint/learn/apps.html>.

- **Owner.** The Google User acting as owner of the printer.
- **State.** The state of the printer.

Initially the **Printer** name defaults to *SavaPage*, the **Owner** is *unspecified* and the **State** is *Disabled*.

Tap the name “SavaPage” to set the authenticated Google account.

A new browser tab opens with the Google Cloud Print home page for the authenticated Google User of the current browser session.

*Make sure you are authenticated with the Google account meant for the **Owner** of the SavaPage Cloud Printer.*

When not authenticated Google invites you to *Sign in to continue to Google Cloud Print*. When already authenticated, logout from an existing Google account different from the intended owner, and tap the SavaPage name again.



Note

Although any Google account can be used as owner, we recommend to create a dedicated account to administer the Google Cloud Printer. A personal account is not convenient since it may be deleted or become out-of-date.

*Go back to the SavaPage Admin Web App and press the **Enable** check-box to enable Google Cloud Printing.*

A panel is shown for entering the Google OAuth credentials and Printer name. The credentials are needed by SavaPage to create and monitor the printer belonging to the owner. Although credentials from any Google account other than the one from the printer owner could be used, it is advised to use one and the same account. This track will assume this is the case.



Note

Cloud Ready Printer manufacturers normally use their own OAuth credentials for all printer registrations. For reasons of security and independence SavaPage let you use your own credentials.

*Press the **Apply** button to save the **Enable** setting.*

Tap the link called “Web Application Credentials” to get the OAuth credentials.

This opens a new browser tab with the *Google Developers Console* of the Google account acting as printer owner (as authenticated in the previous step).

If this is a brand new account, follow Google's instructions to get started. When no API project is present, which will be the case for a new account, follow Google's suggestion to create a project.



Warning

Google's web site is subject to change, so instructions below might not exactly fit the labels you encounter. Just follow the logic and hook into the offered dialog.

At the newly created project:

- Select the APIs & auth → Credentials option from the (left hand side) menu.
- Select **Create new Client ID** in the OAuth section.
- Select **Web application** as **Application type** (the other entry fields are irrelevant).

- Press the **Configure consent screen** button.

Google OAuth 2.0
[Web Application Credentials](#)

Client ID

Client secret

Printer

Figure 4.78. Admin Web App: Options - Google Cloud Print - OAuth

Now that the **Client ID for web application** is created, copy the **Client ID** and **Client secret** from the Google console to the corresponding fields in the SavaPage panel.

Press the **Register** button.

A Google Cloud Printer confirmation window will pop-up.

Press the **Finish printer registration** button in the pop-up.

Registration is now complete, and you can close the pop-up window.

Press the **Refresh** button in the SavaPage status panel.

Notice that the **Printer** name and **Owner** have changed according to your registration, and that a new **Online** button has appeared. Press this button to make the printer available for printing (pressing the **Offline** button makes the printer unavailable again).

This finishes the registration of the Google Cloud Ready Printer.



Important

The **Google Cloud Print Service** parameters are stored in the file `/opt/savapage/server/gcp.properties`. Make a backup of this file now, and store it at a secure place, so you can restore it in case of a server crash or when you need to *migrate to a new server*.

4.10.6.2. Edit Google Cloud Printer

The Cloud Printer can be edited and consulted in the Google Cloud Print page, which can simply be accessed by tapping the Printer name in the **Status panel**. Several actions can be performed here like sharing, renaming or deleting the printer.

After a **Rename** of the Cloud Printer, you need to press the **Offline** and **Online** button if you want to see the new name in the Status panel.

A **Delete** of the printer will result in State “Not found” in the **Status panel** (press **Refresh** to update the panel if it does not show). At this point you need to **Register** again if you want to use Google Cloud Print.

You can **Share** the printer by *inviting* other Google users to use it.

4.10.6.3. Google Cloud Print User Registration

For a **Person** to use Google Cloud Print he must have a Google account. This account may be acquired privately, or provided via the Google Apps environment already present in your organization.

The Owner of the Cloud Printer must *share* the printer by *inviting* Google users. See [Section 4.10.6.2, “Edit Google Cloud Printer” \[135\]](#).



Tip

You may share the Cloud Printer with individual users by entering a list of Google email addresses. But you may also share printers with a Google Group. For example, you could set up a dedicated Google Group and share the printer to this group. A Google Group can be set up for users to self-register, but you may chose need to moderate these registrations. Google provides mechanisms for users to request membership to a Google Group and for a moderator to accept or reject those requests.

A SavaPage Administrator must associate the Google account with the right SavaPage User. This is done in the **User Edit** dialog by making sure that the Google account is present as primary or secondary address. For example, John Brown may be known by his primary email address “john.brown@example.com” while one of other email addresses matches his Google account “john.brown@gmail.com”.

Note that the *primary* email address of *external users* is synchronized from the **User Source**, and can be overwritten. So, take care of using the primary email for a Google account, unless you know for sure that the Google account is part of the user source.

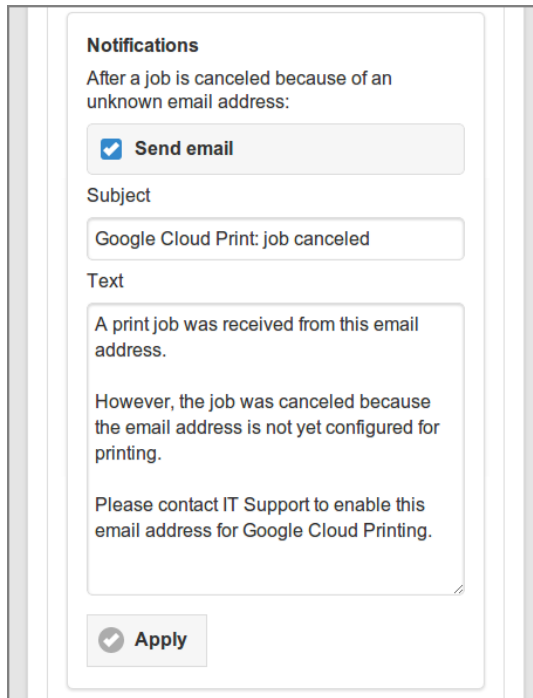


Tip

User email addresses can also be edited with the Server Command Tool. See [Section C.1.19, “setUserProperties” \[263\]](#).

4.10.6.4. User Notifications

In case the associated Google account (email address) of a Google Print Job cannot be matched with a SavaPage user the job is canceled. You can opt to send an email to the user explaining the situation with instructions how to proceed.



The screenshot shows a web interface for configuring notifications. It has a title 'Notifications' and a subtitle 'After a job is canceled because of an unknown email address:'. There is a checkbox labeled 'Send email' which is checked. Below this is a 'Subject' field with the text 'Google Cloud Print: job canceled'. Then there is a 'Text' field containing three paragraphs of text: 'A print job was received from this email address.', 'However, the job was canceled because the email address is not yet configured for printing.', and 'Please contact IT Support to enable this email address for Google Cloud Printing.' At the bottom of the form is an 'Apply' button with a checkmark icon.

Figure 4.79. Admin Web App: Options - Google Cloud Print - Notifications

4.10.7. Mail Print

Mail Print is an implementation of [Driverless Printing](#) which allows users to print documents by mailing them to SavaPage. The email address from the sender is used to find the corresponding [Person](#). See [Section 11.1.14, “Mail Print Authentication”](#) [208].



Note

Mail Print maps to the reserved [Queue](#) /mailprint.

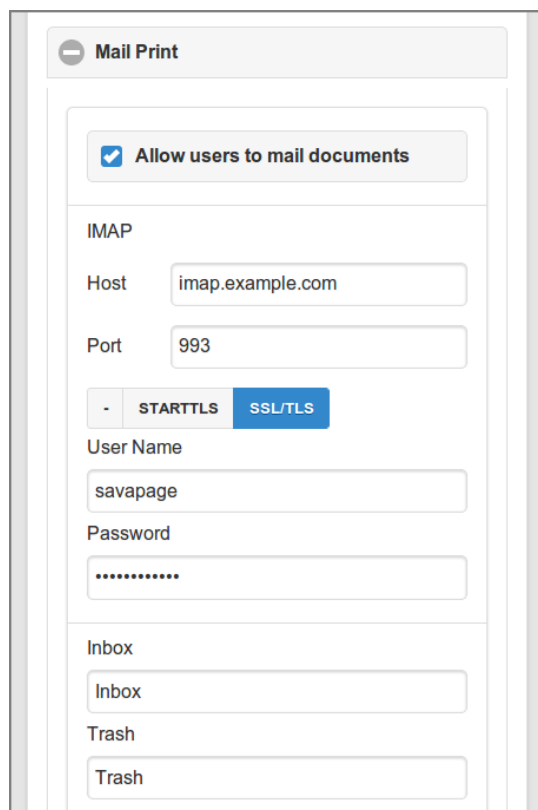


Figure 4.80. Admin Web App: Options - Mail Print (IMAP)

Check the **Allow user to mail documents** to enable the Mail Print function. Then enter the **IMAP** connection parameters:

- The host name or IP address of the **Host**.
- The IP port at **Port**. The standard IMAP ports are: 143 (insecure), 993 (SSL/TLS) and 143 (STARTTLS). The value defaults to 993 (SSL/TLS).
- Select the connection security: - for an insecure connection, and **STARTTLS** or **SSL/TLS** for a secure connection.
- Enter the **User Name** and **Password** for the required authentication.



Important

The IMAP host *must* support the IDLE Command, which is a widely implemented standard extension to the core IMAP protocol. See [RFC2177](https://tools.ietf.org/html/rfc2177)¹⁷.

Print jobs are read from the **Inbox** and moved to the **Trash** folder after processing. Enter the name of both folders:

- **Inbox** : the name of the Inbox folder.
- **Trash** : the name of the Trash folder.

¹⁷ <https://tools.ietf.org/html/rfc2177>

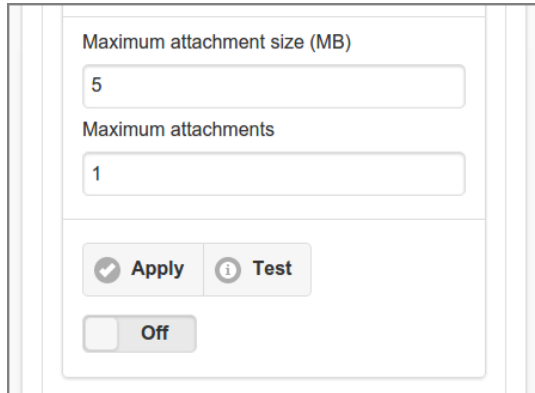


Figure 4.81. Admin Web App: Options - Mail Print (Attachments)

Limit the print job size per email message by setting the **Maximum attachment size (MB)** and **Maximum attachments**. Use integers as value. Leave empty to allow unlimited size.

- Press the **Apply** button to commit the changes.
- Press the **Test** button to test the connection. A feedback message will be displayed with the result.
- Use the flip-switch to turn the Mail Print service **On** and **Off**. Note that after disabling the service it is automatically turned **Off**.



Note

Because Mail Print is an open channel SavaPage does *not* reply to unknown users. This is unlike **Google Cloud Print** notifications, since incoming GCP jobs are from authorized users whose Gmail address is not yet known in SavaPage.

For uploaded file types that do not have a page size defined (HTML, TXT) the **default paper size** is used.

The **Report Font** is used for plain text files (TXT).

4.10.8. Web Print

Web Print is an implementation of **Driverless Printing** which allows users to print documents to SavaPage by simply uploading them from their User Web App. See **Section 3.11, “Upload” [70]**.



Note

Web Print maps to the default **Queue /webprint**.

Figure 4.82. Admin Web App: Options - Web Print

Check the **Allow user to upload documents** to enable the Web Print function. Then enter the restriction parameters:

- Limit the print job size by setting the **Maximum document size (MB)**. Use an integer as value. Leave empty to allow unlimited size.
- Enter IPv4 address ranges as a **CIDR Set** at **IP addresses allowed** to restrict upload based on the requesting IP address. If the field is empty all requesting IP addresses are allowed to upload.
- **Enable Drag & Drop** creates a Drop Zone in the User Web App main view. See [Section 3.11.2, “Upload Drop Zone” \[72\]](#).

4.10.9. Internet Print

Secure **Driver Printing** to SavaPage over public Internet is activated when port 443 of a public IP address is forwarded to port 8632 of the private intranet IP address of the SavaPage server. To authenticate the requesting user a special Printer URI format is used:

```
ipps://[host]/printers/internet/user/[number]/uuid/[uuid]
```

... where [host] is the public DNS name or IP address, and [number] and [uuid] are the ID Number and UUID of the user. See [Section 4.4.4.4, “Card and ID” \[87\]](#), [Section 4.4.4.5, “UUID” \[87\]](#) and [Appendix E, URL Cheat Sheet \[278\]](#).

Figure 4.83. Admin Web App: Options - Internet Print

Enter the protocol://authority of the Internet Printer Device URI as shown to users and press the **Apply** button to commit. When the value is left blank users won't be able to see their private Internet Printer Device URI. See [Section 3.10.1, “Internet Printer”](#) [67].



Important

Internet Print maps to the default [Queue](#) /internet. All print requests over public Internet will have the same remote IP address. To exclusively accept prints from Internet you should set the “IP addresses allowed” to this remote address. See [Section 4.7.3, “Edit Queue”](#) [101].



Caution

Beware that by enabling Internet Print the SavaPage Web Apps are also accessible over public Internet, so take extra care to protect access to these Apps. See [Section 13.2, “Access over Internet”](#) [217].

4.10.10. Proxy Print

Figure 4.84. Admin Web App: Options - Proxy Print General

The **Maximum number of copies per job** restricts the number of *copies* a user can select in the [Print Job Settings](#). Enter a positive number.

The **Maximum number of pages per job** restricts the number of *pages* for proxy print jobs. A proxy print job that exceeds this maximum will be denied. Leave empty to allow unrestricted printing.

To enforce that input documents or pages are deleted after a proxy print, enable **Delete pages after printing**, and select one of the options below. Also see [Section 3.4.4, “Print Job Settings”](#) [42].

- **All documents:** all input documents are deleted.
- **Selected documents:** documents for which pages were printed are deleted.
- **Selected pages:** all pages selected for printing are deleted.
- Check the **Show to user** option to inform the user about the delete scope in the **Print Dialog**.

Check the **Allow Non-Secure Proxy Print** option if you want to allow users to print to any enabled **Proxy Printer** from any device. You can optionally restrict non-secure printer access by entering a **Proxy Printer Group**.

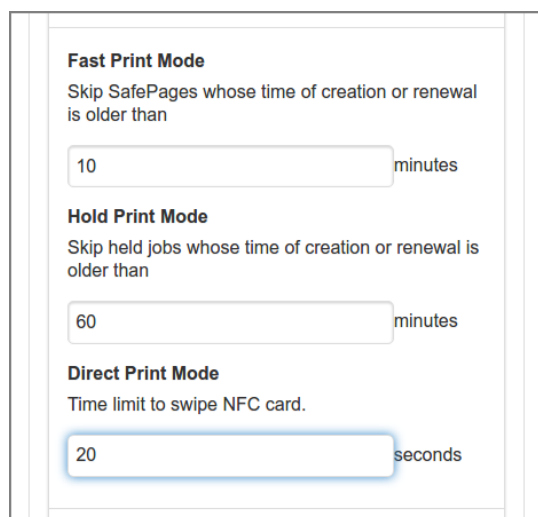
Non-Secure means that users are able to initiate print jobs from locations (desktop, mobile device) remote from the actual printer. This implies that jobs will sit uncollected at the printer, at least for a while. In the mean time, prints containing sensitive information may be read by unauthorized eyes. Or jobs may be forgotten at all, adding up to paper and toner waste.

Any printer that falls beside the non-secure printer pool must be secured by **Terminal** or **Network Card Reader** Devices that have a fixed position at the target printer. See [Section 4.9.1.2, “Proxy Print Authentication” \[114\]](#) and [Section 4.9.2.1, “Custom Proxy Print” \[118\]](#).

Tip for further reading:

- [Appendix A, Proxy Print Scenarios \[247\]](#).

4.10.10.1. Proxy Print Modes



Fast Print Mode
Skip SafePages whose time of creation or renewal is older than
 minutes

Hold Print Mode
Skip held jobs whose time of creation or renewal is older than
 minutes

Direct Print Mode
Time limit to swipe NFC card.
 seconds

Figure 4.85. Admin Web App: Options - Proxy Print Modes

The expiration period for each Print Mode can be entered. See:

- [Section 4.9.1.2.1, “Fast Print Mode” \[115\]](#),
- [Section 4.9.1.2.2, “Hold Print Mode” \[116\]](#)
- [Section 4.9.1.2.3, “Direct Print Mode” \[116\]](#)

4.10.10.2. Proxy Print Delegation

Figure 4.86. Admin Web App: Options - Proxy Print Delegation

In this section you can:

- **Enable Delegated Print** : see [Section 3.4.3, “Delegated Print” \[42\]](#).
- **Integrate Delegated Print with PaperCut** : see [Appendix N, *PaperCut Integration* \[315\]](#).

The **Delegator Invoicing from PaperCut** subsection offers an export of printing cost totals for delegators from selected **Accounts** within a time period export. The result is a CSV file with a line for each delegator. Lines are ordered by user id and specify the cost total within the period and extra data like account and number of transactions per job type, like duplex/simplex,color/grayscale, page format A4, A3, etc. See [PaperCut User Prints](#).

Tip for further reading:

- [Section A.2, “Delegated Print Scenarios” \[247\]](#).

4.10.11. Eco Print

Eco Print is a filter that converts PDF pages to images for eco-friendly proxy printing. The result, including ink and toner savings, is comparable to [Ecofont](#)¹⁸. There is a difference though. While Ecofont uses True Type Font technology at the *start* of the print chain (document editing), SavaPage Eco Print uses bitmap technology at the *end* of the chain. Eco Print intelligently punches holes in *all* non-white areas of the PDF version of a document, just before [proxy printing](#), [downloading](#) or [emailing](#) it.

Since Eco Print processes bitmap patterns it is font agnostic and therefore can handle all font types. And, as an extra, it punches graphics along the way. Contrary to Ecofont, which is a non-free Windows only solution, Eco Print is a Libre solution that works for all client platforms since filtering is performed server side.

¹⁸ <http://www.ecofont.com/>



Warning

The downside of ad-hoc filtering is performance. Eco Print takes about 3 seconds per page (i5 processor, 300 DPI), but is done unobtrusive in the background and need only be done once per PDF document, since the result is cached. Anyhow, Eco Print is not suitable for very large documents.

4.10.11.1. Eco Print Examples

A few Eco Print examples are depicted below.

Plain Print:

Lorem ipsum dolor sit amet,

漓 簞 綳 綰 鹿 格 栢 園 埤,

Eco Print:

Lorem ipsum dolor sit amet,

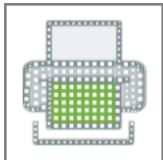
漓 簞 綳 綰 鹿 格 栢 園 埤,

Eco Print magnified:

Lorem ipsum dolor sit amet,

漓 簞 綳 綰 鹿 格 栢 園 埤,

Eco Print Graphics:



4.10.11.2. Eco Print Settings

Figure 4.87. Admin Web App: Options - Eco Print

Check the **Allow users to Eco Print** to enable the Eco Print function. Then specify:

- A **Proxy Printing Discount Percentage** (integer) to be applied to proxy printing costs as specified for any Proxy Printer. See [Section 4.8.2, “Edit Proxy Printer”](#) [105].
- The **Maximum document size (pages) for automatic filtering**. In this example the value of “1” means that any document printed to SavaPage with 1 page is automatically filtered in the background. A value of “3” will automatically filter incoming documents of 3 pages or less. A value of “0” disables this automatism.
- The **Resolution DPI** of the Eco Print page image.

4.10.12. Financial

This section holds the options for [SavaPage Financial](#).

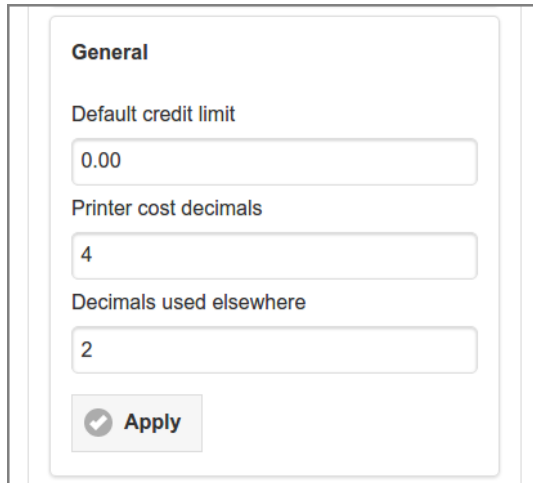
4.10.12.1. Currency Code

Figure 4.88. Admin Web App: Options - Financial - Currency

The [ISO 4217](#)¹⁹ currency code of the financial subsystem can be entered here during installation. When the application status is “ready-to-use” the code can only be changed by using a [Server Command](#). See [Section C.1.4, “changeBase-Currency”](#) [257].

¹⁹ https://www.iso.org/iso/home/standards/currency_codes.htm

4.10.12.2. General Financial Options



The screenshot shows a web form titled "General" with the following fields and values:

- Default credit limit:** 0.00
- Printer cost decimals:** 4
- Decimals used elsewhere:** 2
- Buttons:** A green "Apply" button with a checkmark icon.

Figure 4.89. Admin Web App: Options - Financial - General

General options are:

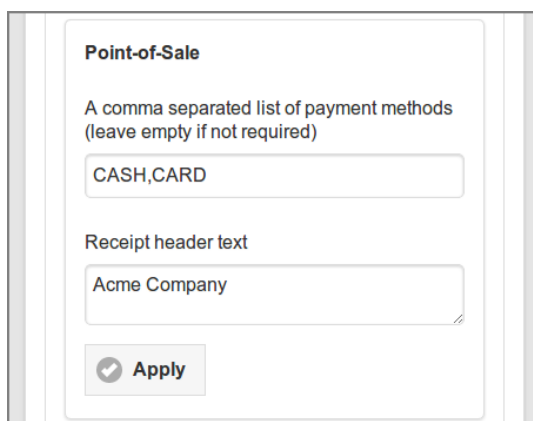
- **Default credit limit:** this is the default value referenced in [Section 4.4.4.6, “Financial” \[87\]](#). The value must be zero or greater.
- **Printer cost decimals:** the number of decimals (max 6) used to specify and display printer costs. See [Section 4.8.2.3, “Media Sources” \[106\]](#) and [Section 4.8.2.4, “Manual Media Sizes” \[108\]](#).
- **Decimals used elsewhere:** the number of decimals (max 6) used to specify financial amounts other than printer costs (e.g. for displaying user account balance).



Note

SavaPage stores financial amounts with a precision of 6 decimals.

4.10.12.3. Point-of-Sale



The screenshot shows a web form titled "Point-of-Sale" with the following fields and values:

- Payment methods:** A text field containing "CASH,CARD" with a description "A comma separated list of payment methods (leave empty if not required)".
- Receipt header text:** A text field containing "Acme Company".
- Buttons:** A green "Apply" button with a checkmark icon.

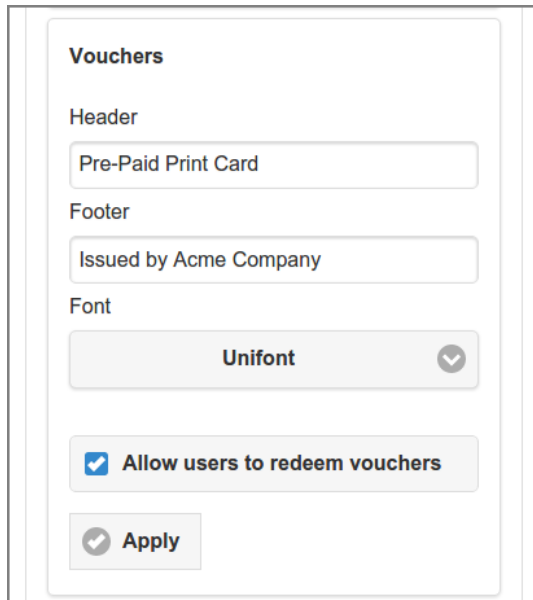
Figure 4.90. Admin Web App: Options - Financial - POS

These are the options for the [Point-of-Sale](#):

- **Payment methods:** see [Section 6.1, “Deposit” \[183\]](#).

- **Receipt header text:** this typically contains a legal text placed in the Receipt header.

4.10.12.4. Vouchers



The screenshot shows a configuration form titled "Vouchers". It contains the following fields and controls:

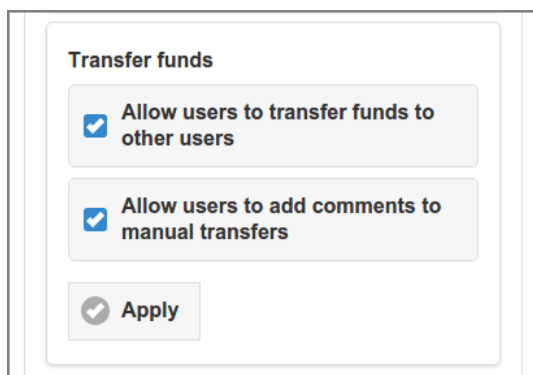
- Header:** A text input field containing "Pre-Paid Print Card".
- Footer:** A text input field containing "Issued by Acme Company".
- Font:** A dropdown menu currently showing "Unifont".
- Allow users to redeem vouchers:** A checkbox that is checked.
- Apply:** A button with a checkmark icon.

Figure 4.91. Admin Web App: Options - Financial - Vouchers

These are the options for the **Voucher System**:

- **Header:** the header text of the voucher card.
- **Footer:** the footer text of the voucher card.
- **Font:** the font used for the PDF Document with vouchers. See [Section 15.2, “Internal Fonts” \[225\]](#).
- You need to explicitly **Allow users to redeem vouchers**.

4.10.12.5. Transfer Funds



The screenshot shows a configuration form titled "Transfer funds". It contains the following fields and controls:

- Allow users to transfer funds to other users:** A checkbox that is checked.
- Allow users to add comments to manual transfers:** A checkbox that is checked.
- Apply:** A button with a checkmark icon.

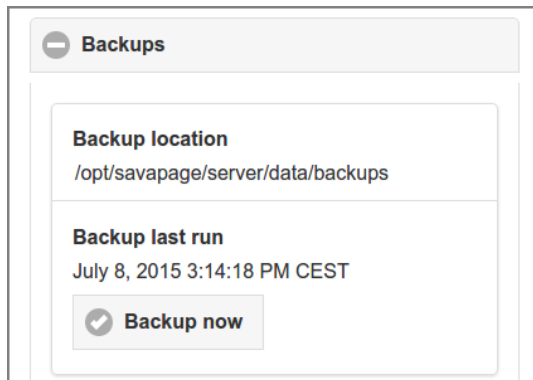
Figure 4.92. Admin Web App: Options - Financial - Transfer funds

These settings apply to **Transfer Credit** dialog in the User Web App. Check the options to **Allow users to transfer funds to other users** and to **Allow users to add comments to manual transfers**.

The minimum and maximum amount to transfer are held in the configuration properties `financial.user.transfers.amount-min` and `financial.user.transfers.amount-max`. They can be changed with the [Configuration Editor](#).

4.10.13. Backups

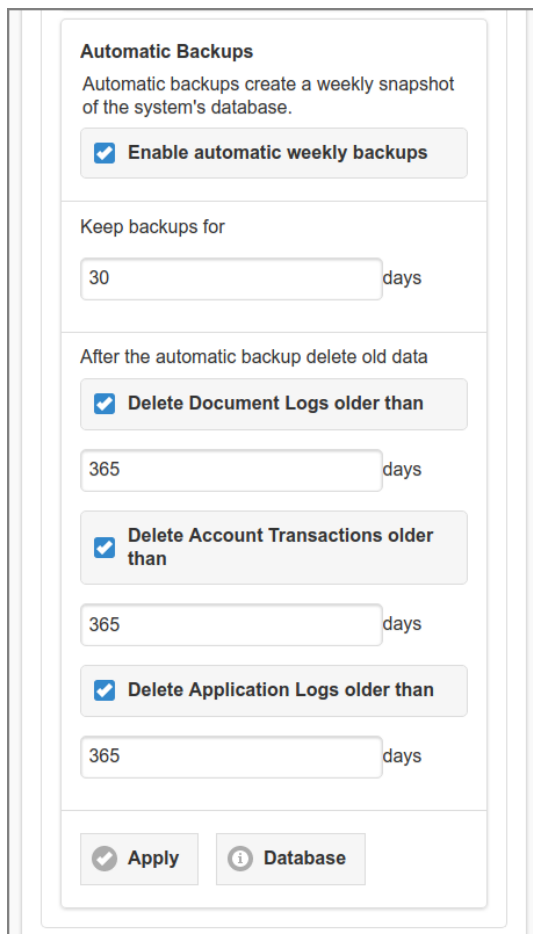
The **Backups** section shows the backup location and time of the last backup.



The screenshot shows a web interface for the 'Backups' section. At the top, there is a header with a minus icon and the text 'Backups'. Below this, there are two main sections. The first section, titled 'Backup location', displays the path '/opt/savapage/server/data/backups'. The second section, titled 'Backup last run', displays the date and time 'July 8, 2015 3:14:18 PM CEST'. At the bottom of the second section, there is a button with a checkmark icon and the text 'Backup now'.

Figure 4.93. Admin Web App: Options - Backups

- Press the **Backup now** button to launch the backup process in the background. The progress and result of the process is not echoed real-time in this section, but can be monitored in the [Real-time Activity](#) section of the [Dashboard](#). Also see [Section C.3.4, “db-export and db-export-to” \[271\]](#).



The screenshot shows a web interface for the 'Automatic Backups' section. At the top, there is a header with the text 'Automatic Backups' and a sub-header 'Automatic backups create a weekly snapshot of the system's database.' Below this, there is a button with a checkmark icon and the text 'Enable automatic weekly backups'. Underneath, there is a section titled 'Keep backups for' with a text input field containing '30' and the text 'days'. Below this, there is a section titled 'After the automatic backup delete old data' with three sub-sections. The first sub-section is 'Delete Document Logs older than' with a checkmark icon and a text input field containing '365' and the text 'days'. The second sub-section is 'Delete Account Transactions older than' with a checkmark icon and a text input field containing '365' and the text 'days'. The third sub-section is 'Delete Application Logs older than' with a checkmark icon and a text input field containing '365' and the text 'days'. At the bottom, there are two buttons: 'Apply' with a checkmark icon and 'Database' with an information icon.

Figure 4.94. Admin Web App: Options - Automatic Backups

The **Automatic Backups** section holds options for creating weekly snapshots of the database.

- Tick the **Enable automatic weekly backups** checkbox to enable the process²⁰.
- The number of **days** a backup should be kept, must be entered at **Keep backups for**.
- A purge of old log data, executed after the backup, can be activated by selecting the **Delete older than** check-boxed for **Application Logs**, **Document Logs** and **Transaction Logs**. Please enter the number of **days** the logs should be held.
- Press the **Apply** button to commit the changes.
- Press the **Database** button to show a pop-up with the number of documents (received, printed, downloaded) and transactions in the database.

4.10.14. Advanced

4.10.14.1. User Client Authentication

The **User Client** uses the system account name of the user to identify itself to the SavaPage server. In a strict **Single Sign-On (SSO)** environment, where a user is already logged in and authenticated as domain user, the system account name can be trusted by default. In environments where non-domain systems are allowed, local accounts are not authenticated by domain services, and therefore can *not* be trusted.

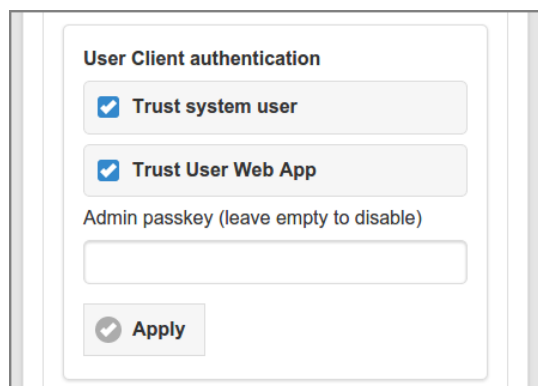


Figure 4.95. Admin Web App: Options - Advanced - User Client

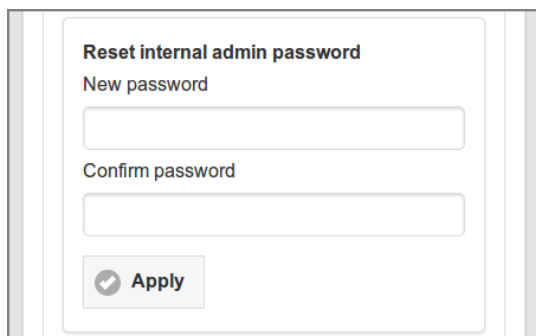
User Client uses the system account name as user identification (unless overridden by a command line option).

- When **Trust system user** is *enabled* the User Client will trust the system account name.
- When **Trust system user** is *disabled* the User Client will pop-up a login dialog to authenticate the user, unless the following trust sources are available:
 - When **Trust User Web App** is *enabled* and the user is already authenticated in a **User Web App** on the same IP address, User Client will trust the Web App user as user identification.
 - When an administrator uses the secret **Admin passkey** in the start-up script it will enforce trust of the offered user identification. See [Chapter 7, User Client](#) [186].
- Press the **Apply** button to commit the change.

4.10.14.2. Admin Password

The **Reset internal admin password** section is the place to change the master password for the built-in **admin account**.

²⁰ Default weekly backups take place at 20 minutes past midnight on Sunday morning, as in the **Cron Trigger Expression** "0 20 0 ? * 1" contained in configuration key `schedule.weekly`. See [Section 4.10.14.10, "Config Editor"](#) [155] on how to change this value.



The screenshot shows a web form titled "Reset internal admin password". It contains two text input fields: "New password" and "Confirm password". Below these fields is a button labeled "Apply" with a checkmark icon.

Figure 4.96. Admin Web App: Options - Advanced - Reset Admin Password

- Enter the new password twice at **New password** and **Confirm password**.
- The password must contain the same minimum number of characters as defined for Internal Users. See [Section 4.4.5, “Create Internal User” \[89\]](#).
- Press the **Apply** button to commit the change.



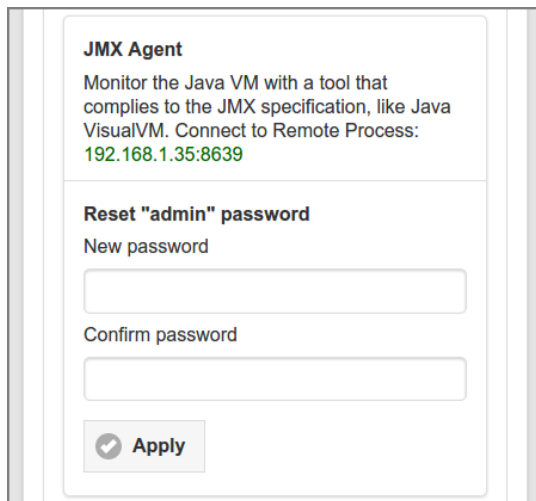
Caution

Keep the new password at a secure place, since it is the master key to your system.

4.10.14.3. JMX Agent

SavaPage runs in a Java Virtual Machine, which has built-in instrumentation that enables client applications to monitor and manage it with the help of Java Management Extensions (JMX). SavaPage configures the built-in JMX agent for *remote monitoring*, so it can be *securely* accessed by remote client management applications, such as Java VisualVM or JConsole.

This section shows the JMX remote process connection string, and enables you to reset the admin connection password.



The screenshot shows a web form titled "JMX Agent". It contains the text: "Monitor the Java VM with a tool that complies to the JMX specification, like Java VisualVM. Connect to Remote Process: 192.168.1.35:8639". Below this text is a section titled "Reset 'admin' password" with two text input fields: "New password" and "Confirm password". Below these fields is a button labeled "Apply" with a checkmark icon.

Figure 4.97. Admin Web App: Options - Advanced - JMX Agent

Java VisualVM is the standard Java JMX client that was first bundled with the Java Development Kit (JDK) version 6, update 7. It can be found in `JDK_HOME/bin`, where `JDK_HOME` is the directory where the JDK is installed.

If `JDK_HOME/bin` is in your system path, you can start Java VisualVM by simply typing **jvisualvm** in a command (shell) prompt. Otherwise, you have to type the full path to the executable file.

Since SavaPage enforces SSL for remote JMX communication, **jvisualvm** needs to be started with two special parameters referring to the Java truststore, holding the trusted SSL certificate, and the truststore password.

The **shared client directory** `/opt/savapage/client/jmx` contains the JMX server certificate `jmxremote.crt`, a ready to use `jmxremote.ts` truststore, and a sample GNU/Linux and Mac shell script `jmxremote.sh` and Windows command file `jmxremote.cmd` to start **jvisualvm** with the right parameters.



Note

The password of the provided `jmxremote.ts` truststore is: `savapage`. Of course you are free to import `jmxremote.crt` into your own truststore and use it with your own password.

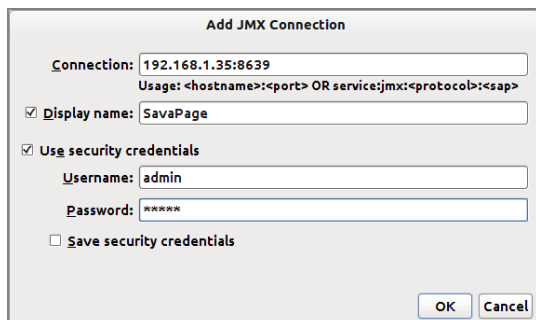


Figure 4.98. Add JMX Connection with Java VisualVM

Add a new JMX Connection and enter the IP address and port number of the **Connection** and as shown in the JMX Agent section, in our case this would be `192.168.1.35:8639`.

Enter the **Username** `admin` and the **Password** as set in the JMX Agent section above. Press the **OK** button to save the connection and start it from the Java VisualVM Applications pane.

Older JDK versions have JConsole as standard JMX client. If you want to use JConsole copy and edit the scripts in `/opt/savapage/client/jmx` so **jconsole** is used instead of the default **jvisualvm**.

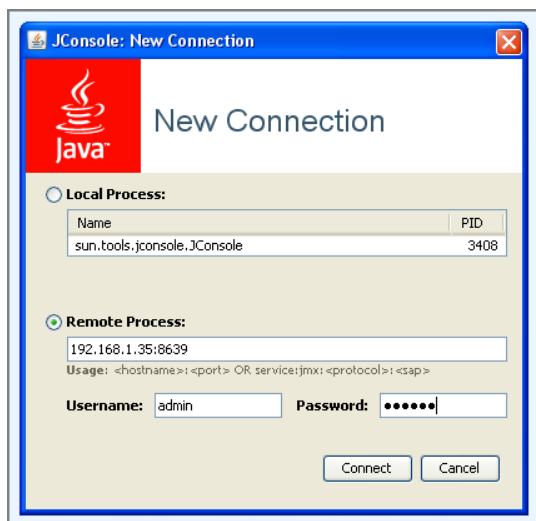


Figure 4.99. Connecting to Remote Process with JConsole

When starting JConsole it prompts you to enter the parameters for the New Connection. Select the **Remote Process** option and enter the IP address and port number as shown in the JMX Agent section, in our case this would be `192.168.1.35:8639`.

Enter the **Username** `admin` and the **Password** as set in the JMX Agent section above. Press the **Connect** button to open the connection.

More information about the JMX configuration can be found in [Section 13.5, “Secured JMX Connection” \[218\]](#).

4.10.14.4. Locale

Enter the System Locale string at the **Locale** section.

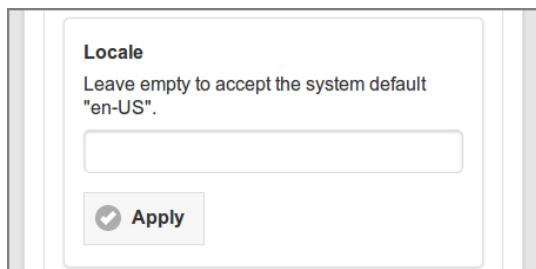


Figure 4.100. Admin Web App: Options - Advanced - Locale

- The format of the locale conforms to IETF BCP 47²¹.
- Some examples are: `en`, `en-GB`, `en-US`, `nl`, `nl-NL`, `nl-BE`.
- You can leave the locale empty to accept the system default.
- The locale is applied to all system messages which are logged in the system log or send by email.
- Press the **Apply** button to commit the change.



Note

This system locale is *not* used for the language and country default used in the Web App. The Web App default is picked up from the locale settings of the Web browser, optionally overruled by the language and country URL parameters. See [Appendix E, URL Cheat Sheet \[278\]](#).

4.10.14.5. Default Paper Size

The **Default Paper Size** is used as the paper size for the printed document of a **Printable File Type** which itself does not have a document structure with a clearly defined page size. These types typically include HTML, TXT and images offered via **Web Print** and **Mail Print**. Choose **Letter** or **A4**, or accept the system **default**.

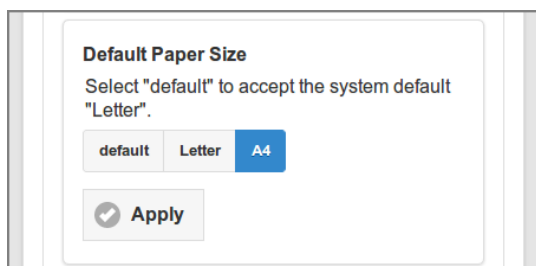


Figure 4.101. Admin Web App: Options - Default Paper Size

See [Section 2.4.1, “Set Default Paper Size” \[15\]](#) on how to set the system default.

4.10.14.6. Report Font

The **Report Font** is used as default font for all PDF reports.

²¹ <https://tools.ietf.org/html/bcp47>



Figure 4.102. Admin Web App: Options - Default Paper Size

See [Section 15.2, “Internal Fonts”](#) [225].

4.10.14.7. Converters

Converters are used to convert files offered for printing via [Web Print](#) or [Mail Print](#) to PDF. This is the place to enable the converters. For installation see:

- [Section G.1.1, “XPS to PDF Installation Instructions”](#) [283]
- [Section G.2, “Advanced File Types”](#) [283]

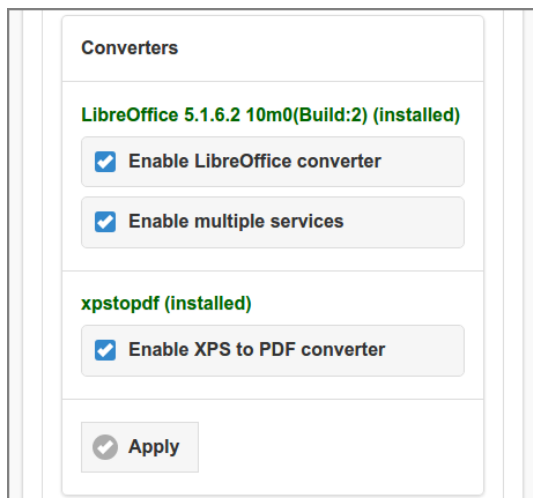


Figure 4.103. Admin Web App: Options - Converters

When [Enable multiple services](#) is checked, the [LibreOffice converter](#) acts as multi-threaded load-balancing service for easy upscaling of conversion throughput. The configuration properties that determine the behavior of this service are summarized in the table below. The defaults will work fine in most situations. By adding extra `soffice.connection.ports` you can enhance conversion throughput, as long as hardware resources permit.



Warning

Tuning LibreOffice configuration values is an advanced task. Please consult your SavaPage Community Representative about which values give the best performance in *your* situation. Then use the [Configuration Editor](#) to change the defaults.

Configuration property	Description
<code>soffice.home</code>	The LibreOffice home location. When empty, a probe to likely candidates is performed to retrieve the location. Default: empty.
<code>soffice.profile.template-dir</code>	When empty, a temporary profile directory is created for each UNO connection process with its own defaults settings. Otherwise, this configuration property must provide a profile directory

Configuration property	Description
	containing customized settings. This template directory will be copied to the temporary profile. Default: empty.
soffice.connection.ports	A comma/space separated list of TCP/IP ports to localhost LibreOffice (UNO) connection instances to be launched by SavaPage. Default: 2002,2003
soffice.connection.restart-task-count	The number of executed tasks after which a UNO connection is restarted. When 0 (zero) the connection is <i>never</i> restarted. Default: 200
soffice.task.queue-timeout-msec	Wait time (milliseconds) for a UNO connection to become available for task execution. Default: 10000
soffice.task.exec-timeout-msec	Wait time (milliseconds) for a conversion task to complete. Default: 20000
soffice.respond.retry-msec	Retry interval (milliseconds) for host process to respond. Default: 250
soffice.respond.timeout-msec	Wait time (milliseconds) for host process to respond (after retries). Default: 30000
soffice.start.retry-msec	Retry interval (milliseconds) for host process to start. Default: 1000
soffice.start.timeout-msec	Wait time (milliseconds) for host process to start (after retries). Default: 120000

Table 4.3. LibreOffice Configuration Properties

4.10.14.8. SafePages

This section contains advanced options regarding encrypted PDF and the expiration of **SafePages** input documents.

SafePages

This option allows Encrypted PDF documents to be printed to SafePages for Proxy Printing. Exporting these SafePages as PDF is not allowed.

☒ Allow Encrypted PDF for Proxy Printing

These options determine the life cycle of input documents.

☐ Delete documents at WebApp logout

Document expiration time (enter zero to disable)

minutes

Expiration time signal (enter zero to disable)

minutes

☒ Apply

Figure 4.104. Admin Web App: Options - Advanced - Proxy Printing

- **Allow Encrypted PDF for Proxy Printing** holds the policy to as described in [Section 10.7, “Printing Encrypted PDF” \[204\]](#). The option is enabled by default. Disable it if you do not endorse the policy: this will ignore every print SavaPage job request holding an encrypted PDF document.
- When **Delete documents at Web App logout** is checked all print-in documents are deleted when the users logs out.
- **Document expiration time** manages the input document life cycle. Any document older than the number of entered minutes is considered expired and will be automatically deleted. For instance, a value of 1440 will delete the SafePages document 24 hours after it was printed. The expiration time is shown in the [Document Details](#) dialog. The user is [notified by pop-up](#) after an expired document is auto-deleted. User action is required to close the pop-up. This way we are sure the user noticed the delete and his expectation is set right. When a user logs out and logs in again after some time, expired documents will be auto-deleted to begin with, but the user will *not* be notified of this event.
- Use the **Expiration time signal** value to signal the user when expiration is due. For instance, a value of 15 will mark the [document thumbnails](#) with a clock icon in a colored (orange) footer, 15 minutes before expiration. This will alert the user, so he can do some last minute actions on old documents.
- Press the **Apply** button to commit the changes.

4.10.14.9. Pagometers

In analogy with the term *Odometer*, the term *Pagometer* is coined as an instrument to count the number of processed pages of SavaPage input and output documents. Pagometers are used to display usage statistics and [Printing Impact](#) from a global viewpoint as in the [Dashboard](#), or in specialized views for [User](#) and [Users](#), [Queues](#) and [Proxy Printers](#). The counters can be reset in the **Reset Pagometers** section.

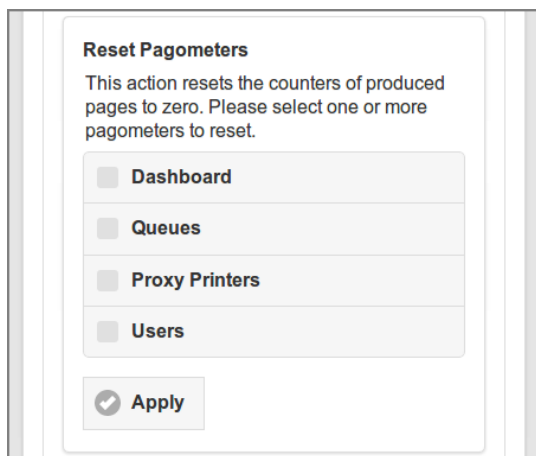


Figure 4.105. Admin Web App: Options - Advanced - Pagometers

- Tick the checkboxes of the pagometers to reset.
- Press the **Apply** button to execute the action.

4.10.14.10. Config Editor

Most of the SavaPage configuration properties can be edited in dedicated sections of the Admin Web App. However, many extra properties are present without an editing interface. Luckily a generic Configuration Editor is available for editing individual configuration properties, so the defaults of "hidden" properties can be changed when needed.



Warning

If you use the Config Editor incorrectly, you may cause serious problems which can only be fixed by re-installation of the application. Use the Config Editor at your own risk.

Tap the **Configuration Editor (advanced)** button to start the editor. See [Figure 4.106, “Admin Web App: Configuration Editor - List” \[156\]](#) for a detailed description.

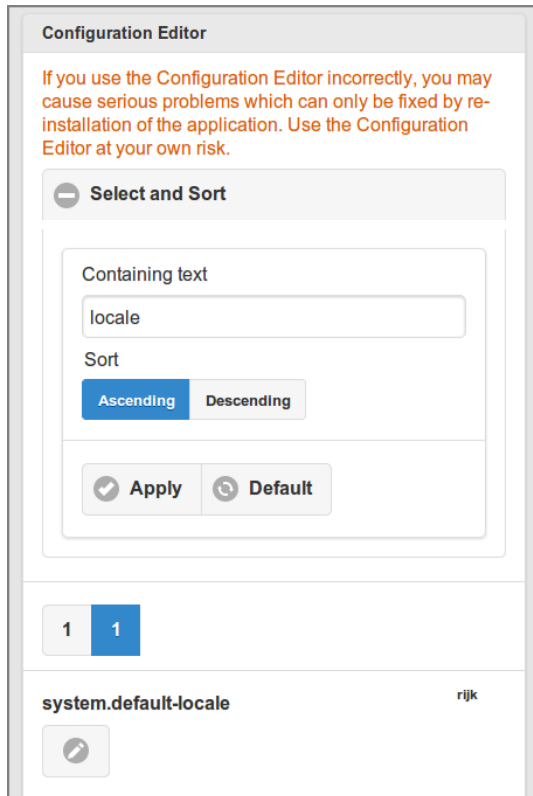


Figure 4.106. Admin Web App: Configuration Editor - List

- All configuration properties are listed alphabetically by default with their name and value. Secret values are encrypted and shown as `*****` in the list, see [Section 13.6, “Encrypted Secrets” \[218\]](#).
- Push the **Select and Sort** button to expand (collapse) the section.
- The list can be traversed by tapping one of the pager at the top or bottom of the page.
- Select items by entering the containing text (fragment) of their name. So, entering "ldap" will select "auth.ldap.port" and "ldap.schema.group-member-field".
- The list can be sorted **Ascending** or **Descending** on name.
- Tap the **Apply** button to (re)display the list.
- A tap on the **Default** button resets the selection and sort field to their default values.
- Tap the **Edit** button to edit the item. See [Figure 4.107, “Admin Web App: Configuration Property - Edit” \[156\]](#).



Note

Due to [Admin Privileges](#) the **Edit** button might *not* be visible.

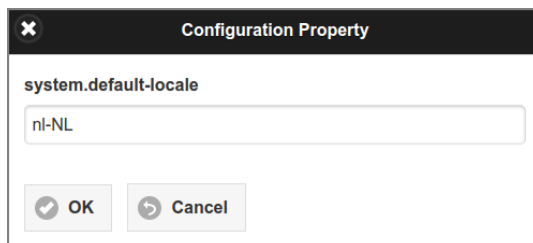


Figure 4.107. Admin Web App: Configuration Property - Edit

- The value of the item is shown in the entry field and can be edited. Secret values are shown decrypted.
- Press the **OK** button to commit the change and return to the list.
- The **Cancel** button brings you back to List without changing anything.

4.11. Documents

This panel is shown after:

- A tap on the **Documents** button in the **Main Menu** : all processed documents are shown.
- A tap on the **Log** button in the **User List**: all documents processed by the selected user are shown. The user's name is shown in the header and the **Select and Sort** is within the scope of this user.
- A tap on the **Log** button in the **Queues List**: the **Select and Sort** is initialized with, and applied for the selected Queue.
- A tap on the **Log** button in the **Proxy Printers List**: the **Select and Sort** is initialized with, and applied for the selected Proxy Printer.

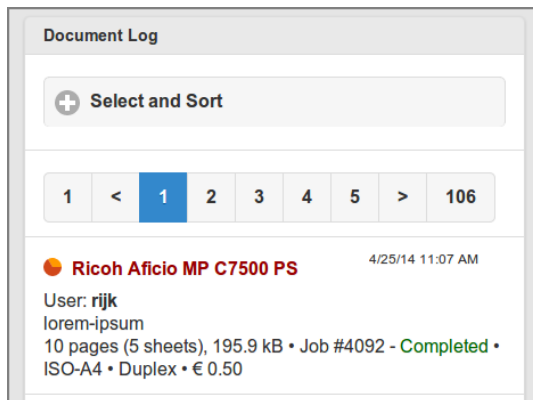
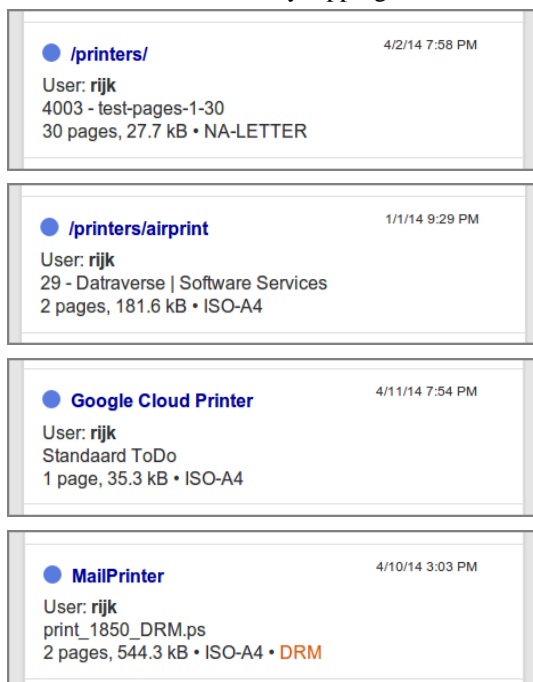


Figure 4.108. Admin Web App: Documents - List

The list can be traversed by tapping one of the buttons at the top or bottom of the page.





Each document is displayed with data depending on its input source (SavaPage Print **Queue**) or output target (**PDF** export, **Proxy Printer** print). From top to bottom:

- The creation date at the top right corner.
- Source or destination, shown in a color depending on its type.
 - A SavaPage Queue, like `"/printers/"`, is displayed in *blue* (when RAW printed to the default Queue the word “Printer” is displayed, driverless printing shows “WebPrinter” or “MailPrinter”).
 - A PDF is shown in *green*.
 - A Proxy Printer, like “Ricoh Aficio MP C7500”, is displayed in *red* prefixed with an inline pagometer Pie-Chart. The *red* color in the chart represents the number of *pages* in the job. The *orange* color represents the number of printed *sheets*.
- User: the user id as creator of the document.
- The document name, with optionally (PDF export) the PDF author name, subject and keywords.
- The number of pages and size (bytes). With optionally...
 - The CUPS Job number (Proxy Printer only).
 - The CUPS printing status: *Pending, Held, Processing, Stopped, Canceled, Aborted, Completed* (Proxy Printer only).
 - The paper size, like “ISO-A4” (Queue and Proxy Printer only).
 - “LH” indicator in case a letterhead was applied (Proxy Printer and PDF only).
 - “Duplex” indicator (Proxy Printer only).
 - “DRM” indicator when exported PDF was encrypted (PDF only), or printed document was an encrypted PDF. (Queue only).
 - “Denied” indicator when printed document was an encrypted PDF and such printing is not permitted (Queue only). See [Section 10.7, “Printing Encrypted PDF” \[204\]](#).
 - Owner (“O”) and User (“U”) password indicators (PDF only).
- Destination (PDF only). The client IP address (**PDF Download**) or the recipient email address (**Send PDF**)

Tap the **Select and Sort** button to expand the section, and select a document **Type** :

- Option **-** selects all document types: **more...**
- Option **In** selects documents printed to a SavaPage Queue: **more...**
- Option **Out** selects documents printed to a Proxy Printer or exported to PDF: **more...**
- Option **PDF** selects documents exported to PDF: **more...**
- Option **Print** selects documents printed to a Proxy Printer: **more...**

Depending on the selected **Type**, selection and sort options are shown or hided. However, there are common selections for all document types as discussed at the [screenshot below](#).

Document Log

Select and Sort

Type: **-** In Out PDF Print Ticket

Document Name:

Period

From:

To:

Sort: **Date** Name Ascending Descending

☒ Apply ☐ Default

Figure 4.109. Admin Web App: Documents - Select and Sort - All

- Select a **Document Name** by entering a name part (fragment).
- Select a creation **Period** by entering a **From** and **To** date. Tap the **x** button after a date to clear it. See this example [Data Selection Dialog](#).
- Documents can sorted **Ascending** or **Descending** by creation **Date** or **Name**.

Select and Sort

Type: **In** Out PDF Print Ticket

Document Name:

Period

From:

To:

Queue: ☒

Sort: **Descending**

- all -
/
/airprint
/gcp
/internet
/mailprint
/raw
/smartschool
/webprint

Figure 4.110. Admin Web App: Documents - Select and Sort - In

As an extra to the **common options**, the **In Type** offers:

- A selection on **Queue**.
- A **Sort** on **Queue**.

The screenshot shows a web form titled "Select and Sort" with a minus icon on the left. The form is divided into several sections. The first section, labeled "Type", contains a row of buttons: "-", "In", "Out" (highlighted in blue), "PDF", "Print", and "Ticket". Below this is a "Document Name" text input field. The next section, labeled "Period", contains "From" and "To" text input fields. Below that is a "Destination" text input field. The "Letterhead" section contains a row of buttons: "-", "Yes", and "No", with "Yes" highlighted in blue. The final section, labeled "Sort", contains a row of buttons: "Date", "Name", "Ascending", and "Descending", with "Descending" highlighted in blue.

Figure 4.111. Admin Web App: Documents - Select and Sort - Out

As an extra to the **common options**, the **Out Type** offers:

- A selection on **Destination**.
- A selection on **Letterhead**.

The screenshot shows a web form titled "Select and Sort" with a minus icon on the left. The form is divided into several sections. The first section contains "Author", "Subject", and "Keywords" text input fields. Below these is an "Encryption" section with a row of buttons: "-", "Yes", and "No", with "Yes" highlighted in blue. The next section contains "User password" and "Owner password" text input fields. The final section, labeled "Sort", contains a row of buttons: "Date", "Name", "Ascending", and "Descending", with "Descending" highlighted in blue.

Figure 4.112. Admin Web App: Documents - Select and Sort - PDF

As an extra to the **Out options**, the **PDF Type** offers:

- A selection on **Author**.

- A selection on **Subject**.
- A selection on **Keywords**.
- A selection on **Encryption**.
- A selection on **User password**.
- A selection on **Owner password**.

Figure 4.113. Admin Web App: Documents - Select and Sort - Print

As an extra to the **Out options**, the **Print Type** offers:

- A selection on **Printer**.
- A selection on **Job State**.
- A selection on **Layout**.
- A **Sort** on **Printer**.

Figure 4.114. Admin Web App: Documents - Select and Sort - Ticket

The **Ticket Type** shows a slightly changed version of the **Print Type options**:

- A selection on **Ticket** is added (at the expense of the Printer selection). You can enter any part of a Ticket Number.
- Selection on **Layout** is dropped.

4.12. Log

After a *tap* on the **Log** button in the main menu this panel is shown. See [Section 4.2, “Menu”](#) [74].

The Log shows Info, Warning or Error messages for application events.



Tip

The size of the Log can be limited by purging old log data after an automatic database backup. See [Figure 4.94, “Admin Web App: Options - Automatic Backups”](#) [148].

Application Log

– Select and Sort

Containing text

Level

– Info Warning Error

Period

From

To

Sort

Date Level Ascending Descending

✓ Apply ⌂ Default

Figure 4.115. Admin Web App: Log - List

- All events are listed by default, ordered by descending date.
- The list can be traversed by tapping one of the buttons at the pager at the top or bottom of the page.
- A different selection and sorting can be entered: see [Figure 4.116, “Admin Web App: Log - Select and Sort” \[162\]](#).

Application Log

+ Select and Sort

1 < 1 2 3 4 5 >

251

Info 4/23/14 9:43 AM
Database backup successfully written to savapage-export-2014-04-23T09-43-50.zip

Info 4/23/14 9:32 AM
Cleaned up 1 input document.

Info 4/23/14 9:32 AM
Cleaned up 1 output document.

Figure 4.116. Admin Web App: Log - Select and Sort

- Events can be selected by entering the **Containing text** (fragment) of the message.

- Select the event **Level**. The **-** button will select all levels.
- Select the time **Period** by pushing the **From** and **To** button. See date picker dialog is shown at [Figure 4.117](#), “Admin Web App: Log - Select Date” [163]. Tap the **x** button after the date to clear it.
- The list can be sorted **Ascending** or **Descending** on **Date** or **Level**.
- Tap the **Apply** button to (re)display the list.
- A tap on the **Default** button resets the selection and sort fields to their default values.

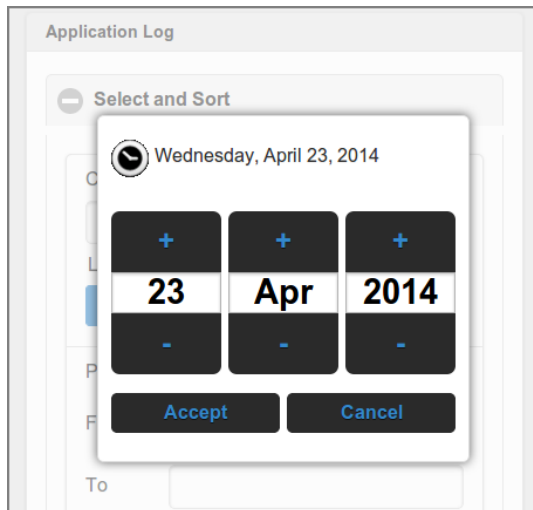


Figure 4.117. Admin Web App: Log - Select Date

- Use the **+** and **-** buttons to select the Month, Day and Year.
- Push the **Accept** button to apply the date.
- Push the **Cancel** button to ignore the date and return to the original setting.

4.13. About

After a tap on the **About** button in the main menu this panel is shown. See [Section 4.2, “Menu”](#) [74].

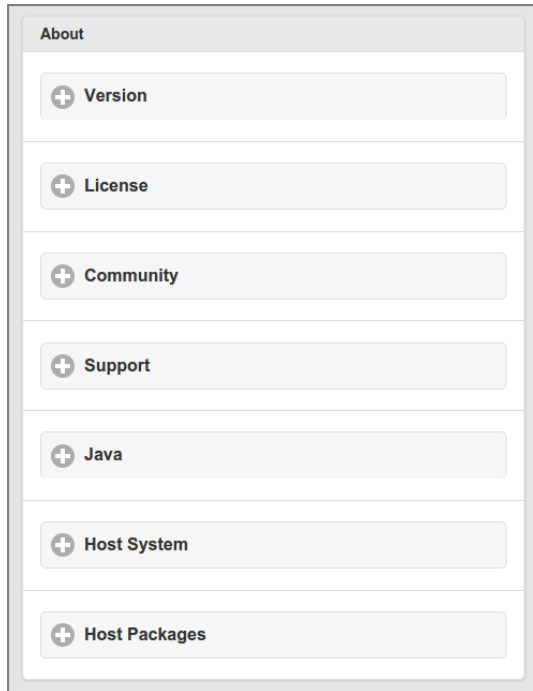


Figure 4.118. Admin Web App: About

A *tap* on one of the options expands (or collapses) the underlying section. The sections are described in the paragraphs below.

4.13.1. Version

The **Version Info** identifies the application and database version. Please include this information when you contact support.

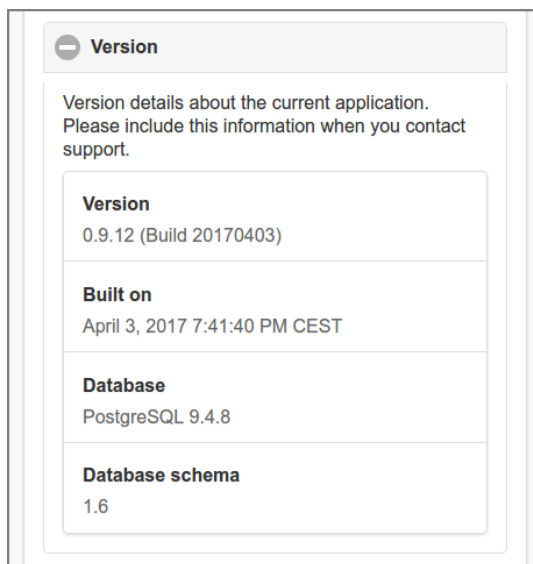


Figure 4.119. Admin Web App: About - Version

4.13.2. License

This section displays the license information with related links in green.

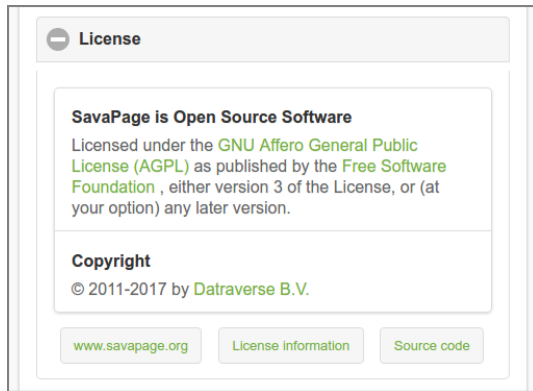


Figure 4.120. Admin Web App: About - License

4.13.3. Community

The **Community** section gives all the information about your Community Membership. See Figure 4.4, “Admin Web App: Dashboard - Status” [77] for a summary of Membership Status values. See Chapter 19, *SavaPage Community* [245] for an explanation about SavaPage Membership in general.

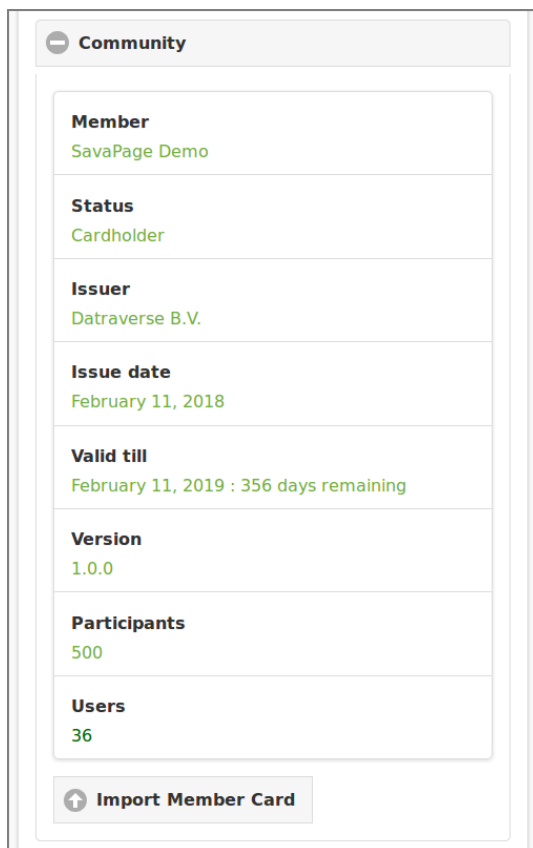


Figure 4.121. Admin Web App: About - Community

Press the **Import Member Card** button to start the **Import Dialog**.

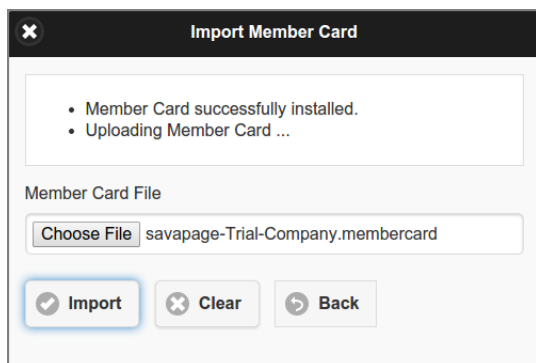


Figure 4.122. Admin Web App: About - Import Member Card

- Select the Member Card file to be uploaded. The actual file selection trigger differs from browser to browser. The screenshot above is from the Chromium browser.
- Press the **Import** button to start the import.
- The progress of the import is displayed at the top of the dialog box.
- The **Back** button brings you back to the Community section.
- Just in case, the **Clear** button clears the messages and selected file.



Note

Due to **Admin Privileges** the **Import Member Card** button might *not* be visible.



Warning

The Member Card Import dialog is tested with Chromium, Firefox and Internet Explorer. There are some known issues with other browsers:

- The Opera 12.11 browser does *not* display the status of the import process. The import is performed correctly though.

Please contact us when you encounter any problems with your browser.

4.13.4. Support

The **Support** section shows the addresses for online information and the Help Desk and offers download links for the **SAVAPAGE.ppd** file.

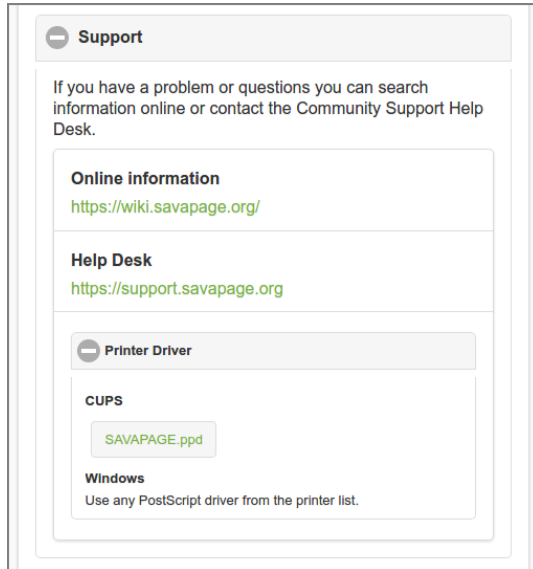


Figure 4.123. Admin Web App: About - Support

4.13.5. Java

This section displays the Java Runtime version, number of processors and max memory available.

4.13.6. Host System

This section displays name, version and architecture of the host operating system and the GNU/Linux parameters described in [Section 18.1, “Linux Kernel Parameters” \[239\]](#) and [Section 18.2, “Linux User Limits” \[241\]](#).

4.13.7. Host Packages

The **Host Packages** gives version information about the required and optional third-party software installed on the SavaPage host.

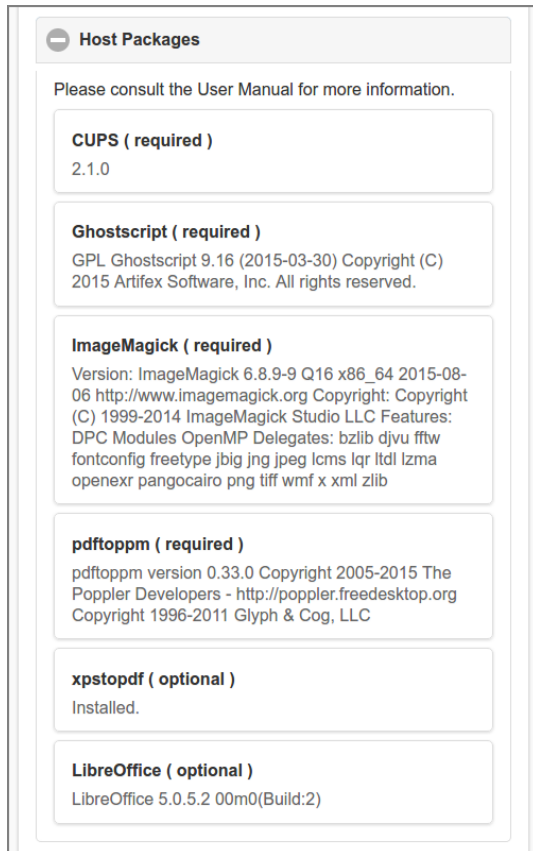


Figure 4.124. Admin Web App: About - Host Packages



Note

Before **xptopdf** and LibreOffice can be used they must be *enabled*. See Figure 4.103, “Admin Web App: Options - Converters” [153].

4.14. Vouchers



Vouchers provide a straightforward and cost effective solution for users to upgrade their account balance. Vouchers are common value tokens in many applications, like for instance pre-paid mobile phones. Unlike solutions that use smart cards, micro-payments or vending machines, voucher systems require no hardware investment. While manual processing is needed to generate, print, distribute and sell the voucher cards, redemption is fully end-user driven and can be processed automatically.

A voucher system is fully integrated in SavaPage, and includes:

- A Web App dialog for administrators to **Create Vouchers**.
- A Web App dialog for end-users to **Redeem Vouchers**.
- A **Voucher Security** safety net for voucher tracking and fraud prevention.

Vouchers

Batch

B-1404

+ Actions

+ Select and Sort

✓ Apply

↺ Default

1

<

1

2

3

4

5

>

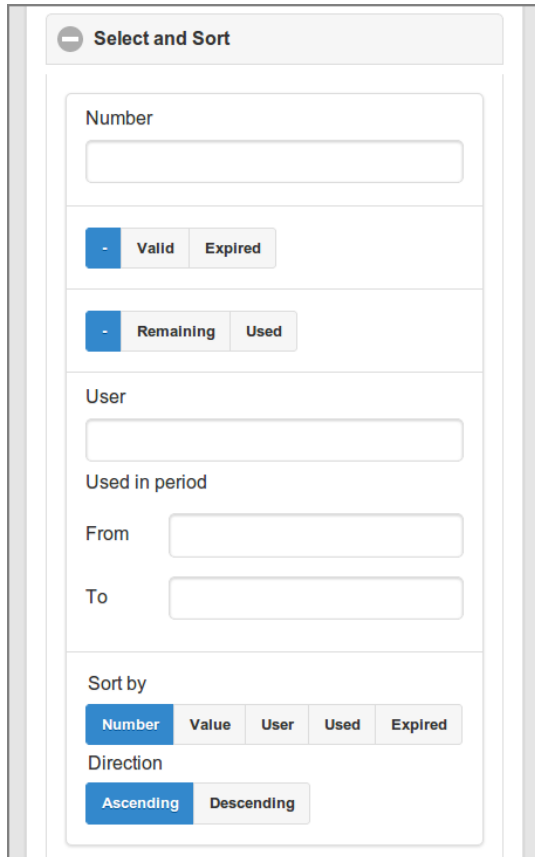
5

B-1404-0021-0661-4775-7916	used
April 25, 2014 1:40:51 PM • rijk	
B-1404-0124-4932-1157-0751	valid
B-1404-0128-3386-0943-7590	valid
B-1404-0296-8174-9238-9160	valid

Figure 4.125. Admin Web App: Voucher List

The list shows the vouchers of a selected Batch-ID and the **extra selections** shown in the next figure.

- The list is refreshed, and the selection applied, after you push the **Apply** button.
- The **Default** button resets the selection items to their default values.



Select and Sort

Number

☒ Valid ☐ Expired

☒ Remaining ☐ Used

User

Used in period

From

To

Sort by

Number Value User Used Expired

Direction

Ascending Descending

Figure 4.126. Admin Web App: Vouchers - Select and Sort

Select vouchers by entering:

- Part of their **Number**.
- Their **Valid** or **Expired** status.
- Their **Remaining** or **Used** status.
- Part of the **User** ID that redeemed any voucher.
- The **From** and **To** date of their usage period.

4.14.1. Voucher Actions



Note

Due to **Admin Privileges** this section might *not* be visible.

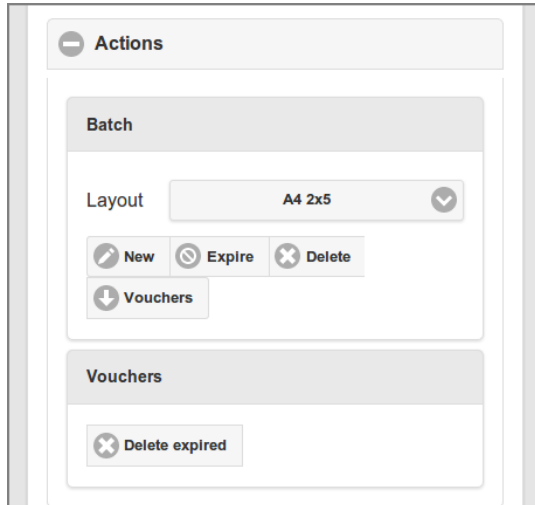


Figure 4.127. Admin Web App: Voucher Actions

- **New** : pops up the dialog to **Create Vouchers**.
- **Delete expired** : deletes vouchers whose expiry date is before the current date (today).

When a Batch-ID is selected extra buttons appear:

- **Expire** : Expires all remaining vouchers of the selected Batch-ID by setting the expiry date to today 00:00.
- **Delete** : Deletes all remaining vouchers of the selected Batch-ID.
- **Vouchers** : downloads a printable PDF document with remaining (non-redeemed) vouchers of the selected Batch-ID according to the selected **Layout**.

4.14.2. Create Vouchers

Create Vouchers

Batch

Allocate a unique ID to this batch of vouchers.

Batch-ID

Vouchers

Select the number of vouchers, their value and expiration.

Number

Value

Expiry Date

Design

Select the layout for the voucher print.

Layout

Figure 4.128. Admin Web App: Create Vouchers

Enter the data for the batch as follows:

- **Batch-ID:** a user defined ID that will be assigned to all vouchers in a batch. The ID is prefixed to each voucher number to easily identify its source. A unique ID should be assigned to each batch.
- **Number:** the number of vouchers in the batch.
- **Value:** the monetary value of each voucher.
- **Expiry Date:** the date after which a voucher can no longer be used. This enforces that vouchers are valid for a limited period of time.
- **Layout:** the page format of the PDF output with the number of voucher columns and rows. Some fixed variants are offered.

Press the **Vouchers** button to create the batch. As a result:

- Each voucher in the batch is assigned a formatted random unique number, for example B-1404-0021-0661-4775-7916, and is stored in the database.
- A printable PDF document is downloaded with all vouchers from the batch according to the selected **Layout**.

4.14.3. Voucher Usage

This is what end-users should know about vouchers.

- Purchase a voucher from an authorized person at an assigned location. Vouchers are unique for your organization and cannot be used elsewhere.

- Use a web browser to open the SavaPage User Web App. After logging in, your current account balance is shown at the **footer bar**.
- Push the account balance button to pop-up the **User Details** dialog, and push the **Redeem Voucher** button in the pop-up.
- Enter the voucher Number in the next dialog box and press **Redeem**. Make sure to enter the number exactly as listed on the voucher including any dashes (-).
- If you entered the number correctly, the value as shown on the voucher will be transferred to your account and a new entry will list in your **transaction log**.

Chapter 5. Job Tickets Web App



The Job Tickets Web App can be accessed by users with role **Job Ticket Operator** to dispatch job tickets. Job Tickets are created by users with role **Job Ticket Creator** in the **Print Dialog** of the User Web App. Operators optionally add or edit print options and select a suitable proxy printer to print the job.

The Web App can be reached at <https://savapage:8632/jobtickets>. See [Appendix E, URL Cheat Sheet \[278\]](#).

5.1. Open Tickets

5.1.1. Select and Sort

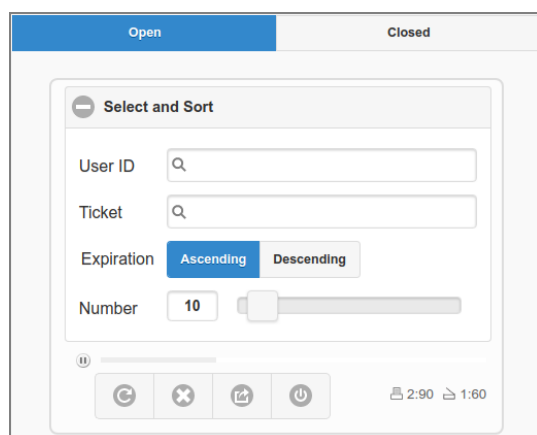


Figure 5.1. Job Tickets: Open Tickets - Select and Sort

A list of pending Job Tickets is shown that refreshes automatically every 60 seconds. The tiny progress bar at the bottom of the **Select and Sort** section shows when the refresh is due.

- You can pause/resume the refresh by toggling the tiny button at the left of the progress bar.
- You can Quick Search Job Tickets for a single **User ID**. Leave empty to select all users.
- You can Quick Search for a single **Ticket ID**. When a User ID is selected, tickets of just this user are shown. Leave empty to select all tickets.
- Tickets can be sorted **Ascending** or **Descending** on expiration time.
- The **Number** of tickets shown in the list is limited, and can be controlled with a slider. The slider range and default can be set with [configuration properties](#).
- At the bottom right, a small printer and copier icon is shown with the total number of pending print/copy tickets and number of sheets to be printed.

5.1.2. Ticket List

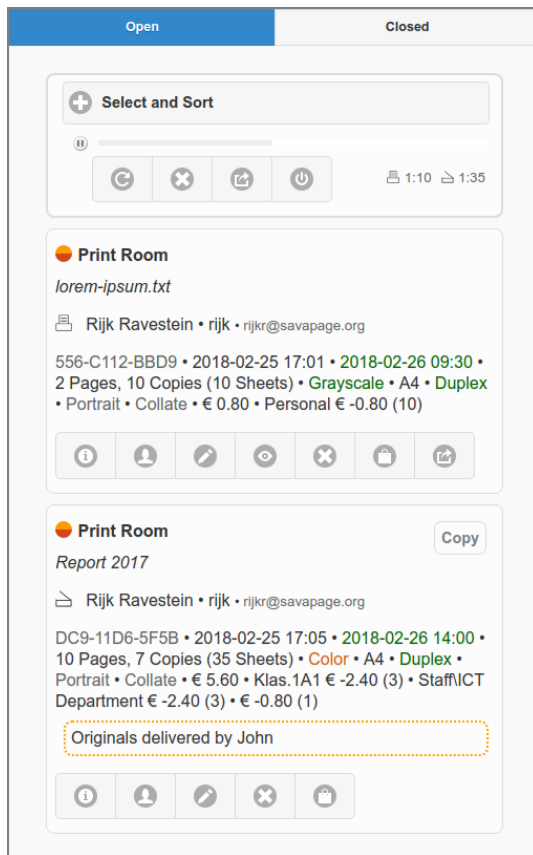


Figure 5.2. Job Tickets: Open Tickets - List

- Each Job Ticket in the list shows details and has buttons for further actions. From left to right:
 - **Settings...** shows a dialog with a summary of printer settings.
 - **Transactions...** shows a dialog with financial transaction details.
 - **Edit...** shows the **Edit Dialog**, so job options can be changed.
 - **Preview** downloads the PDF document to be printed.
 - **Cancel...** shows a dialog to confirm cancellation of the Ticket. A reason can be entered. After cancellation the user is optionally notified by email: see **Job Ticket Configuration Properties**.
 - **Settle...** shows the **Settle Dialog**, so the job can be charged, when not printed from the **Print Dialog**, but from the PDF viewer, after **Preview**.
 - **Print...** shows a **Print Dialog** with compatible printers to redirect the job to.



Note

When pop-up dialogs are visible, automatic list refresh is paused.

5.1.3. Job Ticket Bulk Actions

The *scope* of the **Cancel All...** and **Print All...** is the *visible* list selection. This bulk action must be confirmed in a pop-up.

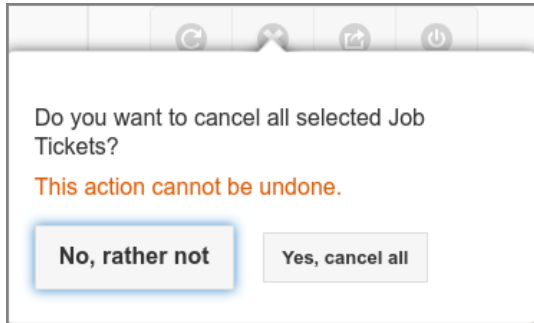


Figure 5.3. Job Tickets: Cancel All

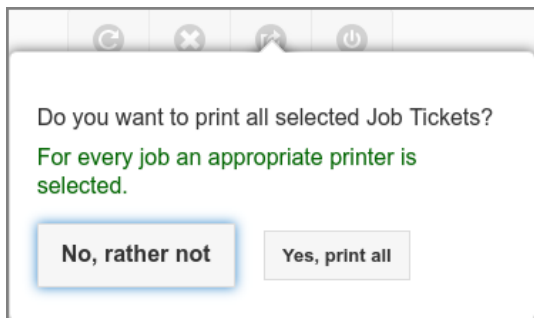


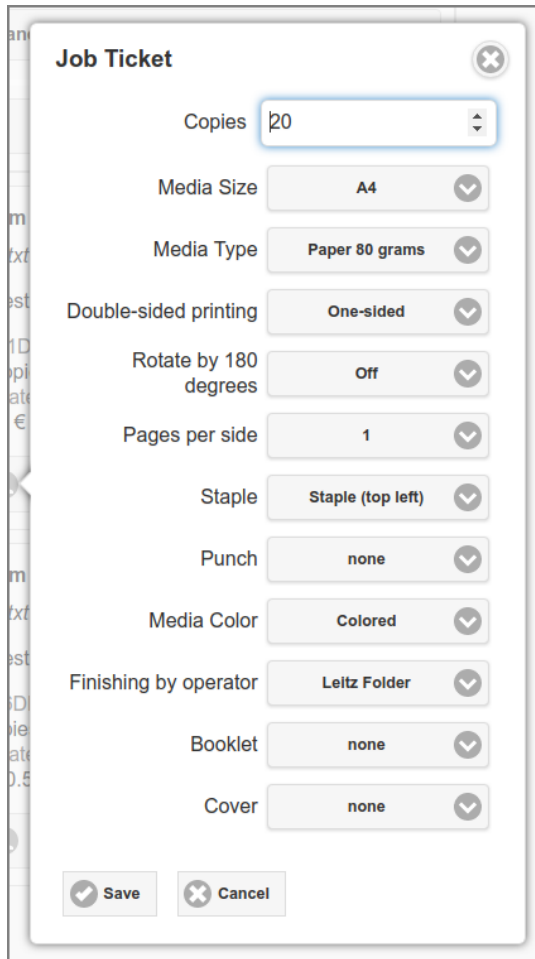
Figure 5.4. Job Tickets: Print All

The **Print All...** action selects a random compatible printer from the **Job Ticket Proxy Printer Group** for each Job Ticket.

- The selected printer *must* support automatic media source selection, since this option will be used for the print job. In this way, media (implicitly) selected through the job ticket media-source, will be printed from the intended tray.
- The **Print All...** action is enabled by default and can be disabled by setting a **configuration property**.

5.1.4. Job Ticket Edit

This dialog pops up after pressing the **Edit...** button in a **Job Ticket List Item**.

A screenshot of a 'Job Ticket' dialog box. The dialog has a title bar with a close button. It contains several settings, each with a label and a dropdown menu. The 'Copies' setting is highlighted with a blue border and shows the value '20'. The other settings are: Media Size (A4), Media Type (Paper 80 grams), Double-sided printing (One-sided), Rotate by 180 degrees (Off), Pages per side (1), Staple (Staple (top left)), Punch (none), Media Color (Colored), Finishing by operator (Leitz Folder), Booklet (none), and Cover (none). At the bottom, there are 'Save' and 'Cancel' buttons.

Setting	Value
Copies	20
Media Size	A4
Media Type	Paper 80 grams
Double-sided printing	One-sided
Rotate by 180 degrees	Off
Pages per side	1
Staple	Staple (top left)
Punch	none
Media Color	Colored
Finishing by operator	Leitz Folder
Booklet	none
Cover	none

Figure 5.5. Job Tickets: Edit Ticket

Job Ticket Options can be changed (overruled) by the Job Ticket Operator. Costs are recalculated after **Save**.



Note

The number of copies can *not* be edited when **Print Delegation** is applied.

5.1.5. Job Ticket Print

This dialog pops up after pressing the **Print...** button in a **Job Ticket List Item**.

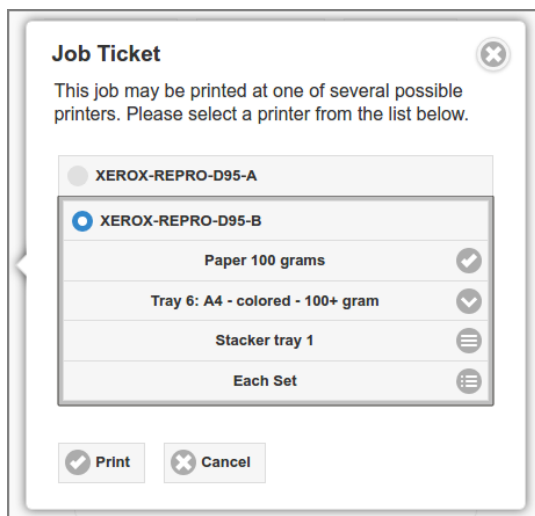


Figure 5.6. Job Tickets: Print Ticket

Redirect printers are shown from the Job Ticket Proxy Printer Group, that are compatible with ticket options. See [Section 4.8.2.1, “Proxy Printer Identity” \[105\]](#).

A selected printer shows the following settings, from top to bottom:

- **media-type** : this is the setting from the Ticket. It can be changed in the [Edit Dialog](#).
- **media-source**: pick a source that is configured on the printer to hold the media-type.
- **output-bin**: pick a suitable output bin.
- **org.savapage-finishings-jog-offset** : select a finishing that shifts “Each Set” in the **output-bin** from the previous one by a small amount (or chose “none”).

After pressing the **Print** button, the dialog closes, and the status of the Ticket will be shown in the list as either **Pending**, **Processing** or **Completed**.



Note

Printer settings are saved in the Web Session, and used as default for next Job Ticket prints. The last jog-offset choice is used as default for *all* printers.

5.1.5.1. Job Ticket Print Pending

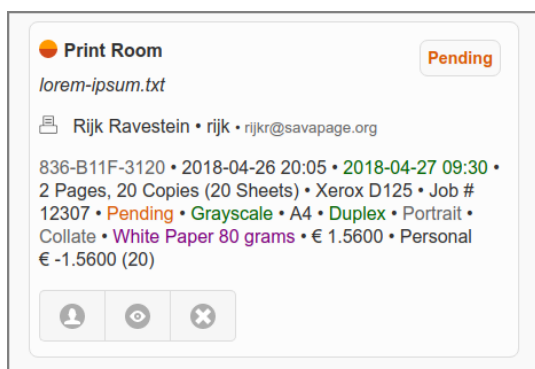


Figure 5.7. Job Tickets: Print Pending

The job is waiting to be printed. By pressing **Cancel**, the job will be removed from the print queue, and the Ticket will show status **Canceled** in the list.

5.1.5.2. Job Ticket Print Processing

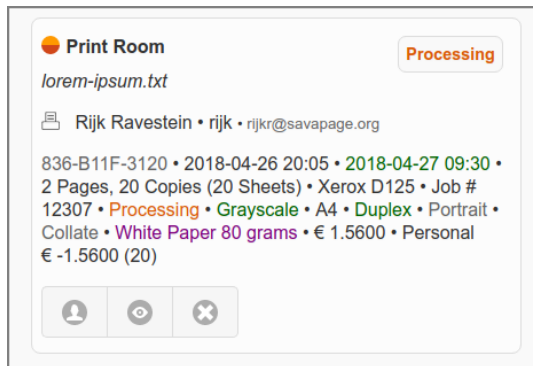


Figure 5.8. Job Tickets: Print Processing

The job is printing. By pressing **Cancel**, printing will be stopped, the job is removed from the print queue, and the Ticket will show status **Canceled** in the list.

5.1.5.3. Job Ticket Print Canceled

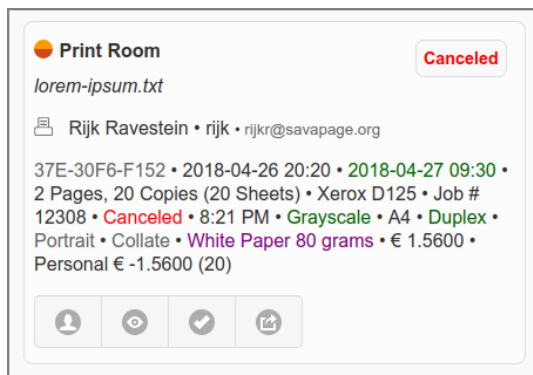


Figure 5.9. Job Tickets: Print Canceled

The print job is canceled. By pressing **Close**, the Ticket is closed and removed from the list. **Retry** to **Print** again, optionally to another printer.

5.1.5.4. Job Ticket Print Completed

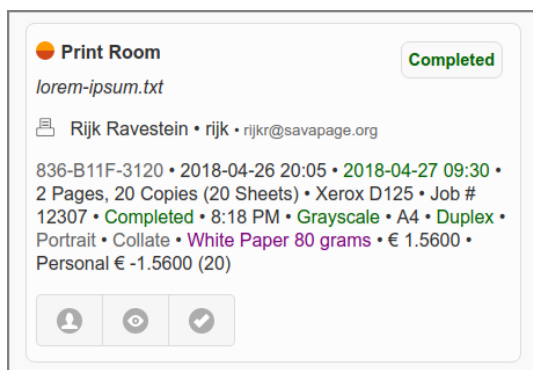


Figure 5.10. Job Tickets: Print Completed

By pressing **Close**, the Ticket is closed and removed from the list, and the user is optionally notified by email: see [Section 3.4.8, “Job Ticket Print” \[51\]](#).

5.1.6. Job Ticket Settle

This dialog pops up after pressing the **Settle...** button in a [Job Ticket List Item](#).

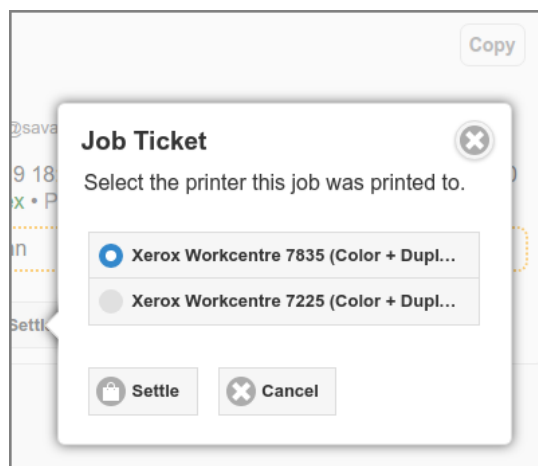


Figure 5.11. Job Tickets: Settle Ticket

After **Settle**, the ticket is effectuated without proxy printing. This implements the scenario where the Job Ticket Operator prints the attached PDF with an external program, and registers the printing occurrence (including financial cost) in SavaPage. After settlement, the Ticket is automatically closed and removed from the list. And, the user is optionally notified by email: see [Section 3.4.8, “Job Ticket Print” \[51\]](#).



Note

Settle is the only option to charge a Copy Job Ticket.

5.2. Closed Tickets

The Closed Ticket List is identical to the [Documents List](#) with an implicit selection of [Type Ticket](#). Here you can query Job Tickets that were closed by the Job Ticket Operator, optionally by User ID of the Job Ticket Creator.

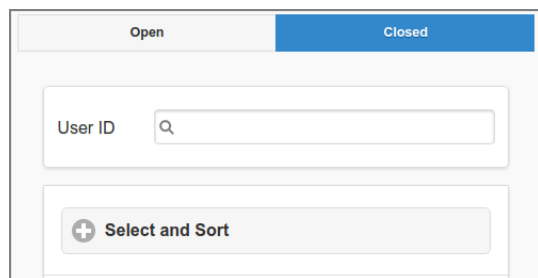


Figure 5.12. Job Tickets: Closed Ticket List

In case costs were incorrectly charged, they can be refunded by pressing the refund button.

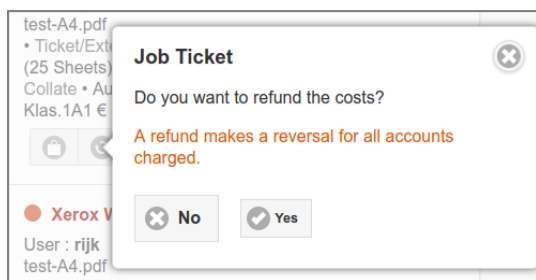


Figure 5.13. Job Tickets: Refund

Refunds are shown in the document transaction details (see screenshot below), and in the transaction list of the Personal, Group and Shared Accounts involved. See [Section 3.8.2, “Transactions” \[61\]](#) and [Section 4.6.1, “Account List” \[96\]](#).

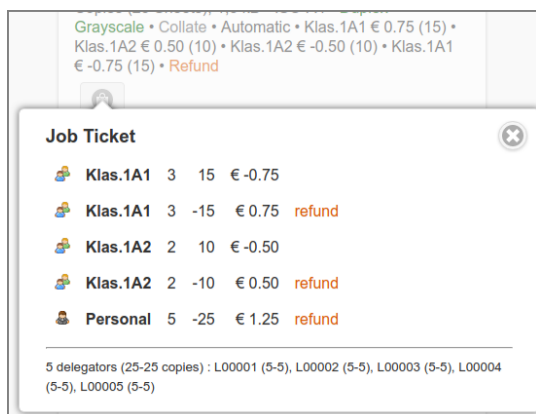


Figure 5.14. Job Tickets: Refund Transactions



Note

When both [PaperCut Integration](#) and [Delegated Print with PaperCut](#) are enabled, and the Job Ticket was printed to a PaperCut managed printer, refund transactions will also be created in PaperCut. See [Section N.1.3, “PaperCut Delegated Print Accounting” \[317\]](#).

5.3. Ticket Configuration Properties

Configuration property	Description
jobticket.notify-email.content-type.html	Set to Y or N (default), to send Job Ticket email notifications as "text/html" (Y) or "text/plain" (N).
jobticket.notify-email.completed.enable	Set to Y (default) or N, to enable/disable notification by email to owner of job ticket when ticket is completed. A standard email message is sent. Email content and layout can be customized if needed. See Section 16.2, “Email Templates” [230] and Section 16.2.6.2, “JobTicketCompleted Email” [235] .
jobticket.notify-email.canceled.enable	Set to Y (default) or N, to enable/disable notification by email to owner of job ticket when ticket is canceled. A standard email message is sent. Email content and layout can be customized if needed. See Section 16.2, “Email Templates” [230] and Section 16.2.6.1, “JobTicketCanceled Email” [234] .

Configuration property	Description
webapp.jobtickets.list-size	The Number of Job Tickets shown in the list. Default: 10.
webapp.jobtickets.list-size-min	The minimum Number of Job Tickets that can be shown in the list. A value of zero means all pending tickets are shown. Default: 5.
webapp.jobtickets.list-size-max	The maximum Number of Job Tickets that can be shown in the list. Default: 50.
webapp.jobtickets.print-all.enable	Set to Y (default) or N, to enable/disable the Print All... bulk action.

Table 5.1. Job Ticket Print Configuration Properties

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to set these items.

Chapter 6. Point-of-Sale Web App

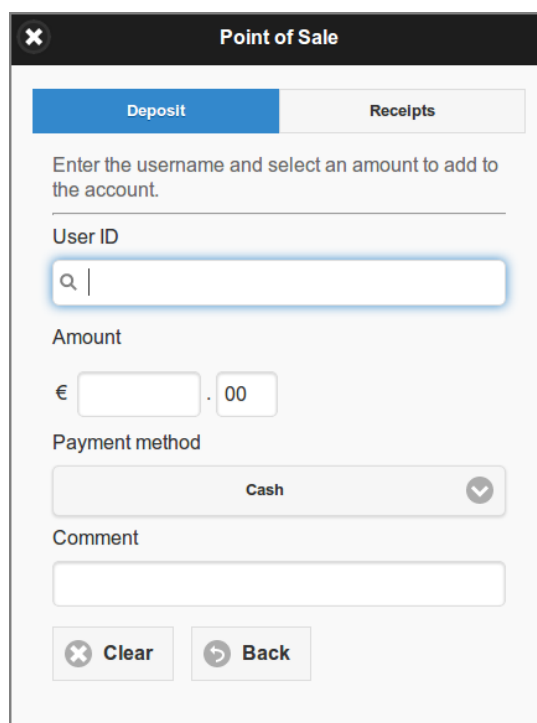


The Point-of-Sale (POS) is a Web App used to deposit funds to a user's printing account, usually after receiving a cash or electronic payment. It can be accessed by users with role **Web Cashier**.

The POS Web App can be reached at `https://savapage:8632/pos`. See **Appendix E, URL Cheat Sheet [278]**.

6.1. Deposit

The **Deposit** tab is the place to handle a user's deposit. The figure below shows the initial content when first used or after the **Clear** button was pushed.

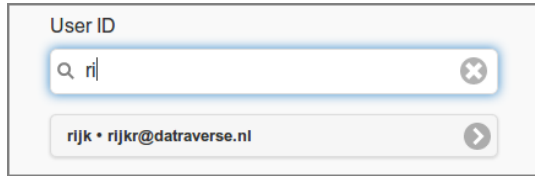


The screenshot shows a web application window titled "Point of Sale" with a close button (X) in the top left corner. Inside the window, there are two tabs: "Deposit" (active, highlighted in blue) and "Receipts". Below the tabs, there is a text prompt: "Enter the username and select an amount to add to the account." followed by a horizontal line. The form contains the following fields and controls:

- User ID:** A search input field with a magnifying glass icon and a vertical line cursor.
- Amount:** A currency input field showing "€" followed by a text box for the integer part, a decimal separator ".", and a text box for the cents part (currently showing "00").
- Payment method:** A dropdown menu currently displaying "Cash" with a downward arrow icon.
- Comment:** A text input field.
- Buttons:** At the bottom, there are two buttons: "Clear" with an 'X' icon and "Back" with a circular arrow icon.

Figure 6.1. Point-of-Sale: Deposit Start

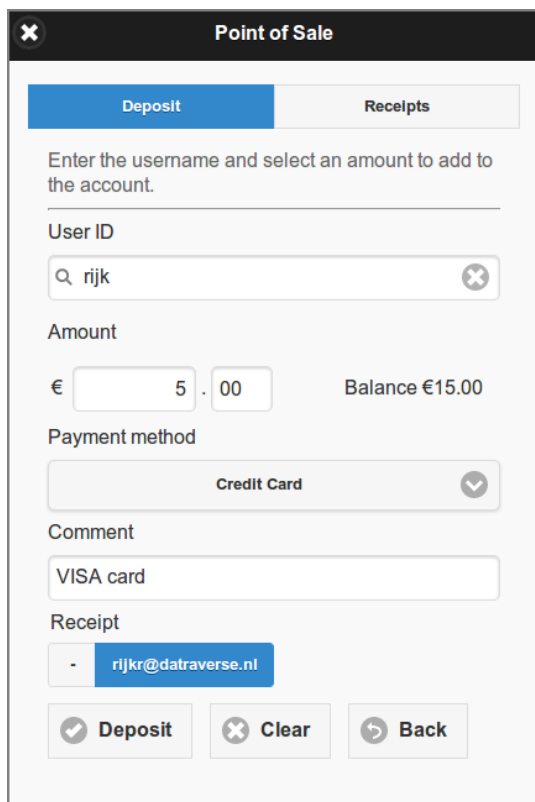
- **User ID:** a quick search entry field to select the User. See the **figure below** for a description.
- **Amount:** enter the integer and cents of the amount separately. The currency sign is taken from the **Financial** settings.
- **Payment method:** select one of the payment methods as specified in the **Financial** settings. When no methods are specified this field will not show.
- **Comment:** a short comment to denote the deposit.
- **Receipt:** select the email address if you want to send the receipt as PDF.



A screenshot of a 'User ID' quick search entry field. It features a search input box with a magnifying glass icon and a close button (X). Below the input box, a dropdown menu displays a search result: 'rijk • rijk@dataverse.nl' with a right-pointing arrow.

- **User ID** is a quick search entry field. By entering part of the user id, a pick-list of matching users is displayed below. The list is refreshed real-time as characters are entered (or deleted).
- By selecting the user from the list the entry field is replaced by the selection. Also, the current account balance of the user and his email is shown.

When all required field are entered the **Deposit** button will show.



A screenshot of the 'Point of Sale' application window. The window has a title bar with a close button (X) and the text 'Point of Sale'. Inside, there are two tabs: 'Deposit' (active) and 'Receipts'. The 'Deposit' tab contains the following fields and controls:

- Instruction: 'Enter the username and select an amount to add to the account.'
- User ID**: A search input box with 'rijk' entered and a close button (X).
- Amount**: A currency input field showing '€ 5 . 00' and a 'Balance €15.00' label.
- Payment method**: A dropdown menu showing 'Credit Card' with a checkmark icon.
- Comment**: A text input box containing 'VISA card'.
- Receipt**: A section showing a receipt icon and the email address 'rijk@dataverse.nl'.
- Buttons**: Three buttons at the bottom: 'Deposit' (with a checkmark icon), 'Clear' (with an X icon), and 'Back' (with a circular arrow icon).

Figure 6.2. Point-of-Sale: Deposit Completed

Push the **Deposit** to make the deposit. As a result:

- When user's email address was selected, the receipt will be emailed to the User.
- The form will be cleared, with the focus on the User ID quick search field.

6.2. Receipts

The **Receipts** tab is the place to query deposit history. The figure below shows a content sample.

Point of Sale

Deposit Receipts

Enter date and time as offset to find receipts.

yyyymmddhhmm

Q 20140423

rijk 4/23/14 3:26 PM

€5.00 • VISA card

PDF rijk@dataverse.nl

Back

Figure 6.3. Point-of-Sale: Receipts

- By entering a date/time offset in the prescribed **yyyymmddhhmm** format, a pick-list of matching receipts is displayed, sorted in descending date/time order. The list is refreshed real-time as characters are entered (or deleted). Note: the date/time defaults to the current yyyymmdd date (today).
- Each entry in the list has buttons to download the Receipt PDF or email it to the User (again).

Chapter 7. User Client



The User Client is a Java application for desktops and notebooks that resides in the *system tray*.

According to the [Oracle Java Documentation](#)¹:

"The system tray is a specialized area of the desktop where users can access currently running programs. This area may be referred to differently on various operating systems. On Microsoft Windows, the system tray is referred to as the Taskbar Status Area, while on the GNU Network Object Model Environment (GNOME) Desktop it is referred to as the Notification Area. On K Desktop Environment (KDE) this area is referred to as the System Tray. However, on each system the tray area is shared by all applications running on the desktop."

The SavaPage User Client is provided as a notifier of personal user events like:

- A successful print to SavaPage. See [Chapter 10, SavaPage as Printer](#) [195].
- A Proxy Printer started printing one of your jobs. See [Section 4.8, "Proxy Printers"](#) [101]

A notification message is typically displayed near the SavaPage tray icon in the form of a balloon (Windows) or message box (GNU/Linux, macOS).

The [User Web App](#) opens for the authenticated user at a double-click on the tray icon, a click in the notification message or selecting the Open Web App... item from the tray icon context menu. When the User Client is trusted as authentication source no extra login is needed.

User Client authentication is explained in [Section 4.10.14.1, "User Client Authentication"](#) [149].

Client access can be restricted by IP address with these configuration properties:

Configuration property	Description
cliapp.ip-addresses-allowed	A CIDR Set of Client IP addresses that are allowed to use the User Client App (when void, all client addresses are allowed).
cliapp.auth.ip-addresses-denied.enable	Enable (Y) or Disable (N) User Client Authentication for clients that are denied for their IP address.

Table 7.1. User Client Access Configuration Properties

See [Section 4.10.14.10, "Config Editor"](#) [155] on how to enter these items.



Important

When using the User Client concurrently with the [User Web App](#) and [Proxy Print Authentication](#) you are strongly advised to use an external database like PostgreSQL. See [Chapter 17, Using an External Database](#) [236].

¹ <https://docs.oracle.com/javase/tutorial/uiswing/misc/systemtray.html>



Warning

Java applications with system tray icons do not work properly with GNOME Shell. This is a persistent [Java bug²](#) that is still not resolved. As a workaround, use the `--anchor` command line option for an alternative display.

7.1. User Client Options

In order of precedence, User Client options can be set ...

1. On the Command-line.
2. As value in a `client.properties` file. An annotated template is available in the `/opt/savapage/client/app/config/` directory.
3. As Configuration Property.

```
usage: savapage-client <options>
  --anchor <ne|nw|se|sw>    Show on desktop at anchor position instead of tray.
                             --notify-send switch is auto activated (Linux only).
  -d,--debug                Write debug messages to the log file.
  -h,--help                 Display help text in GUI.
                             --help-tui          Display help text in TUI.
                             --log-dir <dir>      Log file directory. Default: $HOME
                             --notify-send        Use 'notify-send' command to send desktop
                             notifications (Linux only).
  -p,--print-dialog         Show action dialog at print-in event.
                             --passkey <key>      The admin passkey (optional).
                             --print-dialog-btn <arg> Button text on print-in action dialog for opening
                             User Web App (optional).
                             --print-url-query <arg> URL query for opening User Web App at print-in event
                             (optional).
                             --properties <file>   File with default command-line options (optional).
                             Default: $APP/config/client.properties
                             --server-host <arg>   IP address or name of SavaPage server
                             --server-port <number> SSL port of SavaPage server (optional). Default:
                             8632.
                             --user <name>        A different username than current user $USER
                             (optional).
  -x,--hide-exit           Hide the "Exit" menuitem (optional). Default: false.
```



Note

The `passkey` option can also be applied as environment variable `SAVAPAGE_CLIAPP_ADMIN_PASSKEY`. This is the preferred way to use in generic login scripts, since the command-line option might be visible in system process viewers.

On Debian, the `notify-send` utility is part of the `libnotify-bin` package.

Configuration property	Command-Line Option
<code>cliapp.print-in.url-querycliapp.print-in.url-query</code>	<code>--print-url-query</code> The query string to be appended to the base URL when opening the User Web App in response to a print-in event. Do <i>not</i> prefix the value with a <code>?</code> or <code>&</code> character.
<code>cliapp.print-in.dialog.button-open</code>	<code>--print-dialog-btn</code>

² https://bugzilla.redhat.com/show_bug.cgi?id=1014448

Configuration property	Command-Line Option
	Action button text on print-in action dialog for opening User Web App.

Table 7.2. User Client Options Configuration Properties

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to enter these items.

7.2. User Client Deployment

Since the software is written in Java it can be deployed on any platform where Java SE 7 or higher is installed, like Microsoft Windows, macOS and GNU/Linux. The User Client software is installed on the server in the `/opt/savapage/client/app` directory.



Caution

Running the User Client directly from a network share can cause performance problems, resulting in an unresponsive application. Therefore it is recommended to run the client from a local copy.

7.2.1. Deployment on Windows

The `savapage-client.bat` batch file that starts the application is available in the `/opt/savapage/client/app` directory on the server.



Note

Make sure to enable Balloon Notification in Windows Group Policy to allow users to see notification messages.

7.2.2. Deployment on macOS

The `savapage-client` shell script that starts the application is available in the `/opt/savapage/client/app` directory on the server.




7.2.3. Deployment on GNU/Linux

The `savapage-client` shell script that starts the application is available in the `/opt/savapage/client/app` directory on the server.

Chapter 8. SavaPage Financial



SavaPage Financial captures many aspects of user activity. Obviously, proxy printing is the main trigger for financial accountability and monitoring, since it consumes tangible resources like paper, ink and toner. This chapter introduces the main financial concepts with references to more detailed parts of the manual.

- Account are used to register financial status (balance) and history (transactions). SavaPage has three types of accounts:
 -  *User Account* : The personal account of an User, optionally restricted by a Credit Limit. See [Section 4.4.4.6, “Financial”](#) [87].
 -  *Shared Account* : Shared accounts act as *cost center* to track printing expenses in a specific area. It does not have a credit Limit: its balance is initialized to zero and is allowed to count down into the negative.
 -  *Group Account* : A shared account that is tied to a [User Group](#) by name.

Shared Accounts are discussed in [Section 4.6, “Accounts”](#) [96].

- Printing costs are configured per Proxy Printer. Pay-per-Print is active for each Proxy Printer that has costs greater than zero.
 - [Section 4.8.2.3, “Media Sources”](#) [106]
 - [Section 4.8.2.4, “Manual Media Sizes”](#) [108]
- Printing costs are charged to Accounts.
- Users get feedback about printing costs and their personal account balance.
 - [Section 3.4.5, “Direct Print Release”](#) [45]
 - [Section 3.2.2.2, “Hold Print Jobs”](#) [30].
 - [Section 3.2.2, “Footer”](#) [29]
 - [Section 3.10.3, “Financial”](#) [68]
- Restricted users can upgrade their account balance with vouchers (pre-paid printing cards), by making a deposit at a point-of-sale, or by transferring money from an external account.
 - [Section 4.14, “Vouchers”](#) [168]
 - [Section 3.10.4, “Redeem Voucher”](#) [68]
 - [Chapter 6, *Point-of-Sale Web App*](#) [183]
 - [Section M.1.1, “Payment Gateway Plug-in”](#) [311]
- All financial transactions can be inspected by administrators. Users can inspect their own transactions.
 - [Section 4.4.1, “User List”](#) [81]
 - [Section 3.8.2, “Transactions”](#) [61]
- Global financial options can be set by administrators.
 - [Section 4.10.12, “Financial”](#) [145]

Chapter 9. SavaPage on GNU/Linux



This section is a supplement to the Install Guide (see [Chapter 2, *Server Installation*](#) [11]). It provides an in-depth explanation of the GNU/Linux installation process, the directory layout and tools involved.

Information in this chapter is technical in nature. It is expected that readers have prior experience with:

- The Unix command line environment
- Unix file permissions

9.1. The Installation Process

SavaPage is supplied as a pre-compiled self-installing application. The installation process is designed to work with all major GNU/Linux distributions. To be sure if your GNU/Linux distro is supported out of the box, please check [Section 1.2, “System Requirements”](#) [3]. Due to the varied nature of some installations and administrator preferences, often some manual configuration is required. This section describes the installation process in detail as well as some additional options available to system administrators.

9.1.1. Manual extraction

SavaPage is supplied in a self-extracting, self-installing archive. The archive is simply a tar archive compressed with gzip, and headed with a shell script to facilitate self-extracting. After extraction is complete, the installation script named `install` is executed to begin the install process in the directory where the archive resides (usually `/opt/savapage`). Some system administrators may like to inspect the contents of the archive, and possibly the installation process itself prior to the actual install.

The self-extracting installer takes a number of command line arguments. The `-e` option will extract the archive into the current working directory ready for inspection. With the `-n` switch the `-i` install will be non-interactive. With this mode you implicitly agree with the AGPL license, and root tasks are collected in a `MUST-RUN-AS-ROOT` script located in the install directory. This script must be run manually as `root` after the installation. Further usage information is available via the `-h` switch.

```
SavaPage Install (c) 2010-2016, Datraverse B.V.

License: GNU AGPL version 3 or later <https://www.gnu.org/licenses/agpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Usage: savapage-setup-0.9.11-linux-x64.bin [-h|-i|-e|-l] [-n] [-v] [FILE]...
  -h      This help text.
  -i      Install after extracting the files (default).
  -n      Non-interactive install: execute MUST-RUN-AS-ROOT afterwards.
  -e      Extract all files or a FILE list and exit without installing.
  -l      List the contents of the archive and exit without extracting.
  -v      Verbose. Print the names of the files as they are extracted.
```

9.1.2. The install process

Even though the majority of the installation process is completed under the identity of the system user account called `savapage`, most administrators would like to know what the install process does. The main steps are outlined in the next paragraphs.

9.1.2.1. Extraction

The first stage in the install process extracts the archive to `/tmp` or a location as defined by an environment variable `TMPDIR`. The command-line programs **tar** and **gunzip** are used during this phase.

9.1.2.2. Installation

After extraction is complete the **install** script is called. The current directory is passed as `-d` option argument, to be used as install location. Also, the `-n` switch, used at the setup binary, is propagated to this script. The extracted files are copied to the install location. *Care is taken not to overwrite any existing data or configuration files if this is an install-over-the-top upgrade.*

```
SavaPage Install (c) 2010-2016, Datraverse B.V.

License: GNU AGPL version 3 or later <https://www.gnu.org/licenses/agpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Usage: install [OPTION]...
       -d <dir>  The target location, usually the /opt/savapage directory.
       -h        This help text.
       -n        Non-interactive install: execute MUST-RUN-AS-ROOT afterwards.
```

9.1.2.3. Permissions

To ensure the default installation is secure by default, permissions are applied to key files. The following area of the application are restricted to the savapage user only:

Area	Description
<code>/opt/savapage/server/server.properties</code>	Contains server configuration properties. See Section 9.3, “Advanced Configuration” [192].
<code>/opt/savapage/server/admin.properties</code>	Contains the hashed password of the reserved internal admin user.
<code>/opt/savapage/server/gcp.properties</code>	Contains Google Cloud Print configuration properties. See Section 4.10.6, “Google Cloud Printer” [133].
<code>/opt/savapage/server/jmxremote.password</code>	Contains the plain text password of the reserved JMX admin user. See Section 4.10.14.3, “JMX Agent” [150].
<code>/opt/savapage/server/jmxremote.keystore</code>	The private keystore used by the JMX Agent .
<code>/opt/savapage/server/data/</code>	This directory contains application data including database files. Some of this data may contain sensitive information.
<code>/opt/savapage/server/bin/linux-x64/</code>	This directory contains the <code>savapage-pam setuid-root</code> binary. Even though the binary is no use to an end-user or hacker, good security practice stipulates that we should only allow the savapage user access to this directory.

Table 9.1. Secured Application Areas

Permissions can be checked and re-applied any time after the installation by running the script:

```
/opt/savapage/server/bin/linux-*/setperms
```

9.1.2.4. Firewall

The SavaPage Application Server runs in a Java Virtual Machine (JVM) process and listens on ports 8631 and 8632 (SSL). These ports are used for Web App access, printing and other services. Ensure that any firewall or local IP filtering software such as `iptables` is set to allow local network traffic access to this ports.

9.1.2.5. Root Level Tasks

A small part of the install process needs to run as the `root` account. The tasks conducted as root include:

- Setting the `/opt/savapage/server/bin/linux-*/savapage-pam` binary as `setuid-root`. This binary is used for password verification.
- Installing the `/opt/savapage/providers/cups/linux-*/savapage-notifier` binary as CUPS event notifier. This binary is used to send CUPS printer and print job status events to the central SavaPage server.
- Setting up a **systemd** unit for GNU/Linux systems that use the systemd service manager. This is done by creating a `savapage.service` file in the systemd unit library. Depending on the distribution the unit will either be created in the `/lib/systemd/system/` or `/usr/lib/systemd/system/` directory. The unit is started and enabled.
- Setting up a *custom* **systemd** unit for the CUPS scheduler **cupsd**.
 - When the scheduler is run from systemd some systems pass the `-l` parameter so `cupsd` is run on demand by socket and path activation. The advantage of this setup is that CUPS is activated when needed, saving precious boot time and resources, and deactivated again after being idle for a while. This lazy activation scenario is efficient for desktop systems that print occasionally and for which printing is not time critical.
 - However, dedicated print systems like SavaPage, that intensively use IPP to communicate with CUPS, need CUPS to be full-time activated. Therefore a custom systemd `cups.service` unit is installed in `/etc/systemd/system/` to override the default `/lib/systemd/system/cups.service` shipped with the CUPS package. This custom unit starts `cupsd` with the `-f` parameter so it runs steadily in the foreground (without dependencies for `cups.socket` and `cups.path`).
- Setting up **SysV** style start scripts for Debian based systems that use the SysV service manager. This is done by placing symlinks in the:

```
/etc/init.d/  
/etc/rc3.d/  
/etc/rc5.d/  
and so on...
```

If the administrator decides not to run the root-level tasks during the install process, the tasks can be run again post-install by executing the shell scripts:

```
/opt/savapage/server/bin/linux-*/roottasks
```

and ...

```
/opt/savapage/providers/cups/linux-*/roottasks
```

Alternatively the administrator can view the script and make the required changes by hand.

9.2. Logs

The main application logging is available via the [Application Log](#) section of the Administrator Web App. Additional advanced level logging is maintained in standard text files located at:

```
/opt/savapage/server/logs/*
```

Administrators may wish to consult these logs when attempting to diagnose or troubleshoot problems.

9.3. Advanced Configuration

The majority of SavaPage configuration is conducted in the [Administrator Web App](#). Some additional configuration options are available in the `/opt/savapage/server/server.properties` configuration file. This file contains server configuration including the server's TCP ports and SSL properties and Alternative File Locations.

9.3.1. Alternative TCP/IP Settings

Key	Description
<code>server.port</code>	Server http port. Default: 8631 <ul style="list-style-type: none"> Value 0 : port is disabled.
<code>server.ssl.port</code>	Server https port. Default: 8632 <ul style="list-style-type: none"> The SSL port can <i>not</i> be disabled.
<code>server.html.redirect.ssl</code>	Redirect HTML of non-SSL port to SSL: true or false (default). This only concerns HTML, IPP traffic is not redirected.
<code>server.print.port.raw</code>	The RAW Print Server port. Default: 9100 <ul style="list-style-type: none"> Value 0 : RAW printing is disabled.

Table 9.2. Server Properties for Alternative TCP/IP Settings

9.3.2. Alternative File Locations

To separate created files from installation files you can set alternative locations for temporary files, **SafePages** and public **Letterheads** in the `/opt/savapage/server/server.properties` configuration file.



Important

All alternative file locations must reside on the same disk partition¹.

Key	Description
<code>app.dir.tmp</code>	Location of the application's temporary files. It is created when the application starts and removed when stopped. Default: <code>/opt/savapage/server/tmp</code> <ul style="list-style-type: none"> User <code>savapage</code> must have permission to create the location.
<code>app.dir.safepages</code>	Location where the user's SafePages are stored. Default: <code>/opt/savapage/server/data/internal/safepages</code> <ul style="list-style-type: none"> The location directory must be owned by user <code>savapage</code> and have permission 700.
<code>app.dir.letterheads</code>	Location where the public Letterheads are stored. Default: <code>/opt/savapage/server/data/internal/letterheads</code> <ul style="list-style-type: none"> The location directory must be owned by user <code>savapage</code> and have permission 700.

Table 9.3. Server Properties for Alternative File Locations

9.3.3. Miscellaneous Settings

Key	Description
<code>start.cleanup-doclog</code>	Enable database cleanup at server start-up. See Section 4.10.13, “Backups” [148]. Values: true (default) or false

¹ This constraint is needed because files are initially created in the temporary location and *atomically* moved to their destination. Atomic moves do *not* work cross-partition.

Key	Description
<code>webapp.custom.i18n</code>	See Section 16.1.1.4, “Custom i18n” [230]. Values: true or false (default).

Table 9.4. Server Properties: Miscellaneous Settings

9.4. Upgrading SavaPage

Upgrading SavaPage is just an install of the new version “over-the-top” of the old version. It follows the same procedure as a first time installation, as described in Section 9.1, “The Installation Process” [190].

The process keeps existing data and configuration files as they are.



Important

Always check the [Release Notes](#)² after upgrading, to see if additional actions are needed before you can actually use the new version.

9.5. Removing SavaPage from a GNU/Linux server

SavaPage can be completely removed from a system with the following procedure:

- Remove all files from the `/opt/savapage` install directory.
- Remove the `savapage` system account.
- Remove the `savapage` binary from the CUPS notifier directory.
- For **systemd** installations:
 - Remove the `savapage.service` file in the `systemd` unit library. Depending on the distribution the unit will either be found in the `/lib/systemd/system` or `/usr/lib/systemd/system` directory.
 - Remove the custom `/etc/systemd/system/cups.services` file.
- For **SysV** style installations remove any matching start script:

```
/etc/init.d/savapage
/etc/rc*.d/*savapage
```



Note

Removing SavaPage can also be done by executing the `uninstall` program like this:

```
cd /opt/savapage
sudo uninstall
```

The installation will be reverted including the CUPS notifier installation and the server's `systemd` or `SysV` service scripts.

As a final action the `savapage` system account and the `/opt/savapage` install directory should be removed manually.

² <https://wiki.savapage.org>

Chapter 10. SavaPage as Printer



10.1. Printing with a Driver

Any desktop system can print to SavaPage with a PostScript printer driver. The driver can either be generic, or a mainstream one from a vendor as shipped with the OS, or a dedicated one provided by SavaPage. When printing from public Internet a private Device URI must be used. See [Section 3.10.1, “Internet Printer”](#) [67].



Caution

Although the SavaPage driver is not required, beware that vendor-specific drivers might offer options that are irrelevant, or not supported by the SavaPage Print Server.

10.1.1. SavaPage Printer Driver

The SavaPage Printer Driver comes as a PostScript Printer Description (PPD), as captured in the `SAVAPAGE.ppd` file located in the [shared client directory](#) `/opt/savapage/client`. The driver is optimized for SavaPage printing. Irrelevant options, like *Duplex Printing* are stripped, other options like *Paper Size* and *Resolution*, are narrowed down to the most common choices. If you feel options are missing please let us know.



Important

`SAVAPAGE.ppd` is a PostScript-only driver, and does *not* produce PDF. This is on purpose, since PDF producing drivers print landscape orientated documents with 270 page rotation. As a result these documents will show as portrait orientated SafePages.¹

The driver file can be downloaded from the About section of the [User Web App](#) and [Admin Web App](#).

10.1.2. SavaPage Printer Installation

The installation scripts below use the SavaPage printer driver. When you want to use a PostScript driver already present in the OS, please use the proper selection dialogs.



Caution

The SavaPage JetDirect Server accepts PostScript print jobs only. So do not use the JetDirect protocol unless you are absolutely sure that the print client uses PostScript as Print Job Format. Windows clients can safely use JetDirect. On macOS and GNU/Linux systems IPP and IPPS are the obvious choices.

¹ SavaPage acts as a Virtual PostScript Printer Device. This is separate from the decision on the OSDL Printing Summit in 2006 to switch the GNU/Linux standard print job transfer format from PostScript to PDF. See https://wiki.linuxfoundation.org/en/OpenPrinting/PDF_as_Standard_Print_Job_Format

10.1.2.1. GNU/Linux

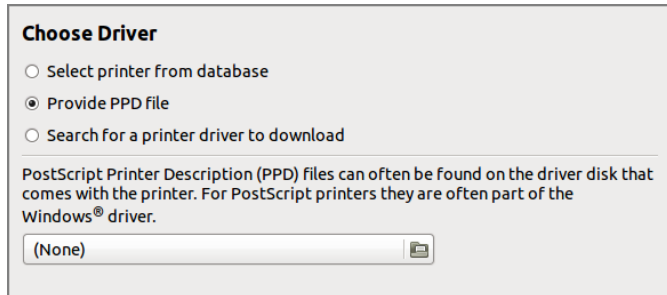


Figure 10.1. SavaPage Printer on Ubuntu: Choose Driver

- When choosing a driver for the newly added printer in Ubuntu, make sure to opt for **Provide PPD file**, and to select the `SAVAPAGE.ppd` file.
- Enter `ipps://savapage:8632/printers` at **Device URI** for the default queue, or `ipps://savapage:8632/printers/[queue]` for any other specific queue. See [Appendix E, URL Cheat Sheet \[278\]](#).

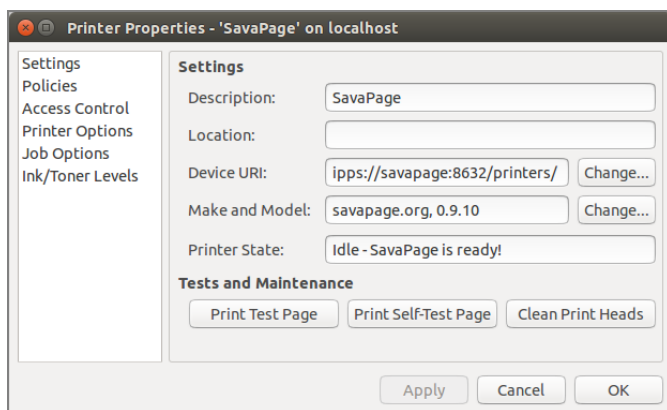


Figure 10.2. SavaPage Printer on Ubuntu: Printer Properties

- This is what the Printing Properties look like for a ready-to-print SavaPage printer in Ubuntu.

10.1.2.2. macOS

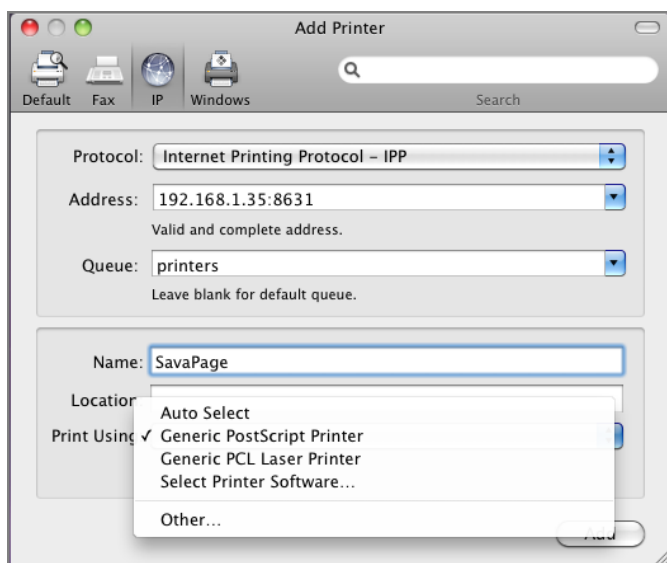


Figure 10.3. SavaPage Printer on macOS: Add Printer

Add a new printer and enter data in the Add Printer dialog as follows:

- Click the **IP** printer button and select IPP for **Protocol**.
- At **Address**, enter the IP address or host name of the SavaPage Print Server *including* the port number.
- Enter printers at **Queue** for the default queue, or printers/[queue] for any other specific queue. See [Appendix E, URL Cheat Sheet \[278\]](#).
- Enter the **Name** of the queue. SavaPage is the obvious choice here.
- Choose **Other...** in the **Print Using** selection box. This will immediately pop up a dialog where you can select the SAVAPAGE .ppd as shown in [Figure 10.4, “SavaPage Printer on macOS: Select PPD” \[197\]](#).

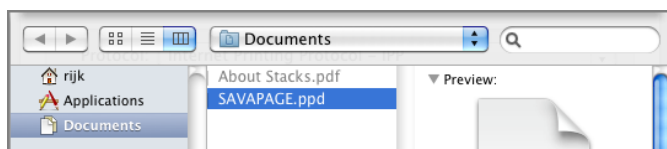


Figure 10.4. SavaPage Printer on macOS: Select PPD

- This dialog selects the SAVAPAGE .ppd file from the local Documents directory.

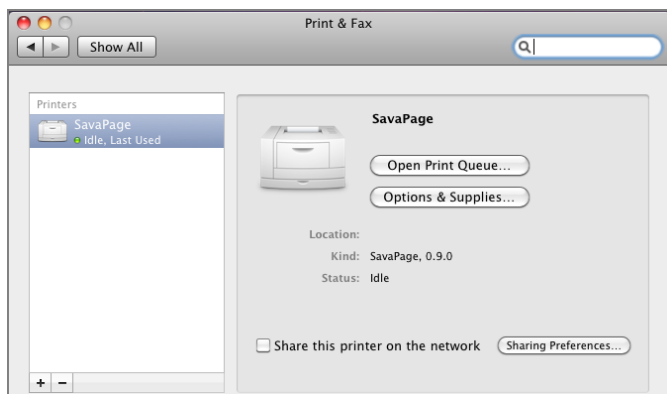


Figure 10.5. SavaPage Printer on macOS: Print & Fax

- This is what a ready-to-print SavaPage printer in macOS looks like.



Note

When clicking the **Default** printer button in the **Add Printer** dialog, any Bonjour enabled SavaPage printer will show up, as configured in [Section 10.3, “Printing from iOS” \[200\]](#).

10.1.2.3. Windows

This section covers the installation for Windows (including x64).

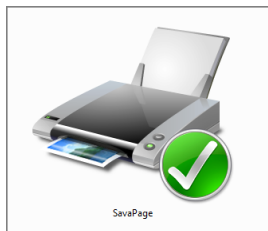


Figure 10.6. SavaPage Local Printer on Windows

To add SavaPage as Local Printer, start the "Add Printer" dialog and choose add a new Local Printer.

1. Create a new printer port of type *Standard TCP/IP Port*, and click the **Next** button.
2. Choose device type **TCP/IP Device** and enter the hostname or IP address of the SavaPage server.
3. When asked for a printer driver, choose a PostScript printer driver from the list. Any type/model will do, as long as it generated PostScript spool files. It makes sense to select just a simple type/model, without fancy options.
4. Assuming you named the printer “SavaPage”, you should now have a printer as shown in [Figure 10.6, “SavaPage Local Printer on Windows” \[198\]](#).
5. Print a test page to see if everything works as expected.



Figure 10.7. SavaPage Shared Local Printer on Windows



Tip

Install SavaPage as shared printer on a Windows Print Server. This makes the printer a member of Active Directory. See [Figure 10.7, “SavaPage Shared Local Printer on Windows” \[198\]](#).

Queues created on Windows Print Server can easily be deployed on workstations using Windows Domain Group Policy or using Logon Script. Please consult the Microsoft Windows server documentation for more information.



Figure 10.8. SavaPage Network Printer on Windows

To add SavaPage as Network Printer, start the "Add Printer" dialog and choose add a new Network Printer.

1. Select "Connect to a printer on the Internet..."
2. Enter the URL for the SavaPage printer queue.
3. Choose a PostScript printer driver from the list. Any type/model will do, as long as it generated PostScript spool files. It makes sense to select just a simple type/model, without fancy options.
4. Assuming you named the printer "SavaPage", you should now have a printer as shown in Figure 10.8, "SavaPage Network Printer on Windows" [199].
5. Print a test page to see if everything works as expected.

10.2. Printing with AirPrint

Devices running macOS 10.7 and higher or iOS 4.2 and higher (like iPad, iPhone and iPod Touch) can use AirPrint® to print to SavaPage.



Note

AirPrint maps to the reserved `Queue /airprint`.



Important

Avahi needs to be installed on your GNU/Linux host. See Section 1.2.1.6, "Avahi" [5].

To setup SavaPage AirPrint printing follow the steps described in the sections below.

10.2.1. Step 1: Enable IPv4 in Avahi

Since SavaPage uses IPv4 for **IP Based Authentication** IPv4 should be enabled in the avahi-daemon. This is normally the case, but you can check by editing the Avahi configuration file:

```
sudo vi /etc/avahi/avahi-daemon.conf
```

Make sure the `use-ipv4` settings is as follows:

```
use-ipv4=yes
```

When you made changes to the configuration file, restart the daemon as follows:

```
sudo service avahi-daemon restart
```

10.2.2. Step 2: Create AirPrint Queue

Create a SavaPage Queue (see Section 4.7, "Queues" [99]) with a comprehensible **URL path** mnemonic like "airprint". It is important *not* to check the **Trusted** option, since the queue should be *untrusted* to enforce **IP Based Authentication**.

This is needed because iOS printing is *unauthenticated*, i.e. all print jobs have “guest” as originating user. **IP Based Authentication** finds the “real” user by matching the IP address of the print request with the authenticated user in the SavaPage Web App Session on the same IP address.

10.2.3. Step 3: Create Avahi Service File

Copy the `/opt/savapage/server/examples/linux/avahi/savapage.service` file to your personal home directory.

`savapage.service` is an Avahi service file with annotations explaining how to adapt it to your own situation. Follow the `$Customize$` annotation to insert your settings. Probably, you can just accept the defaults.

Copy your tailored service file to Avahi, with this command (assuming the source file resides in your home directory):

```
sudo cp ~/savapage.service /etc/avahi/services
```

Check if Avahi has published the SavaPage printer as intended by typing this command:

```
avahi-browse -a -t
```

Assuming your GNU/Linux host is called `savapage` and you named your Avahi print service `SavaPage`, you should find entries in the output like this :

```
+ eth1 IPv4 SavaPage @ savapage Internet Printer local ❶
```

❶ The mDNS published SavaPage internet printer on host `savapage` for the IPv4 interface.

To check if the SavaPage DNS printer is can be found, use this command on a macOS or GNU/Linux workstation :

```
ippfind --remote
```

Assuming your GNU/Linux server host is called `savapage` in the Avahi service file, you should see this entry as output:

```
ipp://savapage.local:8631/printers/airprint
```



Note

Configuring Avahi to use `ipps` failed so far. Service file settings as below have no effect:

```
<type>_ipps._tcp</type>
<subtype>_universal._sub._ipps._tcp</subtype>
<port>8632</port>
```

`ippfind` does *not* identify the service.

10.3. Printing from iOS

Make sure AirPrint is configured as described in [Section 10.2, “Printing with AirPrint” \[199\]](#). Follow the steps below to use AirPrint on iOS.

10.3.1. Step 1: Install iOS Web Clip

For your convenience `https://savapage:8632/ios/install` is available to add a SavaPage icon to the iOS Home Screen automatically. See [Appendix E, URL Cheat Sheet \[278\]](#). A click on the icon opens the SavaPage User Web App full-screen and will therefore be part of the multitasking bar.

This convenience comes with a penalty though, since Apple treats full-screen WebApps in a “special” way, i.e. when they are selected from the multitasking bar and regain focus the Web App is reloaded.

Luckily, since SavaPage utilizes an **authentication token**, an automatic login is performed. However, the page needs to be retrieved from the server again, giving some performance penalty.



Note

Only one SavaPage Web Clip can be present on an iOS device. A new install overwrites the previous one.



Warning

When using the **Payment Gateway Plug-in**, the redirect URL as forwarded to and applied by the payment provider does not show in the same User WebApp as where the payment started, but is shown on a new tab in the default browser.

If you don't care about the full-screen User Web App, and want optimal performance, you can just add any SavaPage Web App to the Home Screen manually by surfing to the URL, then click the **Share** button and choose **Add to Home Screen**. Clicking the home screen icon will not open the Web App full-screen, but as a tabbed instance in the browser. Also, it will not be reloaded by definition as the browser regains focus.

10.3.2. Step 2: Test

As a start, first login to SavaPage on your iOS device. This is because **IP Based Authentication** is needed by the SavaPage printer.



Warning

When printing while not logged in a dialog will pop up saying “You do not have permission to use this printer”, with **Cancel** and **Retry** buttons.

In many iOS apps, tapping the action button displays options for sharing, as well as other actions such as printing or copying. The options vary depending on the app you're using. **Figure 10.9, “iPad App Sharing Options” [201]** shows the sharing options from the Notes App.

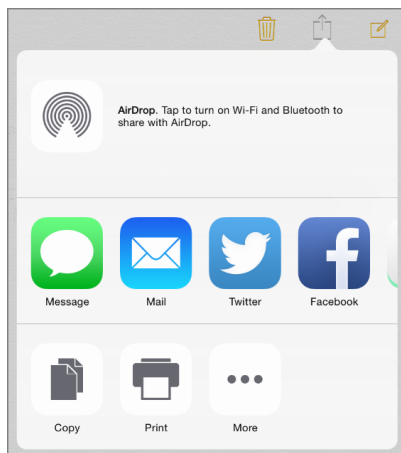


Figure 10.9. iPad App Sharing Options

Tapping the Print icon will bring forward the Printer Options, as shown in **Figure 10.10, “SavaPage Printer Options on iPad” [202]**.

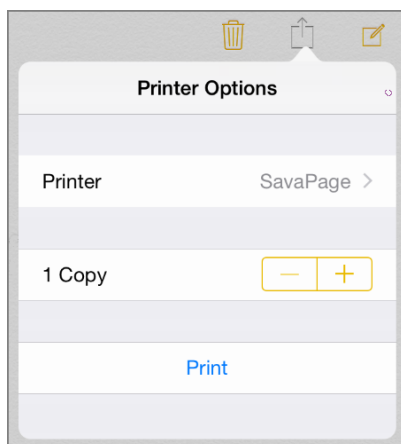


Figure 10.10. SavaPage Printer Options on iPad

If you are printing for the first time, or the previously selected printer is not available, or if you just want to change printer, you will need to select the printer first by tapping the **Printer** button. In this example Figure 10.11, “Select SavaPage Printer on iPad” [202] shows a list with just a single printer (who needs more :-)

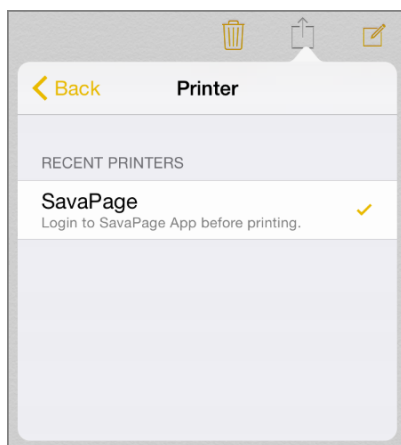


Figure 10.11. Select SavaPage Printer on iPad

Now, after you selected SavaPage as printer and are sure *you logged into SavaPage at the same device*, tap the **Print** button. As a result the printed output should appear in the SavaPage App.



Note

You can check the Print Queue by double-tapping the Home button to show the recently used apps. Then tap the Print Center. Note that the Print Center is only available while printing is in progress.

10.4. Printing from Android and Chrome OS

10.4.1. SavaPage Google Cloud Ready Printer

The **Google Cloud Print**² (GCP) service provides seamless integration with the Google Android and Chrome OS ecosystems. Print jobs can be triggered by a share action, and take a detour via the cloud to the remote printer. GCP offers the same user experience as **Printing from iOS**.

² <https://developers.google.com/cloud-print/>

SavaPage can be configured as Google Cloud Printer. See [Section 4.10.6, “Google Cloud Printer” \[133\]](#).

10.4.2. PrinterShare™ Mobile Print

PrinterShare™ Mobile Print³ is a native Android printing solution from Mobile Dynamix.

PrinterShare uses a local printer driver to convert the candidate pages to graphical images. Unfortunately, when using the generic PostScript driver, this results in a seriously degraded performance at the client side, and non-searchable PDF in SavaPage.



Note

To actually print beyond a simple test page you need to purchase PrinterShare™ Premium Key⁴, which is a separate small application that needs to be present on the device.

Just as in iOS, you need to print to an untrusted SavaPage queue, because PrinterShare printing is unauthenticated, i.e. print jobs do not hold information about the originating user. It is best to create an AirPrint enabled queue for that purpose, as described in [Section 10.3, “Printing from iOS” \[200\]](#).

Perform these steps to create the SavaPage printer:

1. Login to the SavaPage User Web App on your Android device. This is because of the [IP Based Authentication](#) needed by the SavaPage printer selected in the next step.
2. Select a [Nearby-WiFi](#) printer. PrinterShare will recognize the AirPrint enabled SavaPage printer.
3. When selecting the printer driver do *not* [Use Generic](#)⁵, but [Select Manually](#) instead.
4. Open the folder named “Generic” and select the “Generic PostScript Level 2” driver.



Warning

When not logged in while printing PrinterShare will pop up an error message saying “Can't print. IPP error 1025”, meaning SavaPage understood the request, but is refusing to fulfill it.

10.5. Driverless Printing

Driverless Printing is one of the gateways to authenticated printing from devices belonging to unauthenticated anonymous users, where either native printing support is lacking or where user trust cannot be enforced. This situation is typical for a BYOD⁶ context.

[Mail Print](#) and [Web Print](#) offer casual printing without a hassle, since users are familiar with either file uploading or email and no printer driver needs to be installed.

Since rendering of the document content is not handled by a know-it-all printer driver, not all document types are supported. See [Appendix G, *Printable File Types* \[282\]](#) for a list of supported types.



Note

Mail Print uses 127.0.0.1 (localhost) as the IP address of the requesting user.

³ <https://play.google.com/store/apps/details?id=com.dynamixsoftware.printershare>

⁴ <https://play.google.com/store/apps/details?id=com.dynamixsoftware.printershare.premium>

⁵ The generic driver uses UNIRAST as job format, which is not supported by SavaPage. PrinterShare will pop up an HTTP error 500.

⁶ Bring Your Own Device (BYOD) is the policy of permitting employees to bring personally owned mobile devices (laptops, tablets, and smart phones) to their workplace, and use those devices to access privileged company information and applications. The term is also used to describe the same practice applied to students using personally owned devices in an educational setting.

10.6. IP Restricted Printing

Beware that printer access may be restricted based on the requesting IP address. See:

- [Figure 4.39, “Admin Web App: Queue - Edit” \[101\]](#)
- [Figure 4.82, “Admin Web App: Options - Web Print” \[140\]](#)

CIDR Notation is used to specify the allowed IP address ranges.

10.6.1. CIDR Notation

Classless Inter-Domain Routing⁷ (CIDR) is developed in the 1990s as a standard scheme for routing network traffic across the Internet.

CIDR notation is a syntax for specifying IP addresses and their associated prefix size, the latter being equivalent to the number of leading 1 bits in the routing prefix mask. The notation starts with an IP address expressed according to the standards of IPv4 or IPv6. It is followed by a separator character, the slash (/) character, and the prefix size expressed as a decimal number.

Some examples:

- 172.16.0.0/24 represents the given IPv4 address and its associated routing prefix 172.16.0.0, or equivalently, its subnet mask 255.255.255.0. This represents the host address range 172.16.0.0 - 172.16.0.255.
- CIDR 192.168.1.40/32 represents the single IP address 192.168.1.40.



Tip

A CIDR calculator can be found [here](#)⁸.

10.6.2. CIDR Set

For SavaPage use only we define a CIDR Set as a concatenation of single **CIDR notations** separated by any of the characters *space, comma, colon or semicolon*.

10.7. Printing Encrypted PDF

When you print an encrypted PDF document from Adobe Reader to a PostScript printer like SavaPage, it creates a PostScript file that contains a notice telling the recipient that it is not permitted to convert (re-distill) it to PDF again. The **ps2pdf**⁹ program from the Ghostscript suite respects this notice, and throws an error saying “This PostScript file was created from an encrypted PDF file. Re-distilling encrypted PDF is not permitted”. If, for example, an encrypted PDF allows printing only, it should not be re-distilled to a plain PDF equivalent, where all intended protection is removed.

SavaPage respects this policy. Moreover, on behalf of its users SavaPage would like its own encrypted PDF documents to be respected in the same way.

When an encrypted document is allowed to be printed, SavaPage would like to be able to receive it as printer, so it can be previewed and Proxy Printed. However, for that to happen we need to convert it to SafePages, i.e. to PDF format. That's where we are facing the **ps2pdf** barrier.

⁷ <https://tools.ietf.org/html/rfc4632>

⁸ <http://www.subnet-calculator.com/>

⁹ **ps2pdf** is a work-alike for nearly all the functionality (but not the user interface) of Adobe's Acrobat™ Distiller™ product: it converts PostScript files to Portable Document Format (PDF) files, and is implemented as a command script that invokes Ghostscript. See: <http://www.ghostscript.com/doc/current/Ps2pdf.htm>

To solve this issue SavaPage offers an optional workaround. The workaround ignores the PostScript notice at the point where we need the **ps2pdf** program to create the PDF, so SafePages can be displayed and Proxy Printed as intended. However, the workaround will protect the encryption of the original document, i.e. its pages are not allowed to be exported (downloaded or send) as PDF, directly, or as part of a composite document. And, since every PDF document that SavaPage sends to a Proxy Printer is encrypted with (degraded) printing as only permission, this will also not violate any copyright.

The re-distilled PDF files are transient, internal use only and not accessible to end-users.

The workaround is enabled by default. If you do not endorse it, please disable it in the Admin Web App at [Options](#) → [Advanced](#) → [Proxy Printing](#). If the workaround is disabled, every print job request holding an encrypted PDF document is ignored.

Chapter 11. Authenticated Printing



Authentication in a printing environment is the act of confirming the digital identity of the person who issued a print job. Knowledge of this identity is crucial for SavaPage to securely offer its services to the right user. The next sections discuss authenticated printing in:

- [Single Sign-On \(SSO\) Domains](#)
- [Peer to Peer Networks](#)

But first, let us introduce the key authentication concepts where our discussion is based upon.

11.1. Key Concepts

This section lists the main authentication concepts headed with a short *term*. Each term is defined with a concise description, optionally followed with more details and a list of invariants.

11.1.1. User

An actor with a unique identity.

11.1.2. Person

A [User](#) who represents a real human being, as opposed to an abstract human role, software service or hardware device.

- Only Persons can login to SavaPage.
- Any User can print to a SavaPage Printer. However, SavaPage assigns a print job to a Person.

11.1.3. Abstract User

A [User](#) who is not a [Person](#).

11.1.4. Domain User

A [User](#) defined in a [SSO domain](#).

11.1.5. Synchronized User

A SavaPage [User](#) synchronized from a [User Source](#).

- SavaPage assumes each Synchronized User is a [Person](#), but Administrators can mark a user as [Abstract](#).

11.1.6. Synchronized Person

A [Synchronized User](#) that is a [Person](#).

11.1.7. Internal Person

*A [Person](#) who is *internally defined* in SavaPage (opposed to a [Synchronized Person](#)).*

11.1.8. Authenticated User

A [User](#) authenticated on a [SSO domain](#) by a workstation login.

11.1.9. Authenticated Abstract User

An *Abstract User* authenticated on a *SSO domain* by a workstation login.

- Before Authenticated Abstract Users can print to a SavaPage Printer they need to login to the SavaPage Web App on the same device from which they use the printer.

11.1.10. Authenticated Person

A *Synchronized Person* authenticated on a *SSO domain* by a workstation login.

- Authenticated Persons can print to SavaPage without being logged in to the Web App.

11.1.11. Trusted SavaPage Queue

A SavaPage Print *Queue* whose print jobs are trusted to originate from *Authenticated Users*.

- Each SavaPage Print Queue is trusted by default. However, administrators can mark SavaPage Print Queues as *untrusted*.
- Every job of a Trusted SavaPage Queue is checked for the originating User. When this user is an *Abstract User*, SavaPage uses *IP Based Authentication* to deduce the associated *Person*. When the Person cannot be deduced the job is ignored.
- Note that the “trust” qualification is SavaPage internal use only, and not related to network domain trust in any way.
- SavaPage Print Queues are IPP based and, from a network point of view, are publicly accessible by nature.
- In the Microsoft Active Directory world IPP Printers cannot be encapsulated as native domain resource and subjected to native domain access control like JetDirect compatible devices. This is why SavaPage does not bet on native domain trust alone, and accepts public network access as a given fact. But even in this case, SavaPage Print Queues can still be *internally* trusted if access is limited to authorized users on a network level. Stated the other way round: administrators need to prevent that users who connect to the network unauthenticated, e.g. with their personal laptop, use Trusted SavaPage Queues. SavaPage adds a helping hand here with an option to restrict print queue usage to a specific range of IP addresses. This makes it possible for instance to deny trusted queue access for wireless users who get their IP addresses from a distinct DHCP server issuing leases from a distinct IP range.



Caution

When non-domain users are allowed to print to Trusted SavaPage Printers an accidental match with a *Synchronized Person* may lead to undesirable results.

11.1.12. Public SavaPage Queue

A SavaPage Print *Queue* where print jobs are not trusted to originate from *Authenticated Users*.

- Since each SavaPage Printer is trusted by default, this queue must be explicitly marked as *untrusted* in the SavaPage *Admin Web App*.
- SavaPage handles every job from a Public SavaPage Printer as originating from an *Abstract User*.

11.1.13. IP Based Authentication

Deduction of the printing *Person* by matching the *IPv4* address of the originating host of the print job with the authenticated SavaPage Web App Session on the same host.

- This type of authentication is applied for jobs coming from a *Public SavaPage Printer*, or from a *Trusted SavaPage Queue* where the originating User is *Abstract*.
- When no authenticated Web App session is found the job is ignored.

11.1.14. Mail Print Authentication

*Deduction of the printing **Person** using the email address of the sender.*

- This type of authentication is applied for print jobs coming in from **Mail Print**.
- When no unique matching **Person** is found, or when the **Person** is disabled, authentication fails. Consult [Section 4.4.4, “Edit User” \[84\]](#) on how to mark a **User** as (enabled) **Person**.

11.1.15. Local User

*A **User** defined on a local device.*

11.1.16. Local Abstract User

*An **Abstract User** defined on a local device.*

11.1.17. Local Person

*A **Person** defined on a local device.*

11.1.18. User Alias

*An alternative name for a **User**.*

- A single **User** can have several aliases.
- An alias is applied at the following levels:
 - User login to the User and Admin Web App via the **Login dialog**, or the **XML Web Services API**.
 - Print jobs arriving in the SavaPage queues under the alias name.

For more information see [Section 11.4, “User Name Aliases” \[212\]](#).

11.2. Single Sign-On Domains

In a *Single Sign-On* (SSO) network user authentication is achieved in a login dialog where the **User** supplies his credentials, usually by entering his user name and password¹. SSO networks establish a web-of-trust between *users* and *domain services*. System administrators like SSO domains, because they provide a single interface to control access of **Domain Users** to servers and services.

Making SavaPage part of an SSO domain is the most sophisticated setup possible. In this way access to the SavaPage queues can be controlled on a domain defined user and group level, and by doing so we can create *authenticated queues*.

Authenticated SavaPage **IPP Print Queues** can exclusively be achieved in a macOS or GNU/Linux SSO domain using LDAP or NIS (Network Information Service) as authentication services².

In Windows Domains authenticated SavaPage **Print Queues** can solely be enforced by installing local printers connected to SavaPage by the **JetDirect/RAW** protocol. These RAW IP printers are typically installed on a Windows Print Server. To exclusively grant access to the SavaPage printer by the print server, enter the server IPv4 address (as **CIDR**) as the allowed client IP address in the Default / Queue definition. See [Figure 4.39, “Admin Web App: Queue - Edit” \[101\]](#).

¹ Of course methods like a smartcard and biometric (fingerprint) authentication should be mentioned as alternative.

² NIS (Network Information Service) protocol, also known as Sun Yellow Pages (YP) allows the information contained in the files `/etc/passwd`, `/etc/group` and `/etc/shadow` to be hosted on a central server. Administrators can enter, edit and delete the information on the NIS server so that it is automatically available on all Unix nodes. Authentication services are usually delegated to a Kerberos server, which thanks to tickets and authenticators eliminates the need to move passwords over an open and insecure network. NIS operates on “flat” domains and is therefore unsuitable for large organizations which due to their nature may be organized hierarchically. In such cases, the use of the LDAP (Lightweight Directory Access Protocol) is the way to go.

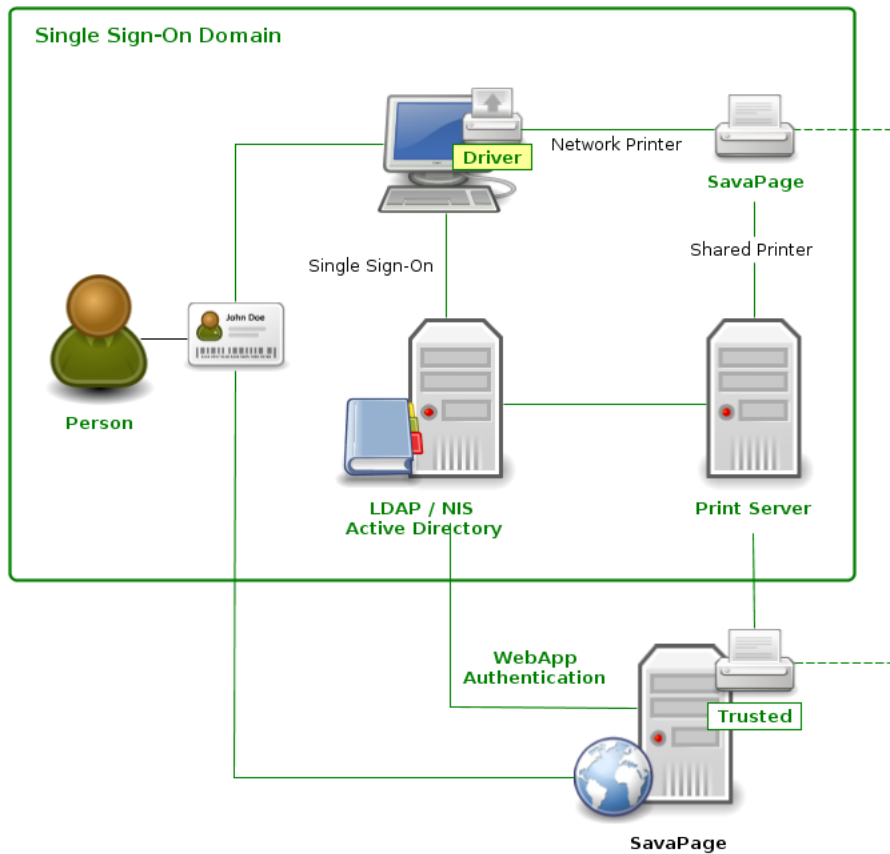


Figure 11.1. SavaPage in a Single Sign-On Domain

Trust is indeed essential to match the print job with the user's *SafePages* as previewed in his authenticated browser session. But, as we shall see in the next section, even in trust relations there are some loopholes to consider, and in networks where access is not fully guarded by SSO, unauthenticated users also need our special attention.

11.2.1. Authentication Loopholes

Although fully closed SSO domains provide unambiguous trust, there are common authentication loopholes that need to be addressed. These loopholes are generic in nature and not related to SavaPage.

1. A loophole is introduced when multiple users use the same account (user name) to authenticate to the network. Because the login is based on the person's *role* we can not retrieve the *unique user identity*. If for example, both John and Mary logged in with the generic *student* account, there is no way to find out if a SavaPage print job from this session was issued by John or Mary. By default the print jobs of John or Mary will end up in the *SafePages* of the one and only unknown *student*. In situations where printing content is private this might pose a problem. In SavaPage this loophole can be solved by marking the generic user account as *abstract*. See [Section 11.1.13, "IP Based Authentication"](#) [207].
2. A similar loophole is introduced when different users (sequentially) use the same machine, which was started in auto-login mode. Because the login is based on the *machine identity* we can not retrieve the *unique user identity*. In SavaPage this loophole is solved by the marking the auto-login user account as *abstract*. See [Section 11.1.13, "IP Based Authentication"](#) [207].

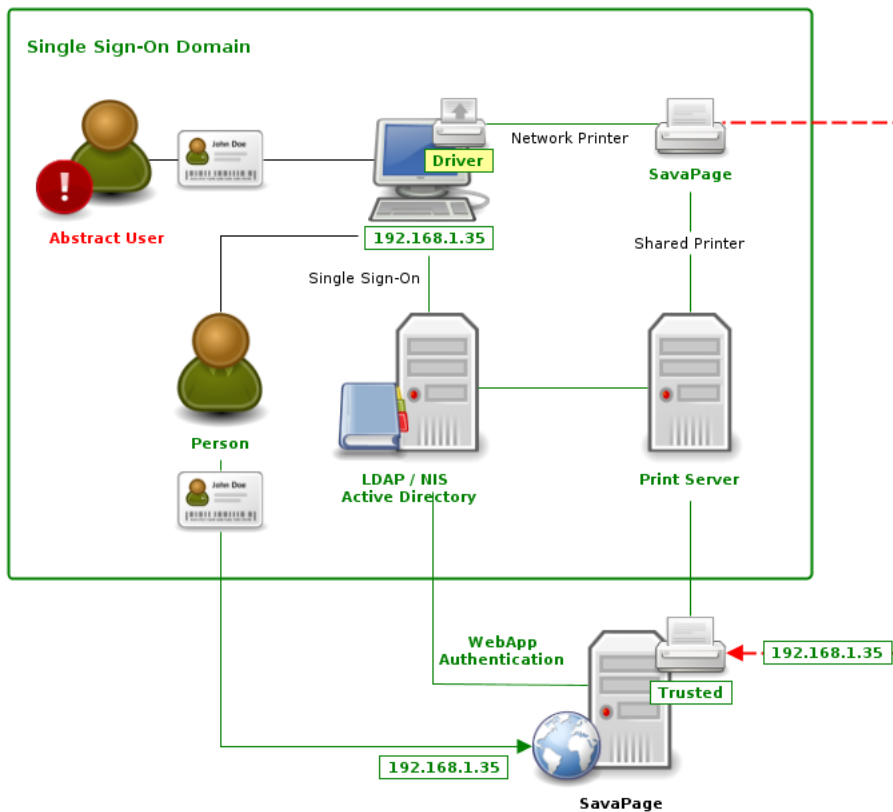


Figure 11.2. IP Based Authentication for Abstract User

11.2.2. Unauthenticated Users

In networks where access is not fully guarded by SSO, SavaPage queues introduce a security issue when they are used by unauthenticated non-domain users. For example, consider a guest user who connects his personal laptop to the network, and installs and prints to a SavaPage printer. In SavaPage this loophole can be solved by marking the *queue* as *untrusted*, i.e. by creating a **Public SavaPage Queue**. See Section 11.1.13, "IP Based Authentication" [207]. In addition the **Internal Users** feature can be used to offer out of domain Web App authentication for guest users.

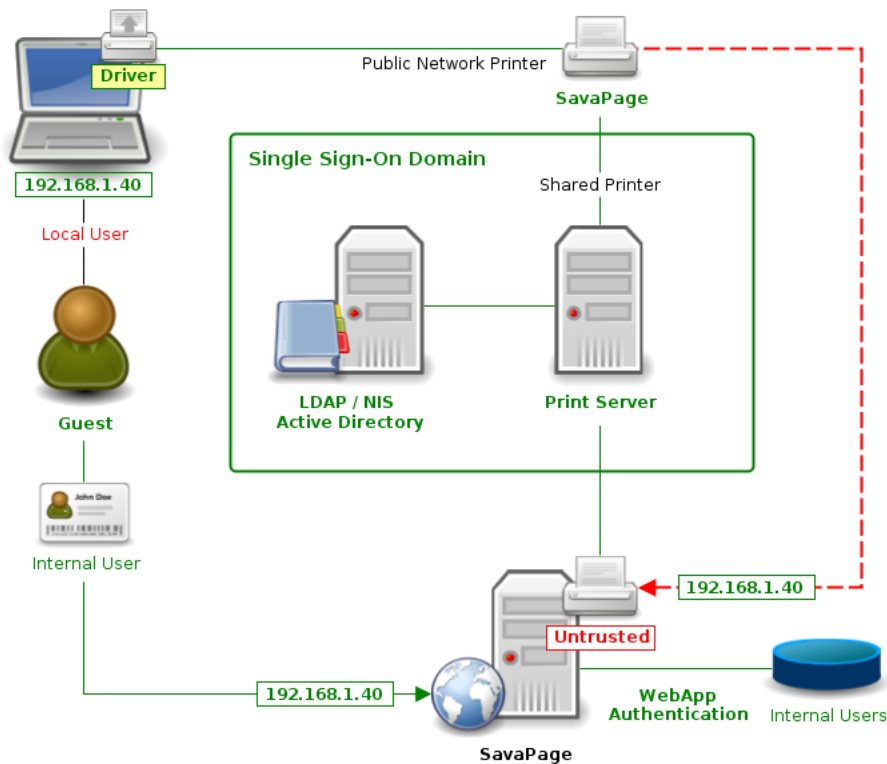


Figure 11.3. IP Based Authentication for Unauthenticated User

11.3. Peer to Peer Networks

A peer-to-peer or workgroup environment differs fundamentally from the network domain model. In the domain model, users authenticate with a unique (password protected) user name, as defined in a central server, while in a workgroup user identity is validated against a **Local User** rather than a central authority. The workstations are either set up to automatically login as a general "user", or user accounts are created locally as required.

Trust can be enforced by creating a **Public SavaPage Queue**. See [Section 11.1.13, "IP Based Authentication"](#) [207], and using the **Internal Users** feature for Web App authentication.

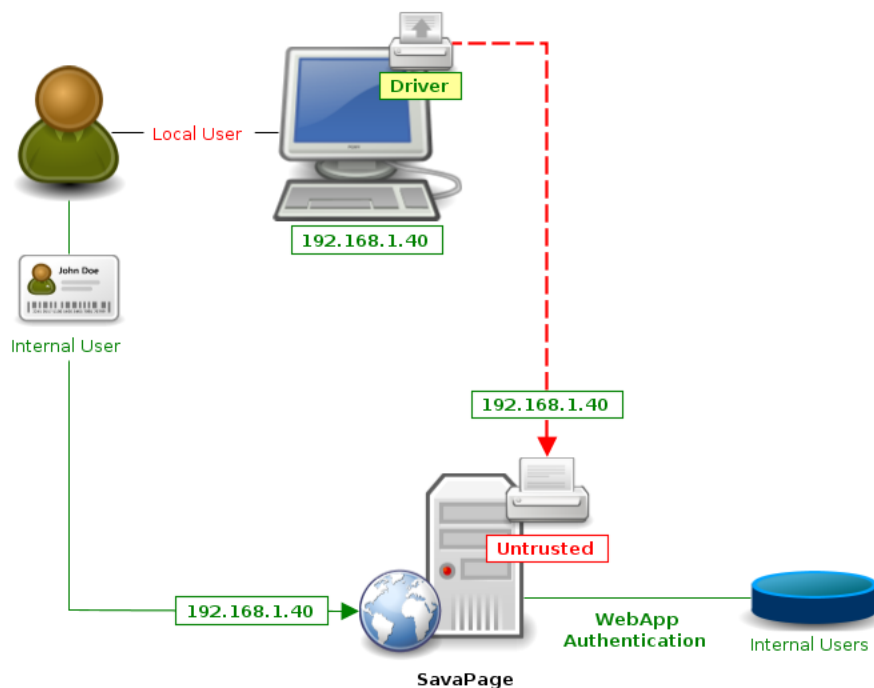


Figure 11.4. IP Based Authentication in Peer-to-Peer Network

11.4. User Name Aliases

A User Alias is a way of mapping a user name in one format to a name in another. It is useful in the following situations:

- Providing extra convenience for users to log into the system with a user name formatted in a different way. So Georg Friedrich Händel can have a **User Alias** of “georg_handel”, “gf_handel”, and “gfhandel”.
- Used as a temporary tool to manage domain or user name format changes. For example, you may have changed names from j.brown to john.brown. An alias can help forgetful users to log in with their old name.

Name aliases are applied at the following levels:

- User Login to the User and Admin Web App.
- Print jobs arriving under the alias name.

The aliases information is kept in the file,

```
/opt/savapage/server/data/conf/username-aliases.txt
```

and can be created based on the provided template file,

```
/opt/savapage/server/data/conf/username-aliases.txt.tmpl
```

You can create your own custom alias file as follows:

- Go to the directory `/opt/savapage/server/data/conf/`
- Open your favorite text editor with the file `username-aliases.txt.tmpl`
- For example, add the following lines to the end:

```
j.brown : john.brown
jbrown : john.brown
```

- Save file as `username-aliases.txt`

The format of the alias file is:

```
aliasname1: username1
aliasnameA: username2
aliasnameB: username2
```

where `aliasname` is mapped to `username` in the system database. A user may have multiple aliases. In this example, `username2` is known both as `aliasnameA` and `aliasnameB`. The separator between `aliasname` and `username` can be `“:”`, `“=”` or tab.



Warning

If an offered user name does not match an alias in the alias file, it is assumed it represents the user's real name. If this user is new to the system he might be created automatically in SavaPage, according to the user creation policy defined in the [Options → User Creation → On demand user creation](#) section of the Admin Web App. So please take care that your alias list is valid and up-to-date.

Chapter 12. Printing Impact



One of the goals of SavaPage is to reduce hard-copy printing by facilitating the use of soft PDF copies instead. Above that, if printing is needed after all, SavaPage offers easy n-up, gray-scale and duplex proxy-printing to reduce printing even more. Giving feedback to users about the costs and environmental impact of their printing habits is used to arouse awareness and achieve behavioral change.

12.1. Financial Impact

In any organization the costs of unrestricted access to office printers can be substantial. With **SavaPage Financial** feedback about the costs from different perspectives (User, ProxyPrinter, Period) is within reach. Future releases will indeed process log data, and present financial statistics from all possible angles.

12.2. Environmental Impact

Environmental issues like global warming, waste management, paper production and consumption are an area of debate and interest to many. Highlighting the environmental impact of printing is one of the ways to influence user behavior.

SavaPage uses the number of printed *Sheet Units* to calculate three impact metrics: trees and energy consumption, and carbon production.



Important

The default values SavaPage uses for environmental metrics can be the subject of debate. Of course you are free to set the metric to any value that works for you. Please inform us about facts and findings you feel confident about.

12.2.1. Printed Sheet Units

A *Sheet Unit* (SU) is the size equivalent of an A4 sheet. So,

```
A4 == 1.00 SU
A3 == 2.00 SU
A2 == 4.00 SU
A1 == 8.00 SU
```

and ...

```
A5 == 0.50 SU
A6 == 0.25 SU
A7 == 0.12 SU
A8 == 0.06 SU
```

Note that SU precision is 2 decimals. As environmental impact is concerned, A4 and US Letter sheets are handled as equivalent, so ...

```
US Letter == 1.00 SU
```

SavaPage uses the media size chosen by the user to calculate the printed Sheet Units of a Proxy Printer print job. Some print examples:

- 6 pages double-sided on A4 : 3 SU
- 6 pages double-sided on A3 : 6 SU
- 6 pages 2-up on A4 : 3 SU
- 6 pages 2-up double-sided on A4 : 2 SU



Warning

SavaPage is not able to anticipate printer intelligence, for instance, when a printer uses different trays (with different media sizes) for different page sizes within the job document.

12.2.2. Trees

This metric is the percentage of a tree used to produce the paper of the printed Sheet Units. The metric is adopted from [Conservatree.org](http://conservatree.org)¹ and is as follows:

- A prototypical tree is 40 feet tall and 6-8 inches in diameter.
- One tree makes 16.67 reams of copy paper or 8,333.3 SU.

The metric 83333 is set as default for the configuration key: `environment.sheets-per-tree`

12.2.3. Energy

This metric is the energy used to produce the paper of the printed Sheet Units. The metric is adopted from [Paperonline.org](http://paperonline.org)² and is as follows:

- Around 500 kWh of energy are required in Europe to make 200 kg of paper.
- So one A4 or Letter sheet of office paper costs 12.5Wh to manufacture³.

The metric 12.5 is set as default for the configuration key: `environment.watt-hours-per-sheet`

12.2.4. Carbon

This metric is the amount of CO₂ released for producing the paper of the printed Sheet Units. The metric is adopted from the Swiss Federal Institute of Technology Zurich⁴ and is as follows:

- One A4 or Letter sheet costs 5.1g of CO₂ to production.

The metric 5.1 is set as default for the configuration key: `environment.co2-grams-per-sheet`



Note

This metric takes into account the CO₂ produced as a byproduct of paper production only. It does not take into account the CO₂ related to the production and operation of the printer and the ink or toner cartridges. Defining broader system boundaries and tracking down all parameters involved requires a major effort, and is beyond the scope of this manual. Of course you are free to set the parameter value for this metric to any value that works for you.

¹ <http://conservatree.org/learn/EnviroIssues/TreeStats.shtml>

² http://www.paperonline.org/uploads/AskGuenter/MR_produce%20energy.pdf

³ A sheet of 5g requires $(5/200.000) * 500 = 0.0125 \text{ kWh} = 12.5 \text{ Wh}$.

⁴ http://www.umwelt.ethz.ch/dokument/factsheets/sustainable_conference_compensation_en.pdf

Chapter 13. Security



This chapter discusses how SavaPage secures sensitive user and application data, and how it communicates with external Information Providers.

13.1. User Authentication

This section discusses how user credentials are protected.

13.1.1. Login Passwords

This section is about the passwords and PIN codes entered in the Web App Login Dialog.



Tip

Users can use the HTTPS protocol for connecting to the Web App, so data is encrypted to and from the server.

13.1.1.1. User Domain Passwords

SavaPage does *not* store or cache user domain login passwords. These passwords are always checked real-time at the source.

13.1.1.2. Internal User Passwords

Passwords of **Internal Users** are stored as SHA1 hash in the database.

13.1.1.3. Internal Admin Password

The SHA1 hashed password of the internal administrator admin is stored in a text file located at `/opt/savapage/server/admin.properties`. Access to this file is restricted to the savapage user.

SavaPage installs with admin as initial password for user admin.



Tip

If you forgot the internal admin password, you can reset it by editing the `admin.password` property in the `/opt/savapage/server/admin.properties` text file. Ignore the existing HASH value. SavaPage will hash your password upon first use.

13.1.2. PIN Codes

User PIN codes are stored in the database as **encrypted secret**.

13.1.3. Authentication Tokens

When Authentication Persistence is enabled for Browser Local Storage, authentication tokens are stored in the “Local Storage” of the browser. See [Section 4.10.3, “User Authentication” \[126\]](#).

Separate authentication tokens are held for the User, Admin, POS and Job Tickets Web App context and the same token is used for different sessions (on different devices) of a single user. A explicit logout in the Web App destroys the token. Authentication tokens are managed in memory on the SavaPage server. So, when the server restarts all local tokens are implicitly invalidated.

13.1.4. One-time Authentication Tokens

A Trusted Third Party (TTP) can acquire a one-time token for Web App user authentication by calling an XML-RPC method. The expiration time of the token should be as short as possible to minimize the risk that an accidentally exposed token can be misused. See [Section C.2.1.1, “onetime-auth.createToken” \[266\]](#) for details.

13.1.5. User Dialog

When authentication fails a neutral "Authentication failed" message is communicated to the user to prevent “Account Enumeration” and “Guessable User Account”.

13.2. Access over Internet

Take extra care when SavaPage is accessible over public Internet as a result of enabled [Internet Print](#). Make sure that access to the [Admin Web App](#) is solidly secured.



Tip

Create a [Terminal Device](#) for the public IP address and configure [Custom User Login](#). Two factor authentication with Local NFC Card and PIN (with “Self Association” disabled) is a secure option when users have an associated NFC Card and [Local Card Reader](#).

13.3. Web Sessions

13.3.1. Web Session Timeout

When [Authentication Tokens](#) are *not* used, Web Sessions guard persistent authorized access to SavaPage.

For security reasons all sessions expire (timeout) after a certain period of inactivity. Each interaction with the Web App that results in a call to the SavaPage Web Server resets the inactivity timer. Closing the browser window will end the session. The default timeout periods for different login types are described in the table below:

Login type	Default value
Admin Web App	1440 minutes (24 hours)
User Web App	60 minutes (1 hour)

Table 13.1. Default Web Session Timeout Values

The timeout value (in *minutes*) can be changed using the configuration properties below. A value of 0 indicates that the session will never time out.

Configuration property	Description
web-login.admin.session-timeout-mins	Inactivity timeout for the Admin Web App

Configuration property	Description
web-login.user.session-timeout-mins	Inactivity timeout for the User Web App

Table 13.2. Web Session Timeout Configuration Properties

See [Section 4.10.14.10, “Config Editor” \[155\]](#) for information about changing configuration properties.

Changed inactivity timeout values take effect for new sessions. Note that some pages periodically refresh the page (or data on the page), such as the [Dashboard](#). A session will not time out if a browser is left on these pages, as it will be considered active.

13.3.2. Web Session Cookies

Session tracking cookies like JSESSIONID and BAYEUX_BROWSER are marked as *HttpOnly*. An HttpOnly cookie cannot be accessed by client-side APIs, such as JavaScript, and may therefore help mitigate certain kinds of cross-site scripting attacks.

13.4. SSL Passwords

During the install process, SavaPage generates a [self-signed key and certificate](#) issued for the host's machine name. This key is used by default when the system is accessed via HTTPS on port 8632. The password of the default-ssl-keystore is generated in `/opt/savapage/server/data/default-ssl-keystore.pw`. Access to this file is restricted to the savapage user.

The passwords for the [installed keystore](#) created from an imported [Existing SSL Certificate](#) are set in the `/opt/savapage/server/server.properties` file. Access to this file is restricted to the savapage user.

13.5. Secured JMX Connection

During the install process, SavaPage generates a dedicated self-signed key and certificate for securing JMX connections with SSL. The generated Java keystore is: `/opt/savapage/server/jmxremote.ks` with password `savapage`. Access to this file is restricted to the savapage user.

The JMX password for the admin user is held in `/opt/savapage/server/jmxremote.password`. Java needs the password to be provided in plain text, so access to this file is restricted to the savapage user.

The initial password is a random string generated by the installation process. It needs to be changed in the Admin Web App as described in [Section 4.10.14.3, “JMX Agent” \[150\]](#).



Note

The default SavaPage JMX port is 8639. This can be changed by editing the file:

```
/opt/savapage/server/jmxremote.properties
```

You need to [restart](#) SavaPage for this change to take effect.

13.6. Encrypted Secrets

Data in the database will not contain any explicit secret data. All secret data is stored encrypted with encryption keys held in file `/opt/savapage/server/data/encryption.properties`. This file is generated by SavaPage at initial installation and contains randomized encryption keys unique for this particular installation instance. Access to this file is restricted to the savapage user.



Important

Make a backup of `encryption.properties` immediately after installation and store it at a secure place, so you can restore it in case of a server crash or when you need to [migrate to a new server](#).



Caution

The `encryption.properties` file is *crucial* for decrypting secret data in the database and verifying the authenticity of [document signatures](#). So, when you restore a database backup on a different server, be sure to also restore this file.

The following secret data is stored encrypted in the database:

- Passwords.
- PIN codes.
- Authentication tokens.
- API keys.

13.7. Document Signature

SavaPage generates a digital signature for every document printed-out or downloaded. Digital signatures are generated using a cryptographic technique called Hash-based Message Authentication Code (HMAC)¹. The algorithm takes various output job attributes such as job time, user name, document name and UUID, and combines them with a secret key. The result is then passed through the MD5 digest algorithm. The resulting signature is unique to the document instance². The applied secret key ensures the *authenticity* of the signature.

The algorithm used is:

- `Digest = Hash(date time || user name || document name || document UUID)`
- `Signature = Hash(Digest || Key)`

where

- Key is a random string generated by SavaPage at initial installation. It is stored as `hmac.key` property in the `/opt/savapage/server/data/encryption.properties` file, which is also used for [Encrypted Secrets](#).
- Hash is the MD5 function.
- `date time` is formatted in ISO 8601 basic format from year to second (`yyyyMMddTHH:mm:ss`). The time is local time (not UTC). E.g. `20120906T151231`.



Note

The signature is stored in the database for future use.

13.8. User Client

The User Client is a notifier of personal user events: see [Chapter 7, User Client](#) [186]. The following security measures apply:

- An SSL connection is used to communicate with the server.

¹ <https://en.wikipedia.org/wiki/HMAC>.

² The SHA1 digest algorithm is a stronger alternative, but MD5 is secure enough for our application and generates shorter signatures, which are easier to enter as argument to find the matching document.

- The server only accepts SSL connections.
- An API key is used as client identification.
- Access can be restricted by client IP addresses.

13.9. Server Commands

The **Server Command** tool provides a command-line interface to SavaPage Server methods. The following security measures apply:

- Only users with read access to the `/opt/savapage/server/server.properties` file have the right to execute the command.
- An SSL connection is used to communicate with the server.
- An API key is used as client identification.
- SavaPage server accepts SSL connections from local host only. When remote access to SavaPage is proxied, e.g. by Apache redirect, the remote address will be 127.0.0.1 in all cases. Therefore, requests from local loop-back address 127.0.0.1 are **not** accepted. The request origin must be a "real" IP address identical to the one of the SavaPage server.
- At the client, SSL host name verification is turned *off*. This allows a mismatch between the server host name and the SSL certificate CN.

13.10. Web Services

Public **XML-RPC** and **JSON-RPC** services are protected by secure endpoints with API keys.

13.11. Log Files

Log files are located in `/opt/savapage/server/logs`. This directory can be accessed by user `savapage` only. Although log files may contain sensitive user and application data, they never contain secrets like passwords and access codes.

Log file rotation is fully customizable in `/opt/savapage/server/lib/log4j.properties`. See the sibling `log4j.properties.template` file for more information. A server **restart** is required for changes to take effect.

13.12. Network Card Reader

A setup where a *public* and *unattended* IP device communicates with a central server is inherently prone to security breaches. Our **Network Card Reader** device is no exception to this rule.

Although SavaPage validates the reader's IP address, the reader could be replaced and mimicked. Also, since communication is non-SSL, the Card Number (UID) of swiped NFC Cards could be hijacked. However, since the only content transmitted is the Card Number, misuse will be limited to a Card Number being offered from an unexpected origin at an unexpected time. Since offered Card Numbers are always processed in well defined transient contexts with short time limits, the risk of unnoticed abuse can be considered minimal.

A security breach of a fundamentally different nature is the rare scenario where it is possible to manipulate the UID of an NFC Card. A hacker could then use the hijacked card number to make a duplicate authentication token.



Tip

As an extra security measure *two-step authentication* can be implemented by requiring an additional PIN (over an SSL connection) after the initial NFC Card authentication. See [Section 4.10.3, “User Authentication” \[126\]](#) and [Section 4.9.3, “Custom User Login” \[118\]](#)

13.14. External Services

13.14.1. Google Cloud Print Service



13.15. Vouchers

- Check the [Application Log](#) regularly for unsuccessful (potentially fraudulent) voucher redemption attempts.
- Use the [Voucher List](#) to monitor successful voucher redemption.
- Delete or Expire a voucher batch immediately when vouchers are reported lost or stolen. See [Section 4.14.1, “Voucher Actions”](#) [170].



Caution

Voucher numbers are *not* encrypted in the database, so be careful to store [database backup](#) files at a safe location.

Chapter 14. Privacy

This chapter explains how digital freedom and privacy is secured in the SavaPage domain. The privacy domain partially overlaps security (confidentiality), which include the protection of information. See [Chapter 13, Security](#) [216].

14.1. Open Source

SavaPage is Open Source Software, and complies to Free Software as defined by the [Free Software Foundation](#)¹. The use of the [AGPL Software License](#) testifies of this compliance. Being free open source software, everyone has the freedom to study how SavaPage works and monitor how digital privacy is enforced.

14.2. General Data Protection Regulation

Digital Privacy is a fundamental right in the primary law of the European Union, and the [General Data Protection Regulation](#)² (GDPR) is one of the main legal instruments. GDPR uses the following terminology (in short):

- *Data Subject* : a natural person, as in [Section 11.1.2, “Person”](#) [206].
- *Personal Data* : information related to a Data Subject. See [Section 4.4.4, “Edit User”](#) [84] and [Section 3.8, “Log”](#) [60].
- *Data Controller* : an entity that controls the Personal Data, and determines the purpose and means of processing. Members and Visitors of the [SavaPage Community](#) are Data Controllers.
- *Data Processor* : an organization that processes the Personal data on behalf of the Data Controller (e.g. in an out-sourcing situation).

14.2.1. Data Portability

Data Portability is the Data Subject’s right to obtain information related to the processing of his Personal Data from the Data Controller. This allows the Data Subject to move, copy or transfer personal data easily from one IT environment to another.

SavaPage Data Portability is implemented as a download of personal data as CSV files in a ZIP container from the [User List](#) in the Admin Web App or optionally from a [GDPR Dialog](#) in the User Web App.

14.2.2. Data Erasure

Data Erasure is the Data Subject’s “right to be forgotten”. This right enables a person to request the removal of personal data when there is no compelling reason for its continued processing. The request can optionally be send from the [GDPR Dialog](#) in the User Web App.

Data Erasure is performed with a [Server Command](#). See [Section C.1.7, “eraseUser”](#) [258] and [Section 4.4.3, “Erased Users”](#) [84].

14.3. Secure Print Release

SavaPage supports several scenario's where documents are not printed directly, but only after the user walks up to the printer and authenticates himself. This enables the rightful owner to directly retrieve printed copies of private or confidential documents. See [Section 4.9.1.2, “Proxy Print Authentication”](#) [114].

¹ <https://www.fsf.org>

² https://ec.europa.eu/info/law/law-topic/data-protection_en

14.4. CUPS Privacy

CUPS makes “job-name” “job-originating-host-name” and “job-originating-user-name” private by default. This means that personal data are anonymized in the CUPS Web interface. See [Section 2.3.3, “CUPS Job Privacy” \[14\]](#).

Chapter 15. Internationalization



Internationalization is the process of designing a software application so that it can potentially be adapted to various languages and regions without engineering changes.

Localization is the process of adapting internationalized software for a specific region or language by adding locale-specific components and translating text.

SavaPage is internationalized software and can easily be localized to different languages, countries or regions.

15.1. Localization

The following localization rules apply:

- The Web App user interface localizes according to either the locale saved in the browser's local storage during a previous visit¹, or to the country/region code of the browser, or finally to the language chosen at [Login](#).
- Reports generated from the Web App localize according to the language of the Web App, except for [Vouchers](#) and Receipts at the [Point-of-Sale](#) and [User Web App](#), which use the [System Locale](#).
- The [System Locale](#) is applied to all system messages as broadcasted to the Admin Web App [Dashboard](#), written to the Application Log, or send by email.
- Text in log and audit files is fixed to the English language. So, when in need for support, these files can easily be understood by SavaPage Support.
- Dates and numbers are formatted according to the localization context.
- The currency symbol of the localization context is used.
- When localized text is not found the fall-back language will be English. So, in case SavaPage is partially translated for a selected locale, the user experience will be fragmented, as part of the text will fall back to English.

Currently SavaPage is fully localized to *English*, *German* and *Dutch*.



Tip

If you are interested in localizing the software to your region, please let us know.

SavaPage handles all localized text and user entered data as Unicode. The Web Browser, and therefore the Web App, natively displays Unicode correctly. However, the correct display of Unicode in PDF reports, needs special attention. Therefore, [Internal Fonts](#) are available to customize PDF generation.

15.2. Internal Fonts

The Unicode range of the displayed text in PDF documents must be covered by the embedded font.

Unfortunately, at present there is no native *outline* font that can display all Unicode characters. The one exception is [GNU Unifont](#), which *does* support the full 65,536 Unicode code point range. However, the glyphs originate from a *bitmap* of 16 pixels high and either 8 or 16 pixels wide, which gives the font a coarser look.

¹After login, the locale of the WebApp is saved in the browser's local storage, together with the [Authentication Tokens](#).

SavaPage contains internal fonts covering specific Unicode Scripts². These fonts can be selected to customize PDF output to the content locale.

15.2.1. Default Font

The default font for PDF output is **DejaVu Sans**³, which supports a broad set of Unicode scripts:

- Latin (including European and African alphabets, IPA, ...)
- Greek (including polytonic)
- Cyrillic
- Armenian
- Georgian
- Hebrew
- Arabic
- N'ko
- Lao
- Canadian Aboriginal Syllabics
- Ogham
- Tifinagh
- Lisu

Next to that, DejaVu also contains a lot of mathematical and other symbols, arrows, braille patterns, etc.



Tip

Coverage of the default font can be seen in `DejaVuSans.pdf`⁴.

15.2.2. CJK Font

Support for Chinese, Japanese and Korean (CJK) is provided by the **Droid Sans "fallback" font**⁵. This font contains over 43,000 glyphs and includes support for Simplified Chinese (GB2312), Traditional Chinese (Big 5), Japanese (JIS 0208) and Korean (KSC 5601). The font uses the Simplified Chinese ideographs for shared Unicode code points.



Tip

Coverage of the CJK font can be seen in `DroidSansFallbackFull.pdf`⁶.

15.2.3. Unifont

GNU Unifont⁷ is a Unicode font with a glyph for every visible Unicode Basic Multilingual Plane code point. The Unicode Basic Multilingual Plane covers the first 65,536 (or 2¹⁶) Unicode code points. GNU Unifont originates from a bitmap font with glyphs of 16 pixels high and either 8 or 16 pixels wide. Therefore it has a coarser look than native outline fonts.

² <http://www.unicode.org/charts/>

³ <http://dejavu-fonts.org>

⁴ <https://www.savapage.org/docs/fonts/DejaVuSans.pdf>

⁵ <http://www.droidfonts.com/droidfonts>

⁶ <https://www.savapage.org/docs/fonts/DroidSansFallbackFull.pdf>

⁷ <https://savannah.gnu.org/projects/unifont>

Chapter 16. Customization



SavaPage can be customized to fit your corporate identity. Customization makes SavaPage an integral part of your organization rather than an external tool.



Note

Customization is an advanced topic. If you need help, please contact your SavaPage Community Representative.

16.1. Custom Web App

Web App customization is controlled in the `/opt/savapage/server/custom/web.properties` file. An annotated `web.properties.template` file is installed for your convenience.



Tip

Each key value in the `/opt/savapage/server/custom/web.properties` file can be overruled at runtime by specifying the key value in the [Configuration Editor](#). When the configuration key value is left empty customization falls back to the value in the properties file.

16.1.1. Web App Look-and-feel

The look-and-feel of Web Apps can be customized by theming and CSS tailoring.

16.1.1.1. Web App Theming

SavaPage uses [jQuery Mobile](#)¹ as user interface system to create responsive Web Apps that are accessible on all smart-phone, tablet and desktop devices. jQuery Mobile supports theming. Themes can be built online with the [ThemeRoller for Mobile](#)² tools and deployed in SavaPage by downloading the zipped theme file and extracting the content of the `/themes/` folder into the `/opt/savapage/server/custom/web/themes` directory.

The `web.properties` file contains entries to specify a separate CSS theme for each Web App, as shown in the example below:

```
webapp.theme.admin=admin.min.css           ❶
webapp.theme.jobtickets=jobtickets.min.css  ❷
webapp.theme.pos=admin.min.css              ❸
webapp.theme.user=user.min.css             ❹
```

- ❶ CSS theme file name for the [Admin Web App](#).
- ❷ CSS theme file name for the [Job Tickets Web App](#).
- ❸ CSS theme file name for the [POS Web App](#).
- ❹ CSS theme file name for the [User Web App](#).

¹ <https://jquerymobile.com>

² <https://themeroller.jquerymobile.com>

SavaPage uses swatch³ “a” for all pages and dialogs. Swatch “b” is used for page and dialog headers, and in some cases for list dividers.

You can store a theme in a subdirectory of `/opt/savapage/server/custom/web/themes` and use its relative path to reference a CSS theme file.

16.1.1.2. Custom CSS

Advanced tailoring can be done with custom CSS files. They are rendered as last, so they have the final say about styling.

The `web.properties` file contains entries to specify a custom CSS file for each Web App, as illustrated in the example below:

```
webapp.custom.admin=admin.css      ❶
webapp.custom.jobtickets=jobtickets.css  ❷
webapp.custom.pos=pos.css          ❸
webapp.custom.user=user.css        ❹
```

- ❶ Custom CSS file for the **Admin Web App**.
- ❷ Custom CSS file for the **Job Tickets Web App**.
- ❸ Custom CSS file for the **POS Web App**.
- ❹ Custom CSS file for the **User Web App**.

Custom CSS files are stored in `/opt/savapage/server/custom/web/`. Subdirectories are allowed, and you can use their relative path to reference the custom CSS file.

Any content placed in `/opt/savapage/server/custom/web/`, such as images, can be accessed in CSS via a URL beginning with `/custom/web/`. For example if a file named `logo.png` is placed in `/opt/savapage/server/custom/web/images` it can be accessed via the URL `/custom/web/images/logo.png`.

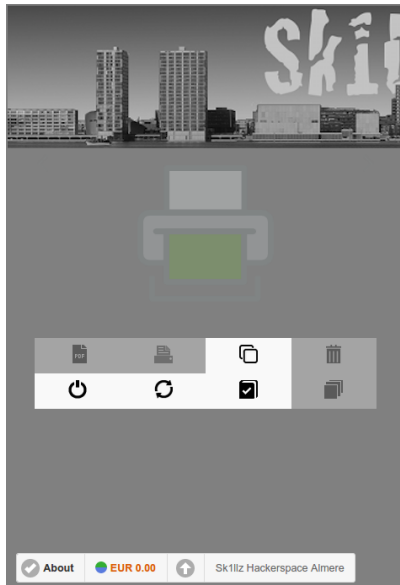


Figure 16.1. User Web App: Custom CSS - Sample #1

³A swatch is one of several colour schemes that can be provided by a jQuery Mobile theme. Single-letter designations are used for swatches. The default theme provides two swatches. The “a” swatch is a neutral, gray swatch, and the “b” swatch has a darker color scheme designed to contrast with the “a” swatch. Swatch “b” is used to draw special attention to certain elements in a user interface styled with “a”.

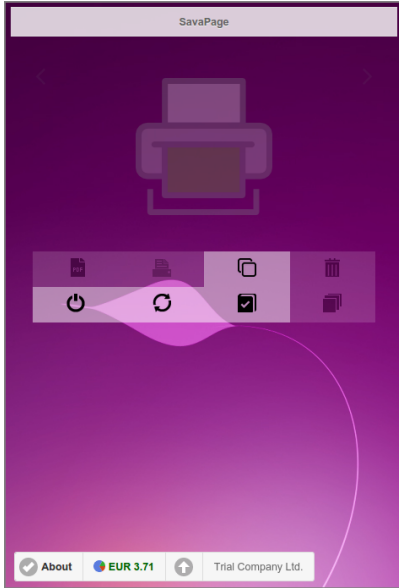


Figure 16.2. User Web App: Custom CSS - Sample #2

16.1.1.3. Custom HTML

Extra tailoring can be done with HTML snippet files to be injected into the Web App. Injection points are defined at the top of the Login and About Page for each Web App. Snippet files must be placed in the `/opt/savapage/server/custom/html/` directory. The default snippet is for the English locale. You can create i18n variants by appending the locale to the base file name. For example: `user-login_de.html` is the German variant of `user-login.html`.

Snippet files are assigned in the `web.properties` file, as shown below. Snippets must refer to the default English language variant. At runtime the locale variant (when applicable and available) is used.

```
webapp.html.admin.about=admin-about.html      ❶
webapp.html.admin.login=admin-login.html       ❷

webapp.html.jobtickets.about=jobtickets-about.html  ❸
webapp.html.jobtickets.login=jobtickets-login.html  ❹

webapp.html.pos.about=pos-about.html            ❺
webapp.html.pos.login=pos-login.html            ❻

webapp.html.user.about=user-about.html          ❼
webapp.html.user.login=user-login.html          ❽
```

- ❶ snippet for **Admin** About Dialog.
- ❷ snippet for **Admin** Login Page.
- ❸ snippet for **Job Tickets** About Page.
- ❹ snippet for **Job Tickets** Login Page.
- ❺ snippet for **POS** About Dialog.
- ❻ snippet for **POS** Login Page.
- ❼ snippet for **User** About Dialog.
- ❽ snippet for **User** Login Page.

You can store custom HTML files in a subdirectory of `/opt/savapage/server/custom/html/` and use its relative path to reference the HTML file.



Important

Please use this template when creating snippets for Login pages. The CSS classes are needed to toggle visibility of sub-parts in different Login modes.

```
<h3 class="sp-login-dialog">Header when in Login Mode</h3>
<h3 class="sp-login-dialog-assoc">Header when in Card Self Association Mode</h3>
<div class="sp-login-dialog">
  <!-- Your custom HTML when in Login Mode -->
</div>
```

16.1.1.4. Custom i18n

In rare cases an application manager might want to override i18n text of Web App pages. Overrides of packaged *.properties.xml files must be placed in /opt/savapage/server/custom/i18n/. Each override must have *.xml name (without the ".properties." part) and be located in a subdirectory with a path identical to the original *.properties.xml file. The content of the override file can be restricted to just the overridden keys.

Example:

```
/opt/savapage/server/custom/i18n/org/savapage/server/pages/Login_de.xml
```

Custom i18n must be activated in `server.properties` with:

```
webapp.custom.i18n=true
```



Warning

Custom i18n is an advanced feature and should be implemented in consultation with SavaPage Tech Support.

16.1.1.5. Custom i18n for IPP

IPP attribute localization can be overridden with special i18n XML files. See [Section L.3, “IPP Localization” \[310\]](#).

16.2. Email Templates

Email templates are a powerful instrument to customize layout and content of email messages. Templates are defined as XML files, and are located in the /opt/savapage/server/custom/template/ directory by default ([alternative locations](#) can be configured).

Default templates are present in the SavaPage i18n jars. Therefore, after a first-time installation the custom template/ directory will be empty. However, when SavaPage finds a suitable i18n XML file in the custom directory, that belongs to an [Email Message Type](#), it will use that template.

You can create i18n variants of template XML files by appending the locale to the base file name. For example: template_de.html is the German variant of template.xml.

16.2.1. Email Template Syntax

The base syntax of an Email Template is described by example in the following XML file:

```
<?xml version= "1.0" encoding= "UTF-8" ?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd"> ❶
```

```

<properties>
  <!-- $obj.attr$ placeholders are replaced at runtime --> ❷

  <!-- TEXT --> ❸
  <entry key="text"><![CDATA[Text with $obj.attr$ placeholders.]]></entry>

  <!-- HTML --> ❹
  <entry key="html"><![CDATA[<b>HTML</b><p>
    with $obj.attr$ and other placeholders</p>]]>
  </entry>
</properties>

```

- ❶ DTD⁴ of XML persisted `java.util.Properties` file.
- ❷ Placeholder object attributes are identified by a `$` character at the beginning and the end. Valid objects are defined for each **Email Message Type**. The global **Application Object** is valid for all types.
- ❸ Content for plain text message body.
- ❹ Content for html message body.

16.2.1.1. Embedded Images

HTML content may contain Embedded (Inline) Images.

Images are first defined in separated template entries with keys that have “cid_” prefix. Those keys can then be used as placeholders for cid scheme values. SavaPage email processor will embed the image file and replace cid placeholders with a unique Content-ID for the Embedded (Inline) Image.

Custom image files are referred to by their relative path from the custom `template/` directory. Internal stock images can simply be referred to by their upper-case identifier, as shown in the table below.

Object.attribute	Value
SAVAPAGE_ICON	SavaPage Icon: 32 x 32 pixels, 705 bytes.
SAVAPAGE_LOGO	SavaPage Logo with "SavaPage" text at the bottom: 148 x 174 pixels, 5.6 kB.

Table 16.1. Stock Image Identifiers

The XML snippet below shows how it works:

```

<entry key="cid_1">SAVAPAGE_ICON</entry> ❶
<entry key="cid_2">images/mysite-logo.png</entry> ❷

<entry key="html"><![CDATA[
  
    
  </a>
]]></entry>

```

- ❶ `cid_1` holds a stock image.
- ❷ `cid_2` holds the relative path to a custom image file.
- ❸ Note that the URL of the image has scheme `cid`. SavaPage email processor will embed the image and replace `cid_1` with a unique Content-ID for the Embedded (Inline) Image with key `cid_1`.

⁴ <http://java.sun.com/dtd/properties.dtd>

- ④ The `cid_2` image is handled just as its `cid_1` sibling.

16.2.2. Email Stationary Template

An Email Stationary Template is used by an [Email Message Template](#) as container to embed its content in. A stationary is typically used by many [Email Messages Types](#), and thus gives a common look-and-feel to various email output.

A Stationary Template contains the following placeholder objects:

- [Stationary](#).
- [Application](#).

And this is how it looks like:

```
<?xml version= "1.0" encoding= "UTF-8" ?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">

<properties>

    <!-- TEXT -->
    <entry key="text"><![CDATA[Plain text with $stationary.header$ ❶
$stationary.content$ and additional app placeholders]]></entry> ❷

    <!-- HTML -->
    <entry key="html"><![CDATA[HTML with <b>$stationary.header$</b>
    <p>$stationary.content$ and additional cid_ and app placeholders</p>]]> ❸
    </entry>

</properties>
```

- ❶ `$stationary.header$` is replaced by the `header` entry of the embedded template.
- ❷ `$stationary.content$` is replaced by the `text` entry of the embedded template.
- ❸ `$stationary.content$` is replaced by the `html` entry of the embedded template.



Note

SavaPage uses its own stationary types. These types can be overwritten. See [Section 16.2.5, “Email Stationary Types”](#) [234].

16.2.3. Email Message Template

Email Message Templates are used by [Email Message Types](#).

Template and Type are tied by name.

For example: the [JobTicketCompleted](#) message type will look for the nearest `il8n` version of a `JobTicketCompleted.xml` template file. In this way, `JobTicketCompleted_de.xml` will be the perfect match when a German email message is requested.

Email Message Template adds the entry key “subject” to the syntax. Optionally, entry keys “stationary” and “header” can be used to link to an [Email Stationary Template](#).

The syntax is described by example in the following XML file:

```
<?xml version= "1.0" encoding= "UTF-8" ?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
```

```

<!-- Email subject --> ❶
<entry key="subject">Text with $obj.attr$ placeholders</entry>

<!-- Optional stationary --> ❷
<entry key="stationary">EmailStationary</entry>
<entry key="header">Stationary header with placeholders</entry>

<!-- TEXT -->
<entry key="text"><![CDATA[Plain text with placeholders.]]></entry>

<!-- HTML -->
<entry key="html"><![CDATA[<p>HTML with placeholders.</p>]]></entry>

</properties>

```

- ❶** The subject text of the email.
- ❷** The stationary value (just the template file basename, without the .xml suffix) refers to EmailStationary.xml, including i18n EmailStationary_*.xml variants. When a stationary is linked this way, the header value is injected into the stationary.header placeholder of the target stationary, and the text and html values are injected into the stationary.content placeholder of their counterpart stationary entry.

16.2.4. Email Placeholders Objects

Placeholder Objects are used in [Email Stationary](#), and in [Email Message Templates](#) tied to [Email Messages Types](#).

16.2.4.1. Stationary

Object uniquely used by [Email Stationary Template](#).

Object.attribute	Value
stationary.header	The header entry of the embedded template.
stationary.content	The text or html entry of the embedded template.

Table 16.2. Placeholder: Stationary

16.2.4.2. Application

A global object with SavaPage application attributes. This object can be used in *any* [Email Message Type](#) context.

Object.attribute	Value
app.name	Application name: “SavaPage”
app.nameVersion	Formatted name and version, like: “SavaPage Major.Minor.Revision”
app.nameVersionBuild	Formatted name, version and build, like: “SavaPage Major.Minor.Revision (Build)”
app.slogan	Short slogan describing the application: “Open Print Portal”

Table 16.3. Placeholder: Application

16.2.4.3. Ticket

A Print Job Ticket. See [Chapter 5, Job Tickets Web App \[174\]](#).

Object.attribute	Value
<code>ticket.number</code>	Ticket number.
<code>ticket.name</code>	Document name.
<code>ticket.operator</code>	Name of the Job Ticket Operator.
<code>ticket.returnValue</code>	The return message to Job Ticket Creator (optional).

Table 16.4. Placeholder: Ticket

16.2.4.4. User

A User as [Person](#).

Object.attribute	Value
<code>user.fullName</code>	The full name of the user.

Table 16.5. Placeholder: User

16.2.5. Email Stationary Types

SavaPage uses its own Email Stationary Types. As the table below shows, currently there is just one type in use. You can use this type in your own custom [Email Message Template](#) files. Or, you can override this type by creating your own `EmailStationary.xml` variants in the custom `template/` directory.

Name	Scope
EmailStationary	All Email messages

Table 16.6. Email Stationary Types

16.2.6. Email Message Types

The sections below describe email message types that can be customized. The list is limited for now, but will grow as more messages are refactored for this purpose.



Note

The global [Application Object](#) is valid for each message type.

16.2.6.1. JobTicketCanceled Email

Email sent when a Job Ticket is canceled. See [Table 3.3, “Job Ticket Print Configuration Properties” \[53\]](#).

The following placeholder objects apply:

Object	Role
<code>ticket</code>	Canceled Job Ticket.

Object	Role
<code>user</code>	Job Ticket Creator.

Table 16.7. Placeholder Objects: JobTicketCanceled

16.2.6.2. JobTicketCompleted Email

Email sent when a Job Ticket is completed successfully. See Table 3.3, “Job Ticket Print Configuration Properties” [53].

The following placeholder objects apply:

Object	Role
<code>ticket</code>	Completed Job Ticket.
<code>user</code>	Job Ticket Creator.

Table 16.8. Placeholder Objects: JobTicketCompleted

16.2.7. Custom Template Locations

Custom templates are located in the `/opt/savapage/server/custom/template/` directory. Sub-locations can be defined for different template groups. These sub-locations can then be activated for different applications, with the following configuration properties:

Configuration property	Description
<code>custom.template.home</code>	Home location of <i>all</i> template files, relative to the default location. For example: a value of “group-1” will resolve to: <code>/opt/savapage/server/custom/template/group-1</code>
<code>custom.template.home.mail</code>	Home location of Email template files, relative to the default location. For example: a value of “group-1/mail” will resolve to: <code>/opt/savapage/server/custom/template/group-1/mail</code>

Table 16.9. Configuration Properties for Custom Template Locations

See Section 4.10.14.10, “Config Editor” [155] on how to set these items.

Chapter 17. Using an External Database



By default SavaPage is packaged with [Apache Derby](https://db.apache.org/derby/)¹ as *internal* database. This gives you the opportunity to evaluate SavaPage on a small scale right away. However, in a production environment with multiple users, we strongly advise you to use [PostgreSQL](https://www.postgresql.org/)² as *external* database server.



Warning

Using the internal database in situations with multiple users and thus concurrent use, will inevitably lead to locking, deadlock and out-of-memory errors, which can make the system totally unresponsive.

Other situations can be extra reason to choose for an external database, like:

- Organizational policy dictates that all applications must be consolidated on a single database infrastructure.
- You want to take advantage of existing maintenance and backup procedures that are present on your current database infrastructure.
- You want to use third party reporting tools to view and analyze the SavaPage database.
- You want optimal (tailored) performance, since SavaPage is intensively used by a very large user population. So, for example, you want to deploy a dedicated database server as a scalable solution.

This chapter describes how to connect and migrate to an external database.

17.1. Supported Databases

SavaPage is able to use any database that has a JDBC driver available. *However, we choose to support PostgreSQL on GNU/Linux servers only.* PostgreSQL is designed to be highly scalable, is optimized for concurrent use, and handles datasets of any size efficiently. We have used it in large scale SavaPage implementations and see it behave robust with thousands of users. Above that, PostgreSQL is free and open source software, and complies to open standards.

17.2. Migrating to an External Database

The migration is a simple process and takes about 15-30 minutes. The sections below describe in more detail the following high-level migration scenario steps:

1. Stop the server.
2. Create a backup of the current internal database.
3. Create and initialize a new external database.
4. Restore the backup into the new external database.
5. Restart the server.

17.2.1. Step 1 - Stop SavaPage

The application server must be stopped in order to make a backup of the current internal database. The command to stop the server is described in [Section C.4, “Stopping and Starting the Server”](#) [272].

¹ <https://db.apache.org/derby/>

² <https://www.postgresql.org/>

17.2.2. Step 2 - Create a Backup

Run the command to backup the current (internal) as described in [Section C.3.4, “db-export and db-export-to” \[271\]](#). The command echoes the name of the created backup file to `stdout`. Take a note of this because you will need this in a future step.

17.2.3. Step 3 - Create new Database in External DBMS

Creating a new database is specific to the external Database Management System and is off-topic for this manual. It is assumed that the database administrator knows how to create a new database. However, the following generic requirements must be honored:

- Create a *dedicated* database user with a strong password to be used by SavaPage to connect to the database.
- Create the new empty database with a Unicode or UTF8 character encoding to make sure that all possible characters can be stored.
- Assign the dedicated user full access to the new database, i.e. grant permission to create and drop tables, and to execute select, insert, update and delete statements in all tables.

17.2.4. Step 4 - Change SavaPage Connection Parameters

Open a terminal session on the SavaPage server as user `savapage` and edit the `/opt/savapage/server/server.properties` file.

- Comment out the line `database.type=Internal` by adding a hash (#) character at the start of the line.
- Uncomment the `database.connection` parameter lines for the external database (in our case PostgreSQL).
- Set the `database.url`, which describes the location and connection details of the external database.

The PostgreSQL URL format is: `jdbc:postgresql://[server]/[database]`

The `[server]` parameter is the name of the server running the PostgreSQL database, and must be resolvable from the SavaPage server. If the PostgreSQL instance is running on the same machine then `localhost` can be used. The `[database]` parameter is the name of the PostgreSQL database you created in the previous step.

- Set the `database.user` and `database.password` used to connect to the database.

A connection example is shown below:

```
#-----
# Database Settings
#-----

# Using the internal database (default)
#database.type=Internal

# PostgreSQL connection
database.type=PostgreSQL
database.driver=org.postgresql.Driver
database.url=jdbc:postgresql://localhost/savapage
database.user=your-db-user
database.password=your-db-user-password
```

17.2.5. Step 5 - Initialize new Database

This step initializes the new database, it creates the required database tables and initial data.

To initialize the database, open a terminal session on the SavaPage server as user `savapage` and run the command as described in [Section C.3.6, “db-init” \[272\]](#).

17.2.6. Step 6 - Restore Backup into new Database

This step restores the backup file exported in the previous step, into the newly initialized database. Open a terminal session on the SavaPage server as user `savapage` and run the command as described in [Section C.3.5, “db-import”](#) [271].

17.2.7. Step 7 - Restart SavaPage

At this point the data have been migrated to the new database and the server can be restarted. See [Section C.4, “Stopping and Starting the Server”](#) [272].

Wait a couple of seconds before logging in to the [Admin Web App](#) to verify that the migration worked successfully.

Chapter 18. Tuning



18.1. Linux Kernel Parameters

GNU/Linux distributions are generally not configured to run more demanding server processes out-of-the-box. So, running SavaPage with high load on a vanilla GNU/Linux OS can easily result in a degraded performance.

Performance bottlenecks are usually due to OS, TCP stack and network settings meant for *desktop* user sessions, and not for server processes that are intensively used by many network clients. Fortunately, it is easy to unleash the full potential of your GNU/Linux host with a few simple tweaks. The message is that SavaPage scales perfectly if you apply the right kernel settings.

Relevant kernel parameters and settings are discussed in the next sections. The last section summarizes the suggested settings and describes how to apply them. See [Section 18.1.5, “Setting Linux kernel parameters with sysctl” \[241\]](#).



Note

Kernel parameters with `ipv4` in their names also apply to TCP over IPv6.

18.1.1. IP Ports

As many outgoing connections are concurrently established from SavaPage, we must make sure Linux does not run low on *ephemeral* local ports¹ and reuse sockets with state `TIME_WAIT`.

```
net.ipv4.ip_local_port_range = 1024 65535 ❶
net.ipv4.tcp_tw_recycle = 0 ❷
net.ipv4.tcp_tw_reuse = 1 ❸
```

- ❶ Broaden the ephemeral local port range.
- ❷ Disable recycling of sockets with state `TIME_WAIT`.
- ❸ Enable the reuse of sockets with state `TIME_WAIT`. This is particularly useful in environments where numerous short connections are open and left in `TIME_WAIT` state, such as in SavaPage.



Note

According to *Vincent Bernat* in [Coping with the TCP TIME-WAIT state on busy Linux servers](#)²:

“On the server side, do not enable `net.ipv4.tcp_tw_recycle` unless you are pretty sure you will never have NAT devices in the mix. Enabling `net.ipv4.tcp_tw_reuse` is useless for incoming connections.”

“On the client side, enabling `net.ipv4.tcp_tw_reuse` is another almost-safe solution. Enabling `net.ipv4.tcp_tw_recycle` in addition to `net.ipv4.tcp_tw_reuse` is mostly useless.”

¹An established TCP/IP connection can be regarded as a 4-tuple (server IP, server port, client IP, client port). Three of the four are evident, i.e. the client uses its own IP address to connect to the server's IP address and service port. However, the connection also needs a port number at the client side. Unless the client program explicitly requests a port number, this port number is called an *ephemeral* port number. Ephemeral ports are temporary issued by the IP stack of the client OS from a dedicated port range.

² <https://vincent.bernat.im/en/blog/2014-tcp-time-wait-state-linux.html>

18.1.2. TCP Buffer Sizes

Linux does a good job of auto-tuning the TCP buffers, but the default maximum sizes are still very small. Here are sample settings for 1Gb and 10Gb network.

```
# Settings for 1Gb network (16Mb buffer)
net.core.rmem_max = 16777216      ❶
net.core.wmem_max = 16777216      ❷
net.ipv4.tcp_rmem = 4096 87380 16777216  ❸
net.ipv4.tcp_wmem = 4096 16384 16777216  ❹

# Settings for 10Gb network (32Mb buffer)
net.core.rmem_max = 33554432
net.core.wmem_max = 33554432
net.ipv4.tcp_rmem = 4096 87380 33554432
net.ipv4.tcp_wmem = 4096 16384 33554432

# Settings for 10Gb network (54Mb buffer)
net.core.rmem_max = 56623104
net.core.wmem_max = 56623104
net.ipv4.tcp_rmem = 4096 87380 56623104
net.ipv4.tcp_wmem = 4096 16384 56623104
```

- ❶ Max size (bytes) of the TCP receive buffer as settable with `setsockopt`.
- ❷ Max size (bytes) of the TCP send buffer as settable with `setsockopt`.
- ❸ Auto-tuning limits (bytes) for TCP receive buffer: min, default, and max number of bytes.
- ❹ Auto-tuning limits (bytes) for TCP send buffer: min, default, and max number of bytes.

18.1.3. Queue Sizes

While a socket is listening and busy, new connection requests will pile up. The kernel keeps pending connection requests in a buffer. When the buffer is full new requests will fail. You can increase several buffer sizes.

```
net.core.somaxconn = 4096      ❶
net.core.netdev_max_backlog = 16384  ❷
net.ipv4.tcp_max_syn_backlog = 8192  ❸
net.ipv4.tcp_syncookies = 1      ❹
```

- ❶ Max number of queued connections on a socket. The default of 128 is too low: we raise this value substantially to support bursts of request.
- ❷ Max number of packets, queued on the input side, when the interface receives packets faster than the kernel can process them.
- ❸ Max number half open SYN requests to keep in memory.
- ❹ Enable **SYN cookies**³ to harden the TCP/IP stack against SYN floods.

18.1.4. Congestion Control

Congestion refers to a network state where a node or link carries so much data that it may deteriorate network service quality, resulting in queuing delay, frame or data packet loss and the blocking of new connections.

In a congested network, response time slows with reduced network throughput. Congestion occurs when bandwidth is insufficient and network data traffic exceeds capacity.

Linux supports pluggable congestion control (avoidance) algorithms. To get a list of congestion control algorithms that are available in your kernel run the command:

³ https://en.wikipedia.org/wiki/SYN_cookies

```
sudo sysctl net.ipv4.tcp_available_congestion_control
```

If `cubic` and/or `htcp` are not listed then you will need to research the control algorithms for your kernel. If available set the control to `cubic`:

```
net.ipv4.tcp_congestion_control = cubic
```

18.1.5. Setting Linux kernel parameters with sysctl

Edit the file `/etc/sysctl.conf` like this:

```
sudo vi /etc/sysctl.conf
```

and add the following lines, that summarize the previously discussed kernel parameters, at the end of the file:

```
#-----
# SavaPage Settings for 1Gb network
#-----
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.ipv4.tcp_rmem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 16384 16777216
net.core.somaxconn = 4096
net.core.netdev_max_backlog = 16384
net.ipv4.tcp_max_syn_backlog = 8192
net.ipv4.tcp_syncookies = 1
net.ipv4.ip_local_port_range = 1024 65535
net.ipv4.tcp_tw_recycle = 0
net.ipv4.tcp_tw_reuse = 1

# Only if cubic is available
net.ipv4.tcp_congestion_control = cubic
```

You can apply the settings without rebooting the server with this command:

```
sudo sysctl -p
```

18.2. Linux User Limits

SavaPage server may run out of file descriptors as the system defaults are normally very low. A file descriptor (FD) is a handle created by a process when a file is opened. Each process can use a limited number of FDs as specified per user in an OS level user limit.

Beware that apart from “regular” files that are accessed by SavaPage from disk, each incoming request that uses a TCP socket also consumes one file descriptor from the total available for the process.

On Debian based systems the number of process FDs for the `savapage` user can be increased as follows.

Edit the file `/etc/security/limits.conf` like this:

```
sudo vi /etc/security/limits.conf
```

and add the following lines at the end of the file:

```
savapage    hard    nofile      65535
savapage    soft    nofile      65535
```

Next, open `/etc/pam.d/su` like this:

```
sudo vi /etc/pam.d/su
```

and uncomment the following line:

```
session    required    pam_limits.so
```

You also need to edit the `/etc/pam.d/common-session` and `/etc/pam.d/common-session-noninteractive` files. Open the files like this:

```
sudo vi /etc/pam.d/common-session
sudo vi /etc/pam.d/common-session-noninteractive
```

and for each file add the following line to the end:

```
session required pam_limits.so
```

Finally, check whether the settings are applied with this command:

```
sudo su - savapage -c "ulimit -n"
```

This should output the value 65535.

18.3. JVM Tuning

SavaPage runs in the Java Virtual Machine (JVM) using the class libraries and other supporting files provided in the JRE.

The SavaPage JVM settings work fine, and generally there is no customization needed. However, if needed the JVM can be tuned by adding extra JVM arguments in the file:

```
/opt/savapage/server/custom/app-server.conf
```

Edit this file as `savapage` user and enter the extra JVM arguments as value of the `CUSTOM_JVM_ARGS` key. The example below shows the JVM arguments as explained in the next sections.

```
# Note: enclose the value with quotes
CUSTOM_JVM_ARGS="-XX:DefaultMaxRAMFraction=2 -XX:+UseConcMarkSweepGC -XX:+CMSIncrementalMode"
```

The location of temporary files can be overwritten with the `JAVA_IO_TMPDIR` key.

```
# Overwrite of JVM system property 'java.io.tmpdir'
# User 'savapage' must have mode 700 access to this directory.
#JAVA_IO_TMPDIR=
```



Important

Before doing any JVM customizing please consult SavaPage Support to discuss your requirements and which customization fits best.

18.3.1. JVM Memory Allocation

The JVM allocates a quarter of host system RAM to the SavaPage Server process by default. This ensures that SavaPage does not consume too many resources and does not get in the way of other applications running on the same system.

However, if the host system is dedicated to running SavaPage, you can safely allocate more memory to SavaPage. With more allocated memory SavaPage will have a better performance, particularly with many users and large printing throughput.

Add one of the following JVM parameters to allocate relative or absolute memory:

```
-XX:DefaultMaxRAMFraction=3
-XX:DefaultMaxRAMFraction=2
-Xmx864m
```

❶
❷
❸

- ❶ Allocate one third of host system RAM.
- ❷ Allocate one half of host system RAM.
- ❸ Allocate 864MB of host system RAM.

18.3.2. JVM Garbage Collection

Customizing Java Garbage Collection (GC) depends on the characteristics of the application involved. The JVM provide proper defaults for SavaPage most of the time.

However, if you consider response time more important than overall throughput and garbage collection pauses must be kept shorter than approximately one second, then select the concurrent collector with the `-XX:+UseConcMarkSweepGC` option. Also, if only one or two processors are available, consider combining this collector with the `-XX:+CMSIncrementalMode` option.

Please consult the Java SE 6 HotSpot [Oracle documentation](#)⁴ for an introduction to these tuning options.

18.3.3. JVM Temporary Files

The Java system property `java.io.tmpdir` determines where the JVM writes temporary files. The default value typically points to the world readable `/tmp` directory. For security reasons SavaPage sets `java.io.tmpdir` to its own `/opt/savapage/server/tmp` directory, accessible by the `savapage` system user only. As a result, any Java component that is part of SavaPage will write its temporary files to that directory.

If you want to separate the location of temp files from the SavaPage application location, you can override the default in:

```
/opt/savapage/server/custom/app-server.conf
```

Edit this file as `savapage` user and specify the alternative temp directory at the `JAVA_IO_TMPDIR` variable.

```
# Overwrite of JVM system property 'java.io.tmpdir'
# User 'savapage' must have mode 700 access to this directory.
#JAVA_IO_TMPDIR=
```



Caution

Use the `JAVA_IO_TMPDIR` setting with the *utmost care*. Make sure the directory is *exclusively used* for SavaPage temporary files, and is accessible by the `savapage` system user only. If these conditions are not met, SavaPage will not start and might get corrupted.

SavaPage writes its own temp files to a subdirectory of `java.io.tmpdir`, called `savapage`. This subdirectory is created when the application starts and removed when stopped.

You can overwrite this location in:

```
/opt/savapage/server/server.properties
```

Edit this file as `savapage` user and specify an absolute path at the `app.dir.tmp` property. Beware that:

1. The custom location must reside on the same partition as the locations for letterheads and safepages, as overwrite (or defaults) values for the `app.dir.safepages` and `app.dir.letterheads` properties.

⁴ <https://www.oracle.com/technetwork/java/javase/gc-tuning-6-140523.html#cms>

2. Make sure the directory is *exclusively used* for SavaPage temporary files, and is accessible by the savapage system user only.



Caution

If these conditions are not met, SavaPage will not function, and crucial data of other applications might be permanently lost.

Chapter 19. SavaPage Community



Organizations join the SavaPage Community by subscribing to the software. The subscription consists of a one-time *enrollment* fee and a *yearly* amount. These fees are needed to financially compensate Community Developers and Deployment Partners for their efforts and expenses when maintaining the software and supporting member organizations. The amount is dependent on the size (number of Participants) of the organization. When you join you get a Member Card, which actually is a digitally signed file that is emailed to you. This file is your token as resident of the SavaPage Community and can be used to confirm your status in the SavaPage Software. Members have the right to request new features and are entitled to high-priority Technical Support.

An organization that uses the software and is *not* a Member is called a Visitor. Visitors are allowed to explore the application to decide if they want to subscribe to the software or not.

The community status is shown on the Admin Web App [Dashboard](#) and [About](#) sections.

19.1. Visitor Period

Without a Community Member Card, any user of the software is considered a Visitor. After 40 days visitors are invited to contact SavaPage Support for a Member Card. Without a card SavaPage will continue to run as normal and will be fully functional, but the missing card will be signaled as system status.



Note

Implementations with 5 active users (or less) in the SavaPage database are welcomed as permanent visitors, and the missing Member Card is not signalled as system status.

19.2. Registered Member

The Member Card supplied to you is the digital proof of your *Community Member* status and holds information about:

- The *Name* of the Member organization.
- The *Number of Participants* in the Member organization.
- The *Application version* of the actual release at the moment the Member Card was issued.
- The *Expiration date* (end date) of the subscription period.

After you import your Member Card file into SavaPage, your membership will be validated against your use of the application. A new Member Card is suggested when one of the following conditions are met:

- *The number of users in the SavaPage database exceed the number of participants* . This happens when extra external users were synchronized into the user database or extra internal users were added. You can extend the subscription and receive a new Member Card with an increased number of participants, or reduce the number of users in the database, by deleting internal users or deleting external users which are not present in the synchronization source, or by importing from a just a single synchronization source group.
- *The expiration date of the Member Card is reached*. The resolving action is to extend your subscription.



Important

Whatever your community status is, you'll always be able to use the software without restrictions. However, when deemed necessary, we will make an appeal to you to apply for the Member Card that covers your runtime situation.

19.3. Importing the Member Card

The SavaPage Community Member Card is issued as a digitally signed file. Installing the file into the application confirms your community status. To install the file supplied by your Community Partner:

1. Save the Member Card file to your hard disk. Your desktop is a handy location. Files are typically named Sava-Page-[orgname].membercard. The file can be loaded into the system as supplied.
2. Log into the SavaPage Admin Web App and navigate to the About page.
3. Scroll down to the **Community Membership** section and click the **Import Member Card** button.
4. Please see [Figure 4.122, “Admin Web App: About - Import Member Card” \[166\]](#) how to proceed in the import dialog.
5. Verify that your Membership is correctly listed in the About page.

If you have a question about your Member Card or need assistance please email SavaPage Technical Support and they will be more than happy to assist you.



Note

The file supplied is simply a digitally signed and zipped text file containing your Membership information. It's converted to ZIP format to minimize size. If you're interested in viewing the contents of the file, rename the file to .zip and simply open it in any ZIP extraction utility.

Appendix A. Proxy Print Scenarios

This chapter summarizes several Proxy Print scenarios in a shorthand catalogue. Please follow the hyperlinks in the summaries for more detail.

A.1. Personal Print Scenarios

A.1.1. Non-secure Print Scenario

See [Section 4.10.10, “Proxy Print”](#) [141].

A.1.2. Authenticated Print Scenarios

See [Section 4.9.1.2, “Proxy Print Authentication”](#) [114].

A.1.2.1. Direct Print Scenario

See [Section 4.9.1.2.3, “Direct Print Mode”](#) [116].

A.1.2.2. Hold Print Scenario

See [Section 4.9.1.2.2, “Hold Print Mode”](#) [116].

A.1.2.3. Fast Print Scenario

See [Section 4.9.1.2.1, “Fast Print Mode”](#) [115].

A.2. Delegated Print Scenarios

See [Section 3.4.3, “Delegated Print”](#) [42].

A.2.1. Delegated Print - (Non) Secure & Job Ticket Scenarios

Delegated Print	
User Story	<p>“As a Delegate, I can print for other users (delegators) so that costs are charged to individual delegators and delegator groups”</p> <p>“As a Job Ticket Operator, I can release or cancel the Job Ticket.”</p> <p>“As an Operator, I can cancel a pending Print Job in CUPS.”</p>
Preconditions	<ul style="list-style-type: none">Delegated Print is <i>enabled</i>.Delegated Print with PaperCut is <i>disabled</i>.Job is printed as <i>Job Ticket</i>, or in <i>Non-secure</i> or <i>Authenticated</i> (Hold or Direct) Print Mode.
Process	<ol style="list-style-type: none">Delegate creates Delegated Print Job.Delegate prints the job.
Scenario 1	<p><i>Non-secure Print</i></p> <p>Given the Delegate printed to a non-secure Proxy Printer, Then:</p>

Delegated Print	
	<ul style="list-style-type: none"> The job is send to the printer. Print-out document for Delegate is created with status Completed. <ul style="list-style-type: none"> SavaPage Account Transactions are created and linked. <ul style="list-style-type: none"> Transactions are for personal delegators and for the delegator groups involved. Printing costs are pro-rata divided over the Account Transactions. Transactions are displayed as Printer Usage with the pro-rata amount and status Completed. Transactions Summary with cost data is displayed with the Print-out document.
Scenario 2	<p><i>Authenticated Print Release.</i></p> <p>Given the Delegate printed to a Hold or Direct Mode Proxy Printer, And released the job by NFC Authentication, Then:</p> <ul style="list-style-type: none"> The Print-out document is created as in “Scenario 1”.
Scenario 3	<p><i>Job Ticket Released.</i></p> <p>Given the Delegate printed a Job Ticket, And the ticket is released by a Job Ticket Operator, Then:</p> <ul style="list-style-type: none"> The Print-out document is created as in “Scenario 1”, using <i>Job Ticket Cost</i> as printing cost.
Scenario 4	<p><i>Job Ticket Canceled.</i></p> <p>Given the Delegate printed a Job Ticket, And the ticket is canceled by the Delegate or an Operator (in SavaPage or CUPS), Then:</p> <ul style="list-style-type: none"> The Job Ticket is just removed.
References	<ul style="list-style-type: none"> Section 3.4.3, “Delegated Print” [42]. Section 4.9.1.2, “Proxy Print Authentication” [114]. Section 3.4.8, “Job Ticket Print” [51].

Table A.1. Delegated Print - (Non) Secure & Job Ticket Scenarios

A.2.2. Delegated Print - Job Ticket - PaperCut - Scenario

Delegated Print via Job Ticket to PaperCut	
User Story	<p>“As a Delegate, I can create a SavaPage Job Ticket for other users (delegators) so that costs are charged to individual delegators and delegator groups in PaperCut.”</p> <p>“As a Job Ticket Operator, I can release or cancel the Job Ticket.”</p> <p>“As an Operator, I can release or cancel a pending Print Job in PaperCut or CUPS.”</p>
Preconditions	<ul style="list-style-type: none"> PaperCut Integration is enabled. Delegated Print with PaperCut is enabled.
Process	<ol style="list-style-type: none"> Delegate creates Delegated Print Job. Delegate prints job as <i>Job Ticket</i>. Job Ticket Operator releases the ticket to a PaperCut managed printer. Print-out document for Delegate is created with status Pending (external) <ul style="list-style-type: none"> Shadow SavaPage Account Transactions are created and linked. <ul style="list-style-type: none"> Transactions are for personal delegators and for the delegator groups involved. Transactions act as template for transactions created in PaperCut after the print is completed. Transactions are displayed as Printer Usage with zero amount and status Pending (external). Transactions Summary without cost data is displayed with the Print-out document. PaperCut print status is monitored till end state is reached. <ul style="list-style-type: none"> Status is monitored for max 5 days , after which the Print-out document gets status Error, and the ticket can be closed.
Scenario 1	<i>Print Completed.</i>

Delegated Print via Job Ticket to PaperCut	
	<p>Given the printer is a PaperCut Hold/Release queue And the Operator Released the job (in PaperCut), Or the job was printed directly, Then:</p> <ul style="list-style-type: none"> Print-out document gets status Completed. SavaPage Job Ticket Cost is pro-rata divided over the Shadow SavaPage Account Transactions. <ul style="list-style-type: none"> Transactions are displayed as Printer Usage with the pro-rata amount and status Completed. Transactions Summary with cost data is now displayed with the Print-out document. PaperCut Account Transactions are created according to the Shadow Accounts template.
Scenario 2	<p>Print is Canceled or Timed out.</p> <p>Given the printer is a PaperCut Hold/Release queue And the Operator Canceled the job (in PaperCut or CUPS) or the job Expired (in PaperCut), Then:</p> <ul style="list-style-type: none"> Print-out document gets status Canceled. <ul style="list-style-type: none"> Transactions <i>continue</i> to be displayed as Printer Usage with zero amount but have status Canceled. Transactions Summary without cost data <i>continues</i> to be displayed at the Print-in document.
References	<ul style="list-style-type: none"> Section 3.4.3, “Delegated Print” [42]. Section 3.4.8, “Job Ticket Print” [51]. Section N.1, “Delegated Print to PaperCut” [315].

Table A.2. Delegated Print - Job Ticket - PaperCut Scenario

A.2.3. Delegated Print - PaperCut Scenario

Delegated Print to PaperCut	
User Story	<p>“As a Delegate, I can print for other users (delegators) so that costs are charged to individual delegators and delegator groups in PaperCut.”</p> <p>“As an Operator, I can release or cancel a pending Print Job in PaperCut or CUPS.”</p>
Preconditions	<ul style="list-style-type: none"> PaperCut Integration is enabled. Delegated Print with PaperCut is enabled. Proxy Printer used for printing is managed by PaperCut. Proxy Printer used for printing is <i>non-secure</i>. <ul style="list-style-type: none"> If Hold/Release printing is required it must be handled in PaperCut.
Process	<ol style="list-style-type: none"> Delegate creates Delegated Print Job. Delegate prints job to PaperCut Managed printer. Print-out document for Delegate is created with status Pending (external) <ul style="list-style-type: none"> Shadow SavaPage Account Transactions are created and linked. <ul style="list-style-type: none"> Transactions are for personal delegators and for the delegator groups involved. Transactions act as template for transactions created in PaperCut after the print is completed. Transactions are displayed as Printer Usage with zero amount and status Pending (external). Transactions Summary without cost data is displayed with the Print-out document. PaperCut print status is monitored till end state is reached. <ul style="list-style-type: none"> Status is monitored for max 5 days , after which the Print-out document gets status Error, and the ticket can be closed.
Scenario 1	<p>Print Completed.</p> <p>Given the printer is a PaperCut Hold/Release queue And the Operator Released the job (in PaperCut) , Or the job was printed directly, Then:</p> <ul style="list-style-type: none"> Print-out document gets status Completed. PaperCut Printing Cost is pro-rata divided over the Shadow SavaPage Account Transactions. <ul style="list-style-type: none"> Transactions are displayed as Printer Usage with the pro-rata amount and status Completed.

Delegated Print to PaperCut	
	<ul style="list-style-type: none"> • Transactions Summary <i>with</i> cost data is now displayed with the Print-out document. • PaperCut Account Transactions are created according to the Shadow Accounts template.
Scenario 2	<p><i>Print is Canceled or Timed out.</i></p> <p><i>Given</i> the printer is a PaperCut Hold/Release queue <i>And</i> the Operator Canceled the job (in PaperCut or CUPS) or the job Expired (in PaperCut) , <i>Then</i>:</p> <ul style="list-style-type: none"> • Print-out document gets status Canceled. • Transactions <i>continue</i> to be displayed as Printer Usage with zero amount but have status Canceled. • Transactions Summary without cost data <i>continues</i> to be displayed at the Print-in document.
References	<ul style="list-style-type: none"> • Section 3.4.3, “Delegated Print” [42]. • Section N.1, “Delegated Print to PaperCut” [315].

Table A.3. Delegated Print - PaperCut Scenario

Appendix B. NFC Authentication



SavaPage supports Radio Frequency Identification (RFID) as authentication method.

RFID is the technology for uniquely identifying items using radio waves. A basic RFID system comprises a *passive tag*¹, a *reader*, and an *antenna*, where the reader sends an interrogating signal to the tag via the antenna, and the tag responds with its Unique Identification (UID).

In this way RFID tags are commonly used as authentication token: the RFID reader connected to the authenticator just passes the UID (Card Number) of the tag. Applications are abundant, ranging from tags embedded into retail products to help stores keep tabs on inventory, to tags embedded into animals to keep track of life stock. RFID is also applied in passports and credit cards, as well as identification badges that let employees access secure areas.

Near Field Communication (NFC) is a more recent, finely honed version of RFID with a much broader application. While RFID is a one-way communication system only, with data flowing from tag to reader, NFC can also be set up for two-way communication. However, NFC operates at a maximum range of about 4 inches (10 centimeters) and uses High Frequency (HF) RFID readers at 13.56 MHz.

Since SavaPage is targeted at the same HF RFID readers and tags, albeit in one-way communication, this manual uses the more common terms *NFC Card* and *NFC Reader* for the tag and reader role. In some contexts the terms *Card* and *Card Reader* will be used as shorthand.

SavaPage supports two Card Reader types.

- A *Local* Card Reader: a keyboard emulating device that “types” the UID (Card Number) each time a Card is swiped.
- A *Network* Card Reader: a software component, implemented on a dedicated device (like a Raspberry Pi®), that interacts with an NFC Reader after a card swipe and sends the UID to the central SavaPage server.

B.1. Card Number Format

SavaPage stores the Card Number (UID) in lower case HEX format, with Least Significant Byte (LSB) first. So, at the interfaces where the UID is captured, the output format and byte order must be specified as HEX or DECIMAL and LSB or MSB (Least or Most Significant Byte) first. This information is used by SavaPage to convert the captured Card Number to its internal HEX/LSB standard.

B.2. Local Card Reader

A *Local Card Reader* is an NFC Reader that functions as USB Keyboard Emulator. At each card swipe the reader must react by “typing” the card's UID (Card Number) appended by a Carriage Return (CR). SavaPage makes use of this function by capturing² the keystrokes at Login time in the Web App.

¹ RFID tags are either Active or Passive. Active tags have their own power supply by which they can broadcast with a read range of up to 100 meters. Passive tags do not have their own power source. Instead, they are powered by the electromagnetic energy transmitted from the RFID reader. Because the radio waves must be strong enough to power the tags, passive RFID tags have a read range from near contact to a few meters.

² SavaPage uses a short time limit to capture the keystrokes from a Local Card Reader. The time limit (milliseconds) is contained in the configuration key `webapp.card-local.keystrokes-max-msecs`. Do not change this value, except when requested by the SavaPage support desk.



Note

The way a reader formats the UID can deviate from the SavaPage HEX/LSB standard. Therefore you need to specify the format at the interfaces where the reader's UID is used. Most keyboard emulating readers can be configured to a specific output format and byte order.



Tip

At the time of this writing [StrongLink](http://www.stronglink-rfid.com)³ sells a reliable Plug and Play USB Keyboard Emulating Card Reader (SL040A) for a competitive price. The reader supports UID reads for Mifare Mini, Mifare 1k, Mifare 4k, Mifare Plus, Ultralight, DesFire and Mifare_ProX cards.

B.3. Network Card Reader Service

SavaPage ships with a Network Card Reader Service to be deployed on a Raspberry Pi®. The service is tested to work with the popular ACS ACR122U⁴ reader. The installation instructions can be found in:

```
/opt/savapage/providers/nfc/linux-armv6/README
```



Note

Other ACS USB readers mentioned in the README should work as well.

Any deployed service must be entered as SavaPage Device. See [Section 4.9.1, “Network Card Reader” \[113\]](#). At each card swipe the UID of the card is read and send to the central SavaPage server, where it is handled in context of the device definition.

You can link sounds and scrips to various events. Sample files are provided for your own customization, for example to communicate with [PiGlow](http://www.piglow.com/)⁵, [Pibrella](http://pibrella.com/)⁶ or [PiFace Control & Display](http://www.piface.org.uk/)⁷ add-on boards.

³ <http://www.stronglink-rfid.com>

⁴ <http://www.acs.com.hk/en/products/3/acr122u-usb-nfc-reader/>

⁵ <http://www.pimoroni.com/>

⁶ <http://pibrella.com/>

⁷ <http://www.piface.org.uk/>

Appendix C. Tools



C.1. Server Commands

The **savapage-cmd** tool provides a command-line interface to SavaPage Server methods. It can directly be executed on the command-line or be part of more elaborate shell scripts.

For security reasons only users with read access to the `/opt/savapage/server/server.properties` file have the right to execute the command. So, the sure way to go is ...

```
sudo su - savapage
cd server/bin/linux-x64
```

... and to execute `savapage-cmd` from here, and ...

```
./savapage-cmd --help
```

... will echo all methods available:

SavaPage Command Line Interface

Note: use `METHOD --help` for method details.

usage: `[METHOD] [OPTION]...`

<code>--add-internal-user</code>	Creates a new or Updates an existing Internal User.
<code>--add-user-group</code>	Adds a user group from the external user source: synchronized external users belonging to this group are added as member.
<code>--change-base-currency</code>	Changes the base currency of the application.
<code>--delete-user</code>	Logically deletes a User.
<code>--delete-user-group</code>	Deletes a user group.
<code>--erase-user</code>	Erases a User, complying to GDPR Data Erasure (Right to be Forgotten).
<code>--get-config-property</code>	Gets configuration property value.
<code>--list-users</code>	Lists the names of all the Users in the system, sorted by user name, one per line.
<code>--list-user-groups</code>	Lists the names of all the User Groups in the system, sorted by name, one per line.
<code>--list-user-group-members</code>	Lists the names of the user group members in the system, sorted by user name, one per line.
<code>--list-user-group-memberships</code>	Lists the names of the groups a user belongs to, sorted by name, one per line.
<code>--list-user-source-groups</code>	Lists the names of all the groups in the user source, sorted by name, one per line.
<code>--list-user-source-group-members</code>	Lists the names of the (nested) user group members in the user source, sorted by user name, one per line.
<code>--list-user-source-group-nesting</code>	Lists a space indented hierarchy of nested groups within a group. Nested groups are only supported by Active Directory, all other user sources return an empty list.
<code>--printer-access-control</code>	Controls user groups to either allow or deny

<code>--printer-snmp</code>	access to a proxy printer.
<code>--set-config-property</code>	Reads SNMP info from a printer.
<code>--set-user-properties</code>	Sets configuration property value.
	Sets properties for an existing Internal or External User.
<code>--set-user-group-properties</code>	Sets properties of an Internal or External User Group.
<code>--sync-user-group</code>	Synchronizes a user group with the external user source, updating group membership.
<code>--sync-users-and-groups</code>	Starts user and group synchronization with external user source.
<code>--system-status</code>	Gets the system status enum value: READY, SETUP, UNAVAILABLE, MAINTENANCE.
<code>-help, --help</code>	Displays this help text.
<code>--help-all</code>	Displays help text of all methods.



Note

The number of available methods will grow according to customer needs. Please contact support if you need a method that is missing.

C.1.1. Common Options

C.1.1.1. Keep Switches

`--keep-*` option switches are used to *not* overwrite existing values.

For example, the `--keep-card`, `--keep-pin` and `--keep-password` switches make their corresponding `--card`, `--pin` and `--password` options act as defaults in those cases where values have not yet been set.

Some examples:

```
# Overwrite any PIN set by user.
--add-internal-user --username "guest-john" --pin "1234"

# Preserve any PIN set by user.
--add-internal-user --username "guest-john" --pin "1234" --keep-pin
```

C.1.1.2. Remove Switches

`--remove-*` option switches are used to *clear* values. Since the *absence* of a command-line option (or an empty value in *batch mode* CSV/TSV files) can *not* be interpreted as *no value* (null), the `--remove` switch comes to help to explicitly nullify values.

This implies that blank values on the command-line and in *batch mode* input files are ignored. So, this command has no effect ...

```
--add-internal-user --username "guest-john" --pin ""
```

... use this command instead ...

```
--set-user-properties --username "guest-john" --remove-pin
```

When an option does *not* have a `--remove-*` switch, there is no way to clear the corresponding field. For example, since `--remove-full-name` is not available, there is no way to clear the User field “full-name” from the command-line (see [Section C.1.19, “setUserProperties” \[263\]](#)).

C.1.1.3. Locale Option

Some methods pass numeric values that are formatted according to the locale. In these cases the locale can be specified with a separate option like this:

```
-locale <arg>           The IETF BCP 47 Locale used for numeric values.
                        Example values are: en, en-GB, en-US, nl,
                        nl-NL, nl-BE. [defaults to system default
                        en-US].
```



Note

The actual system default locale depends on your terminal session settings.

C.1.1.4. Batch Mode Options

Some methods have options for passing values in batch mode. Below are the standard batch mode parameters:

```
-batch                Enables batch mode: executing from CSV or TSV
                        input.
-continue             Continues batch processing after a batch line
                        execution error.
-input <arg>          Batch input file: optional with stdin as
                        default.
-charset <arg>        IANA Charset Name of batch input character
                        encoding [default: utf-8].
```

So instead of using these three commands ...

```
./savapage-cmd --add-internal-user --username john --password rTf4g
./savapage-cmd --add-internal-user --username dave --password 9j6Tw
./savapage-cmd --add-internal-user --username mick --password f75L2
```

... you can use this single batch command ...

```
./savapage-cmd --add-internal-user -batch -input /home/rijk/add-internal-user.csv
```

.. where the file `add-internal-user.csv` looks like this:

```
"username", "password"
"john", "rTf4g"
"dave", "9j6Tw"
"mick", "f75L2"
```

Input files must have the extension `.csv` or `.tsv` as indication for a comma or tab separated file format.

The first line in the file must be the comma or tab separated list of parameters. The convention is that the parameter names are identical to their command line counterpart, except for the `--` prefix. The next lines simply contain the comma or tab separated parameter values.

Option *switches* like applied in the command below ...

```
--set-user-properties --username "john" --pin 1234 --remove-card --full-name "John Brown"
--set-user-properties --username "carol" --pin 4713 --keep-pin --full-name "Carol Johnson"
```

... can be applied in a CSV file like this:

```
"username", "pin", "keep-pin", "remove-card", "full-name"
"john", 1234, , "true", "John Brown"
```



```
"carol",4713,"true",,"Carol Johnson"
```



Important

By default, batch processing is interrupted after a batch line execution error. With the `-continue` switch set, it will instead continue processing. After the batch finishes it will return error code 5 to distinguish continuation from an immediate termination, which is reported with error return code 1.



Note

In a CSV/TSV file any *blank* switch value is interpreted as not present (*false*), any non-blank value as present (*true*).

C.1.2. addInternalUser

```
./savapage-cmd --add-internal-user --help
```

... gives the options:

SavaPage Command Line Interface

```
Method : addInternalUser
Version : 0.30
```

Creates a new or updates an existing Internal User.

```
usage: --add-internal-user [OPTION]...
--username <text(50)>           [required] Unique user name.
--password <text(64)>           [optional] Password.
--full-name <text(255)>         [optional] Full user name.
--email <text(255)>             [optional] Primary Email address.
--email-other <list>           [optional] List of space separated other
                                (secondary) Email addresses.
--card <text(16)>               [optional] NFC Card Number.
--card-format <HEX|DEC>        [optional] NFC Card Number Format [default:
                                HEX].
--card-first-byte <LSB|MSB>    [optional] NFC Card Number First Byte [default:
                                LSB].
--id <text(16)>                 [optional] ID Number.
--pin <text(16)>                [optional] PIN for ID and Card.
--yubikey <text(12)>            [optional] YubiKey Public ID.
--uuid <text(36)>               [optional] The user's secret UUID.
--balance <decimal>            [optional] The user's initial account balance.
                                This value is ignored when a balance is already
                                assigned.
--balance-comment <text(255)>   [optional] A comment to be associated with the
                                --balance transaction.
--credit-limit                 [optional] Assign default credit limit amount.
--credit-limit-amount <decimal> [optional] Assign custom credit limit amount.
--credit-limit-none            [optional] no credit limit restriction (opposed
                                to --credit-limit and --credit-limit-amount).
--keep-card                    [optional] Keep existing Card Number, or use
                                --card value when not present.
--keep-email-other             [optional] Keep existing other (secondary)
                                Email addresses, or use --email-other value
                                when not present.
--keep-password                [optional] Keep existing Password, or use
                                --password value when not present.
--keep-pin                     [optional] Keep existing PIN, or use --pin
                                value when not present.
```

<code>--keep-uuid</code>	[optional] Keep existing UUID, or use <code>--uuid</code> value when not present.
<code>-h,--help</code>	Displays this help text.
<code>-batch</code>	Enables batch mode: executing from CSV or TSV input.
<code>-continue</code>	Continues batch processing after a batch line execution error.
<code>-input <arg></code>	Batch input file: optional with stdin as default.
<code>-charset <arg></code>	IANA Charset Name of batch input character encoding [default: utf-8].
<code>-locale <arg></code>	The IETF BCP 47 Locale used for numeric values. Example values are: en, en-GB, en-US, nl, nl-NL, nl-BE. [defaults to system default en-US].

C.1.3. addUserGroup

```
./savapage-cmd --add-user-group --help
```

... gives the options:

SavaPage Command Line Interface

Method : addUserGroup
Version : 0.10

Adds a user group from the external user source: synchronized external users belonging to this group are added as member.

usage: --add-user-group [OPTION]...

<code>--groupname <text(255)></code>	[required] Unique group name.
<code>-h,--help</code>	Displays this help text.

C.1.4. changeBaseCurrency

```
./savapage-cmd --change-base-currency --help
```

... gives the options:

SavaPage Command Line Interface

Method : changeBaseCurrency
Version : 0.10

Changes the base currency of the application.

This action creates financial transactions to align each account to the new currency: the current account balance is nullified by a debit transaction and replaced with the new currency according to the exchange rate via a credit transaction.

Individual credit limits are converted as well, default credit limits are not.

WARNING: Create a database back-up before executing this command!

usage: --change-base-currency [OPTION]...

<code>--from <text(3)></code>	[required] The current currency code (ISO 4217).
<code>--to <text(3)></code>	[required] The new currency code (ISO 4217).
<code>--exchange-rate <decimal></code>	[required] The exchange rate.
<code>--test</code>	[optional] Dry run, changes are not committed.

```
-h,--help           Displays this help text.
```

C.1.5. deleteUser

```
./savapage-cmd --delete-user --help
```

... gives the options:

```
SavaPage Command Line Interface

Method  : deleteUser
Version : 0.10

Logically deletes a User.

usage: --delete-user [OPTION]...

    --username <text(50)>    [required] Unique user name.
-h,--help                  Displays this help text.
-batch                     Enables batch mode: executing from CSV or TSV input.
-continue                  Continues batch processing after a batch line
                           execution error.
-input <arg>               Batch input file: optional with stdin as default.
-charset <arg>              IANA Charset Name of batch input character encoding
                           [default: utf-8].
```

C.1.6. deleteUserGroup

```
./savapage-cmd --delete-user-group --help
```

... gives the options:

```
SavaPage Command Line Interface

Method  : deleteUserGroup
Version : 0.10

Deletes a user group.

usage: --delete-user-group [OPTION]...

    --groupname <text(255)>  [required] Unique group name.
-h,--help                  Displays this help text.
```

C.1.7. eraseUser

```
./savapage-cmd --erase-user --help
```

... gives the options:

```
SavaPage Command Line Interface

Method  : eraseUser
Version : 0.10

Erases a User, complying to GDPR Data Erasure (Right to be Forgotten).

usage: --erase-user [OPTION]...
    --username <text(50)>    [required] Unique user name.
-h,--help                  Displays this help text.
```

-batch	Enables batch mode: executing from CSV or TSV input.
-continue	Continues batch processing after a batch line execution error.
-input <arg>	Batch input file: optional with stdin as default.
-charset <arg>	IANA Charset Name of batch input character encoding [default: utf-8].

This command clears the user's personal data in the **active database**, including all identifying content from the transaction log, document log and personal account. Transaction and document log details are still held in the database for reporting purposes, but without any user related data. The user is also logically deleted, when not already done so: see [Section C.1.5, “deleteUser” \[258\]](#) and [Section 4.10.2, “User Creation” \[124\]](#).



Caution

When a database is **restored**, erased users might be restored as identifiable users again. Therefore, keep **Data Erasure** requests in a separate administration, so that they can easily be retrieved and re-executed.



Warning

Erasing users without deleting them from their **external user source** has a temporary effect, since they will be created again at the next **synchronization**.



Note

Data Erasure does *not* include database **backups** and **log files**.

C.1.8. getConfigProperty

```
./savapage-cmd --get-config-property --help
```

... gives the options:

SavaPage Command Line Interface

Method : getConfigProperty
Version : 0.10

Gets configuration property value.

This is an advanced command. Please contact SavaPage Support before use.

usage: --get-config-property [OPTION]...
 --name <text(100)> [required] Property name.
 -h, --help Displays this help text.



Note

Values that are stored encrypted are returned decrypted. See [Section 13.6, “Encrypted Secrets” \[218\]](#).

Also see [Section C.3.1, “db-config-get” \[270\]](#) and [Section C.1.18, “setConfigProperty” \[263\]](#).

C.1.9. listUsers

```
./savapage-cmd --list-users --help
```

... gives the options:

```
SavaPage Command Line Interface

Method  : listUsers
Version : 0.10

Lists the names of all the Users in the system, sorted by user name, one per line.

usage: --list-users [OPTION]...

-h,--help    Displays this help text.
```

C.1.10. listUserGroups

```
./savapage-cmd --list-user-groups --help
```

... gives the options:

```
SavaPage Command Line Interface

Method  : listUserGroups
Version : 0.10

Lists the names of all the User Groups in the system, sorted by name, one per line.

usage: --list-user-groups [OPTION]...

-h,--help    Displays this help text.
```

C.1.11. listUserGroupMembers

```
./savapage-cmd --list-user-group-members --help
```

... gives the options:

```
SavaPage Command Line Interface

Method  : listUserGroupMembers
Version : 0.10

Lists the names of the user group members in the system, sorted by user name,
one per line.

usage: --list-user-group-members [OPTION]...

    --groupname <text(255)>    [required] Unique group name.
-h,--help                    Displays this help text.
```

C.1.12. listUserGroupMemberships

```
./savapage-cmd --list-user-group-memberships --help
```

... gives the options:

```
SavaPage Command Line Interface
```

```
Method : listUserGroupMemberships
Version : 0.10
```

Lists the names of the groups a user belongs to, sorted by name, one per line.

usage: --list-user-group-memberships [OPTION]...

```
    --username <text(50)>    [required] Unique user name.
-h,--help                  Displays this help text.
```

C.1.13. listUserSourceGroups

```
./savapage-cmd --list-user-source-groups --help
```

... gives the options:

```
SavaPage Command Line Interface
```

```
Method : listUserSourceGroups
Version : 0.10
```

Lists the names of all the groups in the user source, sorted by name, one per line.

usage: --list-user-source-groups [OPTION]...

```
-h,--help    Displays this help text.
```

C.1.14. listUserSourceGroupMembers

```
./savapage-cmd --list-user-source-group-members --help
```

... gives the options:

```
SavaPage Command Line Interface
```

```
Method : listUserSourceGroupMembers
Version : 0.10
```

Lists the names of the (nested) user group members in the user source, sorted by user name, one per line.

usage: --list-user-source-group-members [OPTION]...

```
    --groupname <text(255)>    [required] Unique group name.
    --nested                  [optional] Accumulate members from nested groups
                              (Active Directory only).
-h,--help                    Displays this help text.
```

C.1.15. listUserSourceGroupNesting

```
./savapage-cmd --list-user-source-group-nesting --help
```

... gives the options:

```
SavaPage Command Line Interface
```

```
Method : listUserSourceGroupNesting
```

Version : 0.10

Lists a space indented hierarchy of nested groups within a group. Nested groups are only supported by Active Directory, all other user sources return an empty list.

usage: --list-user-source-group-nesting [OPTION]...

 --groupname <text(255)> [required] Unique group name.
 -h,--help Displays this help text.

C.1.16. printerAccessControl

./savapage-cmd --printer-access-control --help

... gives the options:

SavaPage Command Line Interface

Method : printerAccessControl
 Version : 0.10

Controls user groups to either allow or deny access to a proxy printer.

usage: --printer-access-control [OPTION]...

 --printername <text(255)> [required] CUPS name of the proxy printer.
 --allow [optional] Allow access to --groupname (existing denied user groups are removed).
 --deny [optional] Deny access to --groupname (existing allowed user groups are removed).
 --remove [optional] Remove --groupname from the access list.
 --groupname <text(255)> [optional] Name of the user group to --allow, --deny or --remove access
 --remove-all [optional] Remove all user groups from the access list.
 --list [optional] Echoes the access list to stdout in CSV format.
 -h,--help Displays this help text.

C.1.17. printerSnmp

./savapage-cmd --printer-snmp --help

... gives the options:

SavaPage Command Line Interface

Method : printerSnmp
 Version : 0.20

Reads SNMP info from a printer.

usage: --printer-snmp [OPTION]...

 --printername <text(255)> [optional] CUPS printer name used to resolve host name (required when --host is not set).
 --host <text> [optional] Host name or IP address of the printer (required when --printername is not set).
 --port <number> [optional] SNMP port number (default 161).
 --community <text> [optional] SNMP community (default "public").
 --version <1|2c> [optional] SNMP version (default "1").
 -h,--help Displays this help text.

C.1.18. setConfigProperty

```
./savapage-cmd --set-config-property --help
```

... gives the options:

SavaPage Command Line Interface

Method : setConfigProperty
Version : 0.10

Sets configuration property value.

The property must be present in the database. Not all properties are available for update. This is an advanced command. Please contact SavaPage Support before use.

```
usage: --set-config-property [OPTION]...
  --name <text(100)>      [required] Property name.
  --value <text(1000)>    [required] Property value.
  -h,--help              Displays this help text.
  -batch                 Enables batch mode: executing from CSV or TSV input.
  -continue               Continues batch processing after a batch line execution
                          error.
  -input <arg>            Batch input file: optional with stdin as default.
  -charset <arg>          IANA Charset Name of batch input character encoding
                          [default: utf-8].
```



Note

Some values will be stored encrypted. See [Section 13.6, “Encrypted Secrets” \[218\]](#).

Also see [Section C.3.2, “db-config-set” \[271\]](#) and [Section C.1.8, “getConfigProperty” \[259\]](#).

C.1.19. setUserProperties

```
./savapage-cmd --set-user-properties --help
```

... gives the options:

SavaPage Command Line Interface

Method : setUserProperties
Version : 0.30

Sets properties for an existing Internal or External User.

```
usage: --set-user-properties [OPTION]...
  --username <text(50)>      [required] Unique user name.
  --password <text(64)>      [optional] Password (Internal User only).
  --full-name <text(255)>    [optional] Full user name.
  --email <text(255)>        [optional] Primary Email address.
  --email-other <list>      [optional] List of space separated other
                          (secondary) Email addresses.
  --card <text(16)>          [optional] NFC Card Number.
  --card-format <HEX|DEC>   [optional] NFC Card Number Format [default:
                          HEX].
  --card-first-byte <LSB|MSB> [optional] NFC Card Number First Byte [default:
                          LSB].
  --id <text(16)>            [optional] ID Number.
  --pin <text(16)>           [optional] PIN for ID and Card.
```


<code>--yubikey <text(12)></code>	[optional] YubiKey Public ID.
<code>--uuid <text(36)></code>	[optional] The user's secret UUID.
<code>--balance <decimal></code>	[optional] The user's current account balance.
<code>--balance-comment <text(255)></code>	[optional] A comment to be associated with the --balance transaction.
<code>--credit-limit</code>	[optional] Assign default credit limit amount.
<code>--credit-limit-amount <decimal></code>	[optional] Assign custom credit limit amount.
<code>--credit-limit-none</code>	[optional] No credit limit restriction (opposed to --credit-limit and --credit-limit-amount).
<code>--keep-card</code>	[optional] Keep existing Card Number, or use --card value when not present.
<code>--keep-email-other</code>	[optional] Keep existing other (secondary) Email addresses, or use --email-other value when not present.
<code>--keep-password</code>	[optional] Keep existing Password, or use --password value when not present (Internal User only).
<code>--keep-pin</code>	[optional] Keep existing PIN, or use --pin value when not present.
<code>--keep-uuid</code>	[optional] Keep existing UUID, or use --uuid value when not present.
<code>--remove-email</code>	[optional] Remove Primary Email address (opposed to --email).
<code>--remove-email-other</code>	[optional] Remove other (secondary) Email addresses (opposed to --email-other).
<code>--remove-card</code>	[optional] Remove NFC Card Number (opposed to --card).
<code>--remove-id</code>	[optional] Remove ID Number (opposed to --id).
<code>--remove-password</code>	[optional] Remove Password (Internal User only).
<code>--remove-pin</code>	[optional] Remove PIN (opposed to --pin).
<code>--remove-yubikey</code>	[optional] Remove YubiKey Public ID (opposed to --yubikey).
<code>--remove-uuid</code>	[optional] Remove UUID (opposed to --uuid).
<code>-h,--help</code>	Displays this help text.
<code>-batch</code>	Enables batch mode: executing from CSV or TSV input.
<code>-continue</code>	Continues batch processing after a batch line execution error.
<code>-input <arg></code>	Batch input file: optional with stdin as default.
<code>-charset <arg></code>	IANA Charset Name of batch input character encoding [default: utf-8].
<code>-locale <arg></code>	The IETF BCP 47 Locale used for numeric values. Example values are: en, en-GB, en-US, nl, nl-NL, nl-BE. [defaults to system default en-US].

C.1.20. setUserGroupProperties

```
./savapage-cmd --set-user-group-properties --help
```

... gives the options:

SavaPage Command Line Interface

Method : setUserGroupProperties
Version : 0.11

Sets properties of an Internal or External User Group.

usage: --set-user-group-properties [OPTION]...

<code>--groupname <text(255)></code>	[required] Unique group name.
<code>--balance <decimal></code>	[optional] The user's initial account

```

--credit-limit                balance.
                              [optional] Assign default credit limit
--credit-limit-amount <decimal> amount to new users.
                              [optional] Assign custom credit limit amount
--credit-limit-none          to new users.
                              [optional] Assign no credit limit
                              restriction to new users (opposed to
--role-job-ticket-creator <Y|N|U> --credit-limit and --credit-limit-amount).
--role-job-ticket-operator <Y|N|U> [optional] Assign Job Ticket Creator role.
--role-print-creator <Y|N|U>       [optional] Assign Job Ticket Operator role.
--role-print-delegate <Y|N|U>      [optional] Assign Print Creator role.
--role-print-delegator <Y|N|U>     [optional] Assign Print Delegate role.
--role-web-cashier <Y|N|U>         [optional] Assign Print Delegator role.
-h,--help                     [optional] Assign Web Cashier role.
                                Displays this help text.
-batch                         Enables batch mode: executing from CSV or
                                TSV input.
-continue                     Continues batch processing after a batch
                                line execution error.
-input <arg>                  Batch input file: optional with stdin as
                                default.
-charset <arg>                IANA Charset Name of batch input character
                                encoding [default: utf-8].
-locale <arg>                 The IETF BCP 47 Locale used for numeric
                                values. Example values are: en, en-GB,
                                en-US, nl, nl-NL, nl-BE. [defaults to system
                                default en-US].

```

C.1.21. syncUserGroup

```
./savapage-cmd --sync-user-group --help
```

... gives the options:

SavaPage Command Line Interface

Method : syncUserGroup
Version : 0.10

Synchronizes a user group with the external user source, updating group membership.

usage: --sync-user-group [OPTION]...

```

--groupname <text(255)>    [required] The name of the group to synchronize.
-h,--help                  Displays this help text.

```

C.1.22. syncUsersAndGroups

```
./savapage-cmd --sync-users-and-groups --help
```

... gives the options:

SavaPage Command Line Interface

Method : syncUsersAndGroups
Version : 0.10

Starts user and group synchronization with external user source.

This is equivalent to clicking "Synchronize now" in the Admin Web App.
Synchronization completes in the background.

```
usage: --sync-users-and-groups [OPTION]...
      --delete-users [optional] Remove users that do not exist in external user
      source.
      -h,--help          Displays this help text.
```

See [Section 4.10.2, “User Creation” \[124\]](#).

C.1.23. systemStatus

```
./savapage-cmd ---system-status --help
```

... gives the options:

```
SavaPage Command Line Interface

Method   : systemStatus
Version  : 0.10

Gets the system status enum value: READY, SETUP, UNAVAILABLE, MAINTENANCE.

usage: --system-status [OPTION]...
      -h,--help    Displays this help text.
```

This command is identical to [Section C.2.2.1, “systemStatus” \[268\]](#).

C.2. Web Services

SavaPage uses both [XML-RPC¹](#) and [JSON-RPC²](#) for its public Web Services. Both protocols are Open Standard, lightweight and have support for all major programming and scripting languages.



Note

Web Services will grow upon request. Please tell us if you need extra methods.

C.2.1. XML-RPC

The secure endpoint is: `https://savapage:8632/xmlrpc/v1`

C.2.1.1. onetime-auth.createToken

With this method a Trusted Third Party (TTP) acquires a one-time token for Web App user authentication.

The requesting User Name and returned Token must be offered to the Web App by hidden HTTP POST input names `auth_user` and `auth_token`. The Web App decrypts the token and honours the authentication request if the offered and encrypted Users match and the token is not expired.

	Name	Type	Description
Parameters	apikey	string	The TTP API key as set in the <code>web-login.tcp.apikey</code> configuration property.
	username	string	The User Name to authenticate. See Section 3.1, “Login” [22] . User Alias mapping is applied.
Returns	token	string	The one-time authentication token holding the encrypted User Name and token creation time.

¹ <https://en.wikipedia.org/wiki/XML-RPC>

² <https://en.wikipedia.org/wiki/JSON-RPC>

	Name	Type	Description
			The expiration criterion in milliseconds is set in the <code>web-login.token.expiry-msecs</code> configuration property.

Table C.1. XML-RPC method: `onetime-auth.createToken`

Note

TTP Web App Login authentication must be enabled by setting the `web-login.ttp.enable` configuration property to `Y`.

C.2.2. JSON-RPC

The secure endpoint is: `https://savapage:8632/jsonrpc/v1`

The following configuration properties apply:

Configuration property	Description
<code>api.jsonrpc.secret-key</code>	The secret key to be passed as <code>X-Auth-Key</code> HTTP header.
<code>api.jsonrpc.ext.ip-addresses-allowed</code>	A CIDR Set of allowed client IPv4 addresses. When void access is denied for <i>all</i> clients.

Table C.2. JSON-RPC Configuration Properties

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to set these items.

The sections below discuss the individual methods. All methods default to the same response layout in case of an error.

Basic Error
<pre> jsonrpc constant string 2.0 JSON-RPC version. id string Reference ID. result constant boolean false Error indicator. error object </pre> <div> <div>Error</div> <pre> code enum integer -32600 -32601 -32602 -32603 -32700 -32600 (invalid request), -32601 (method not found), -32602 (Invalid params), -32603 (internal error), -32700 (parse error). message string Error message. data object </pre> <div> <div>Data</div> <pre> @type constant string BASIC Data type indicator. reason string Error reason. </pre> </div> </div>

Figure C.1. JSON-RPC : Basic Error

C.2.2.1. systemStatus

This method returns the system status as enum value. `READY` indicates the system is ready to use and users have access to the Web App. All other values indicate that user access is *denied*, because: system `SETUP` is needed, `MAINTENANCE` is in progress, or the system is temporarily `UNAVAILABLE` due to running batch jobs (database backup, user synchronization, database cleanup).



Note

Although the system can have multiple "deny status" at the same time, the single most important status is returned, in the order `SETUP`, `MAINTENANCE`, `UNAVAILABLE`.

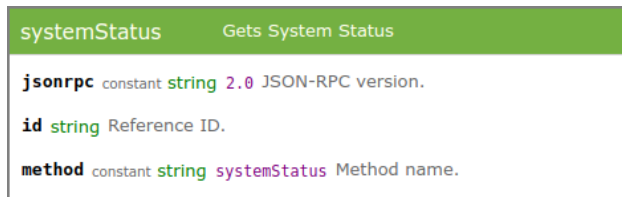


Figure C.2. JSON-RPC : systemStatus (request)

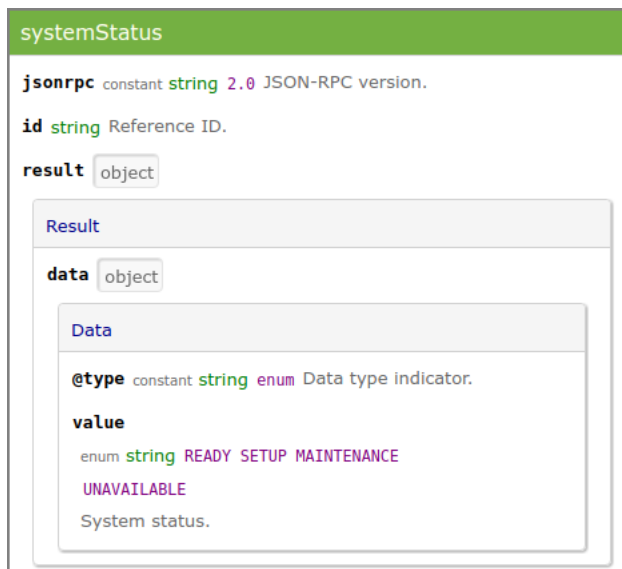


Figure C.3. JSON-RPC : systemStatus (response)

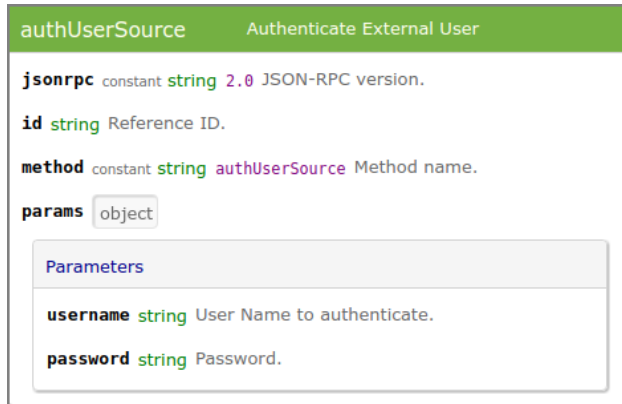
```
# sample request
curl --request POST --header "X-Auth-Key: 7961b5ec-bee4-11e7-8731-406186940c49" \
  --data '{"jsonrpc":"2.0","id":"reference","method":"systemStatus"}' \
  https://savapage:8632/jsonrpc/v1

# sample response
{
  "jsonrpc" : "2.0",
  "id" : "reference",
  "result" : {
    "data" : {
      "@type" : "enum",
      "value" : "READY"
    }
  }
}
```

```
}
}
```

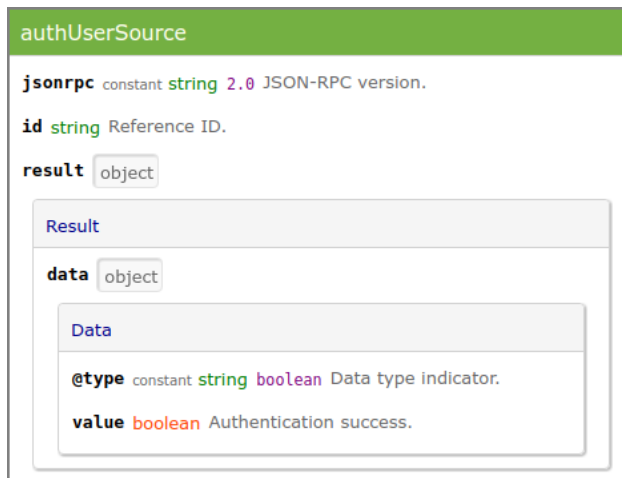
C.2.2.2. authUserSource

This method authenticates an external user against the configured [User Source](#).



The image shows a JSON-RPC request interface for the `authUserSource` method. The title bar is green and says "authUserSource Authenticate External User". The interface includes fields for `jsonrpc` (constant string 2.0), `id` (string Reference ID), `method` (constant string authUserSource), and `params` (object). A "Parameters" section is expanded, showing `username` (string User Name to authenticate) and `password` (string Password).

Figure C.4. JSON-RPC : authUserSource (request)



The image shows a JSON-RPC response interface for the `authUserSource` method. The title bar is green and says "authUserSource". The interface includes fields for `jsonrpc` (constant string 2.0), `id` (string Reference ID), and `result` (object). A "Result" section is expanded, showing a `data` object. The `Data` section is expanded, showing `@type` (constant string boolean) and `value` (boolean Authentication success).

Figure C.5. JSON-RPC : authUserSource (response)

```
# sample request
curl --request POST --header "X-Auth-Key: 7961b5ec-bee4-11e7-8731-406186940c49" \
  --data '{"jsonrpc":"2.0","id":"reference","method":"authUserSource", \
    "params":{"username":"john","password":"AzFi7I"}}'
  https://savapage:8632/jsonrpc/v1

# sample response
{
  "jsonrpc" : "2.0",
  "id" : "reference",
  "result" : {
    "data" : {
      "@type" : "boolean",
      "value" : true
    }
  }
}
```

```
# sample error
{
  "jsonrpc" : "2.0",
  "id" : "reference",
  "result" : false,
  "error" : {
    "code" : -32603,
    "message" : "ldapserver:636 [No route to host (Host unreachable)]",
    "data" : {
      "@type" : "BASIC",
      "reason" : "ldapserver:636"
    }
  }
}
```

C.3. Database Commands

The **savapage-db** command-line tool provides functions for manipulating the database. The tool is located in `/opt/savapage/server/bin/[platform]/` and needs to be executed from a command prompt. The syntax of the command is:

```
usage: [OPTION]
  --db-config-get <NAME>           Gets configuration property value. This is an
                                   advanced option. Please contact SavaPage Support
                                   before use.
  --db-config-set <NAME=VALUE>     Sets configuration property value. This is an
                                   advanced option. Please contact SavaPage Support
                                   before use.
  --db-delete-logs <DAYS>          Deletes application, account transaction and
                                   document log data older than DAYS. A DAYS value of
                                   zero (0) will remove all log data from the system.
  --db-export                      Exports the database to the default backup
                                   location.
  --db-export-to <FILE|DIR>        Exports the database to the specified file or
                                   directory.
  --db-import <FILE>              Imports the database from the specified file.
                                   Deletes any existing data before loading the data.
  --db-init                        Re-initializes the database even if it already
                                   exists.
  --db-run-script <FILE>           Runs SQL statements from the specified script
                                   file. NOTE: Only perform as requested by SavaPage
                                   Support.
  --db-run-sql <STATEMENT>        Runs an SQL statement. NOTE: Only perform as
                                   requested by SavaPage Support.
  -h, --help                      Displays this help text.
```

The command must be run as the **savapage** user. An example:

```
sudo su - savapage
cd server/bin/linux-x64
./savapage-db --db-import /home/john/savapage-backup.zip
```

savapage-db needs exclusive access to the database. It is important that any SavaPage services and processes are stopped before executing a command. Failure to do so will result in a "database in use" error message. The **savapage-db** command is a powerful low-level utility and its use on a production system should be carefully considered. Details of the available commands are discussed below.

C.3.1. db-config-get

With this option the (decrypted) value of a **Configuration Property** can be retrieved. Also see [Section C.1.8, “getConfigProperty” \[259\]](#).

C.3.2. db-config-set

With this option the value of a [Configuration Property](#) can be set (and encrypted). Also see [Section C.1.18](#), “[setConfigProperty](#)” [263].

C.3.3. db-delete-logs

This option delete old log data from the system. This command will permanently delete the following data.

- Document logs - Record all document history and statistics
- Transaction logs - Record all financial history and statistics
- Application logs - Record application status and error messages

The DAYS option determines what data will be deleted. If DAYS is 90, then all log data more than 90 days old will be deleted. A value of zero (0) will remove all historical log data from the system. This is an example:

```
savapage-db --db-delete-logs 90
```

C.3.4. db-export and db-export-to

This option exports the data from the database. The application server must be stopped before performing the export. See [Section C.4](#), “[Stopping and Starting the Server](#)” [272].



Tip

If you want to perform an online backup without stopping the application server you can use the backup function in the Admin Web App.

```
savapage-db --db-export ❶
savapage-db --db-export-to /home/john ❷
savapage-db --db-export-to /home/john/savapage-backup.zip ❸
```

- ❶ The database export file is created in the `/opt/savapage/server/data/backups` directory and the file is named `savapage-export-[date-time].zip`.
- ❷ This option is used to override the default backup directory. The filename will still be named `savapage-export-[date-time].zip`.
- ❸ The full path and filename where the backup is saved is specified.



Note

When executing the command the last line echoed to `stdout` is the canonical path of the database export file.



Caution

If the directory or filename parameters contain spaces, then the argument needs to be quoted.

C.3.5. db-import

This option imports the data (from a previous export) into the database. The application server must be stopped to perform the import. This is an example:

```
savapage-db --db-import /home/john/savapage-backup.zip
```


**Note**

Progress and statistics of the import process are written to `stdout`.

**Warning**

Be careful, existing data in the database are erased.

C.3.6. db-init

This option initializes a database, creating the required tables and initial data. The application server must be stopped before you initialize the database. This is the command:

```
savapage-db --db-init
```

**Warning**

Be careful, existing data in the database are erased.

C.4. Stopping and Starting the Server

Normally there is no need to stop or start the server. This is only required when:

- Performing an **off-line backup**.
- Migrating the an **external database**.
- **Upgrading** the application.

The SavaPage application server may be stopped or started by executing these **systemd** commands:

```
sudo systemctl start savapage.service      ❶
sudo systemctl stop savapage.service       ❷
sudo systemctl restart savapage.service    ❸
systemctl status savapage.service          ❹
```

- ❶ Starts the application server.
- ❷ Stops the application server.
- ❸ Stops and starts the application server in one go.
- ❹ Echoes the run status of the application server.

When SavaPage is installed on a SysV system, use the appropriate **service** commands.

**Important**

When you start the application server, wait approximately 10 seconds for the service to initialize before accessing the Web App interface.

**Note**

When the server is stopped, any SavaPage Web App loses connection and will be unloaded, and a browser message like "This site can't be reached" will be shown. The user must wait till the server is started again, and press any Reload (Try Again) button to resume.

C.5. SSL Key Generation

During the install process, SavaPage generates a self-signed key and SHA-2 certificate issued for the host's machine name. This key is used by default when the system is accessed via HTTPS on port 8632.

The default SSL certificate provides good security, however there are two downsides to using a self-signed certificate, since users accessing the HTTPS site will encounter warnings from the browser.

1. When users access the HTTPS site using a fully-qualified domain name, the browser will issue a “Domain mismatch warning”. To avoid this warning, re-create the self-signed certificate with the machine's fully qualified domain name, see [Section C.5.1, “Re-Create the Self-Signed Certificate” \[273\]](#).
2. The browser will also warn the user that the certificate is not signed by a trusted authority. To overcome this you must obtain a certificate signed by a trusted authority, see [Section C.5.2, “Importing an Existing SSL Certificate” \[273\]](#).

C.5.1. Re-Create the Self-Signed Certificate

The tool **create-ssl-keystore** can be used to re-create the key/certificate (stored in a keystore file) for a different hostname eliminating the browser domain mismatch warning. An example of the command's use:

```
cd /opt/savapage/server/bin
./create-ssl-keystore -f --default --system-name "savapage.mycompany.com"
```

More information is available via the `--help` command line option.

```
usage: [OPTION]...
  --create <FILE>      Creates a specific keystore file.
  -d,--default          Creates the default keystore file
                        /opt/savapage/server/data/default-ssl-keystore.
  -f,--force            Force. Overwrite any existing keystore file.
  -h,--help             Displays this help text.
  --system-name <NAME> The name of the computer/server used for the
                        SSL Certificate. If not set the current
                        computer name is used.
```

C.5.2. Importing an Existing SSL Certificate

If you have an existing SSL certificate you can import it into a Java keystore to be used by SavaPage. Reasons for having an existing signed key include:

- You have obtained a dedicated SSL certificate for use with your SavaPage Application Server.
- Your organization's intranet as served by Internet Information Server (Windows), Apache (GNU/Linux) or another web server uses a certificate that can be re-used for SavaPage.



Note

Unless your intranet server and SavaPage run on the same server (i.e. on different ports), the server name of your intranet server will be different from your SavaPage Application Server. E.g. the intranet address might be `internal.mycompany.com` while the SavaPage Application Server can be reached at `savapage.mycompany.com`. In this case the certificate can only be re-used if it is a so-called wild-card certificate that allows arbitrary subdomains under the `mycompany.com` domain name that it was issued for.

If the SSL certificate is held in a Windows environment you will have a certificate with an attached private key in a so-called PCKS #12 file with `*.p12` or `.pfx` extension³. Please convert this PCKS #12 file to a separate PEM private key and PEM certificate.

³ PKCS #12 is one of the family of standards called Public-Key Cryptography Standards (PKCS), published by RSA Laboratories. It defines a (binary) file format commonly used to store X.509 private keys with accompanying public key certificates, protected with a password-based symmetric key, and is the successor to PFX from Microsoft.



Note

If the certificate with key exist in the certificates store of Windows or IIS Server, you need to export it as a .PFX file first.

On GNU/Linux you will typically already have separate PEM encoded⁴ key and certificate files. In this example, they are called `your_private_key.pem` and `your_certificate.pem` respectively.

However, we are not quite done yet, since we should add the intermediate certificate(s) of the Certificate Authority to the keystore as well. These certificates should be supplied by your CA or are available for download on the CA's web site as a file ending with `.pem` or `.crt`. A single PEM file has to be made, containing your certificate and all the intermediate certificates of your CA.

Use these commands to combine your certificate and the intermediates into one PEM file:

```
cat your_certificate.pem > savapage.pem
cat intermediate_cert_1.pem >> savapage.pem
cat intermediate_cert_2.pem >> savapage.pem
[...]
```

Use this command to combine the private key and the certificates into a single `.p12` file:

```
openssl pkcs12 -export -inkey your_private_key.pem -in savapage.pem -out savapage.p12

Enter pass phrase for your_private_key.pem: (Enter your private key password, if present)
Enter Export Password: (Make up a password)
Verifying - Enter Export Password: (Repeat the password you made up)
```

The **keytool** command used in this section is part of the OpenJDK package as installed on the host. Now, use this command to create a new Java keystore and import the `.p12` or `.pfx` file at hand:

```
keytool -importkeystore -destkeystore my-ssl-keystore -srckeystore savapage.p12 \
-srcstoretype PKCS12

Enter destination keystore password: (Make up another password)
Enter source keystore password: (Enter password you made up with OpenSSL before)
```

At this point your keystore file is ready to use, so follow the instructions in [Section C.5.3, “Installing the Keystore” \[274\]](#) to install it and start serving up your new SSL certificate.

C.5.3. Installing the Keystore

The previous section described how to create a keystore file from an existing SSL certificate. This section describes how to install your keystore so that SavaPage can start serving up your new certificate.

To configure the SavaPage Application Server to use the new key/certificate:

1. Copy your signed keystore onto the server running the SavaPage Application Server. The suggested location is `/opt/savapage/server/custom/my-ssl-keystore`
2. Open the file `/opt/savapage/server/server.properties` with a text editor.
3. Locate the section titled `SSL/HTTP Configuration`.
4. Remove the `#` (hash) comment marker from all lines starting with `"server.ssl"`.
5. Define the location of your keystore, keystore password and key password as chosen previously. The file should look something like this:

⁴ PEM or Privacy Enhanced Mail is a Base64 encoded DER certificate. PEM certificates are frequently used for web servers as they can easily be translated into readable data using a simple text editor. Generally when a PEM encoded file is opened in a text editor, it contains very distinct headers and footers.

```
server.ssl.keystore=custom/my-ssl-keystore  
server.ssl.keystore-password=password  
server.ssl.key-password=password
```

6. Restart the SavaPage Application Server and verify all is working. If the server fails to start, error messages will be recorded in logs located in the server's logs directory.

Appendix D. Capacity Planning



This section discusses capacity planning considerations so administrators can plan future infrastructure requirements and make decisions about how to deploy the application. SavaPage is designed to be self-maintaining, however it is important that the administrator understands the disk-space requirements and how this changes overtime.

D.1. Database Sizing and Growth

Special attention is needed to make sure there is enough disk space to hold a growing database. Database growth is very dependent on the usage patterns and therefore can differ significantly from site to site.

Although, there is some overhead for data like Users, Proxy Printers and Queues, this data is static and does not grow over-time. The majority of database growth is caused by logging the document flow.

So, the best prediction of database growth is based on the estimated number of documents printed to SavaPage and Proxy Printers, and exported to PDF.

The table below provides an indication of growth per 10,000 jobs for SavaPage and Proxy Printing and PDF export. Combining these numbers with your estimate of user activity result in a growth estimate.

Job type	Increase per 10,000 jobs
SavaPage printing	15 MB
Proxy Printer printing	20 MB
SavaPage Financial	5 MB
PDF export	20 MB

Table D.1. Database size increase metrics per document flow.

To demonstrate how to estimate database growth we make a number of assumptions. Please adjust these assumptions to suit your organization. The assumptions are:

- 1 job for each job type per user per day
- 20 working days in a month
- Therefore, 20 jobs for each job type per user per month

Here is a sample database growth calculation based on a 500 user site:

1. Calculate the total number of jobs expected for the month (i.e. the total number of users multiplied by the number of jobs). So in this example, SavaPage is handling 10,000 jobs for each job type a month.
2. Calculate the monthly growth rate by dividing the jobs per month by 10,000 and then multiplying by the database growth rate:
 - SavaPage printing : $10,000 / 10,000 * 15 = 15\text{MB per Month}$
 - Proxy Printer printing: $10,000 / 10,000 * 20 = 20\text{MB per Month}$
 - SavaPage Financial: $10,000 / 10,000 * 5 = 5\text{MB per Month}$
 - PDF export: $10,000 / 10,000 * 20 = 20\text{MB per Month}$

Therefore in this situation the database will grow by approximately $15+20+5+20=60$ MB per month.

3. To estimate the growth per year, multiply the above by 12. Therefore in this situation, the database will grow by $60*12=720$ MB per year.



Tip

You can limit database growth by purging old log data after an automatic database backup. In our example, when you set the number of days document logs are held to 365, database increase will maximize to 720MB. See [Figure 4.94, “Admin Web App: Options - Automatic Backups” \[148\]](#)

D.2. SafePages Sizing and Growth

SafePages are transient. They serve as scratchpad to accumulate and edit SavaPage print jobs. After ProxyPrinting and PDF exporting is done the user will normally clear the scratchpad for a new session. Of course this does not hold for personal Letterheads. Take in consideration that SafePages (including Letterheads) from logically deleted users are removed. This all makes the calculation of required disk space a simple linear function of the number of active users times the average size of a user's SafePages home. Since an average 10-page print job takes about 1 MB to hold in store, making a reservation of 10 MB per user seems fair enough.

D.3. Network Bandwidth Planning

With modern switched Ethernet networks, bandwidth is rarely a factor when planning SavaPage deployments. The bandwidth consumed by SavaPage is usually dwarfed by the print document data - e.g. the PostScript and PDF spool data sent across the network. Bandwidth does however become a consideration when planning deployments crossing physical site boundaries such as networks linked via a WAN.

SavaPage uses JSON based HTTP Requests for communication between browser-to-server (Ajax)¹ and server-to-browser (Comet)². This protocol is very bandwidth efficient and designed to work well on low bandwidth and high latency networks.

¹ *Ajax* (an acronym for Asynchronous JavaScript and XML) is a group of techniques to create asynchronous web applications. With Ajax, web applications send data to, and retrieve data from, a server asynchronously using an XMLHttpRequest object. Despite the name, the use of XML is not needed, and JSON is often used instead. Also, requests do not need to be asynchronous.

² *Comet* (or “Reverse Ajax”) is a web application model in which a long-held HTTP request allows a web server to push data to a browser, without the browser explicitly requesting it.

Appendix E. URL Cheat Sheet



Path	Description / Parameters / Examples
/	<p>User Web App</p> <p>https://savapage:8632/</p>
user/	<p>User Web App</p> <p>sp-user=[user] sp-lang=[de en es fr nl ru ..] sp-ctry=[DE US ES FR NL RU ..] sp-login=[name id nfc-local yubikey] sp-login-local (exclude OAuth, use local login modes only) sp-log=[warn info debug]</p> <p>https://savapage:8632/user?sp-user=tom&sp-lang=en&sp-ctry=US https://savapage:8632/user?sp-user=tom https://savapage:8632/user?sp-lang=ru&sp-ctry=RU</p> <p>Login with Username, ID Number, Local NFC Card or YubiKey:</p> <p>https://savapage:8632/user?sp-login=name https://savapage:8632/user?sp-login=id https://savapage:8632/user?sp-login=nfc-local https://savapage:8632/user?sp-login=yubikey https://savapage:8632/user?sp-login-local</p>
oauth/<provider> user/oauth/<provider>	<p>User Web App login with OAuth provider.</p> <p>https://savapage:8632/oauth/google https://savapage:8632/oauth/smartschool</p> <p>https://savapage:8632/user/oauth/google https://savapage:8632/user/oauth/smartschool</p>
admin/ jobtickets/ pos/	<p>Admin Web App</p> <p>Job Tickets Web App</p> <p>Point-of-Sale Web App</p> <p>sp-user=[user] sp-lang=[de en es fr nl ru ..] sp-ctry=[DE US ES FR NL RU ..] sp-login=[name id nfc-local yubikey] sp-log=[warn info debug]</p> <p>https://savapage:8632/admin?sp-user=admin https://savapage:8632/jobtickets?sp-user=mary https://savapage:8632/pos?sp-user=dmitri&sp-lang=ru</p>
printers/[queue] printers/	<p>Printer Queue</p> <p>Default Printer Queue</p> <p>ipp://savapage:8631/printers/ ⓘ</p>

Path	Description / Parameters / Examples
	<div> <div> <code>ipps://savapage:8632/printers/</code> <code>http://savapage:8631/printers/</code> <code>https://savapage:8632/printers/</code> </div> <div> ^❶ IPP 1.1 scheme: supported by all major operating systems except Windows. ^❷ The SavaPage SSL certificate needs to be trusted by the client workstation^a. See Section C.5.3, “Installing the Keystore” [274]. ^❸ IPP 1.0 scheme: supported by all major operating systems. ^❹ The SavaPage SSL certificate needs to be trusted by the client workstation. See Section C.5.3, “Installing the Keystore” [274] </div> </div>
<code>printers/internet/user/[number]/uuid/[uuid]</code>	<p>The Printer Device URI path for Internet Print. Parameters are <i>not</i> query parameters but are part of the path.</p> <ul style="list-style-type: none"> [number]: the User ID Number. [uuid]: the User UUID. <div> <code>ipps://example.com/printers/internet/user/12345 \</code> <code>/uuid/b0a2f092-8c5b-11e5-a6fb-406186940c49</code> </div>
<code>ios/install/</code>	<p>iOS Web Clip Install</p> <div> <code>https://savapage:8632/ios/install</code> </div>
<code>docs/manual/</code>	User Manual
<code>docs/licenses/</code>	License Information
<code>callback/</code>	Web API Callback
<code>callback/payment/[live test]/[pluginId]</code>	Web API Callback for Payment Gateway
<code>client/</code>	Download of shared Client Files . A link to a directory downloads the zipped content.
<code>xmlrpc/</code>	<p>The secure only XML-RPC endpoint:</p> <div> <code>https://savapage:8632/xmlrpc</code> </div> <p>See Section C.2.1, “XML-RPC” [266].</p>

^a When the SSL certificate is not trusted a "client-error-not-possible" situation will occur when adding the printer.

Table E.1. SavaPage URL Cheat Sheet

Appendix F. File Locations



Path	Description
<code>/opt/savapage/</code>	Install directory.
<code>client/</code>	Client applications.
<code>app/</code>	Chapter 7, <i>User Client</i> [186].
<code>config/</code>	Section 7.2, “User Client Deployment” [188] <ul style="list-style-type: none">• <code>client.properties</code>
<code>jmx/</code>	Section 4.10.14.3, “JMX Agent” [150].
<code>linux/</code>	.desktop templates.
<code>providers/</code>	Section 1.3.1.6, “Information Provider” [7]
<code>cups/</code>	CUPS notifier.
<code>nfc/</code>	Section B.3, “Network Card Reader Service” [252].
<code>server/</code>	Section 13.1.1.3, “Internal Admin Password” [216] <ul style="list-style-type: none">• <code>admin.properties</code> Section 4.10.6.1, “Google Cloud Printer Registration” [133] <ul style="list-style-type: none">• <code>gcp.properties</code> Section 4.10.14.3, “JMX Agent” [150] <ul style="list-style-type: none">• <code>jmxremote.*</code> Section 4.13.3, “Community” [165] <ul style="list-style-type: none">• <code>savapage.membercard</code> Section 9.3, “Advanced Configuration” [192] <ul style="list-style-type: none">• <code>server.properties</code>
<code>bin/</code>	<ul style="list-style-type: none">• Section C.1, “Server Commands” [253].• Section C.3, “Database Commands” [270].
<code>custom/</code>	Section 18.3, “JVM Tuning” [242] <ul style="list-style-type: none">• <code>app-server.conf</code> Section C.5, “SSL Key Generation” [273] <ul style="list-style-type: none">• <code>my-ssl-keystore</code> Section 16.1, “Custom Web App” [227] <ul style="list-style-type: none">• <code>web.properties</code> Appendix K, <i>PPD Extensions</i> [292]

File Locations

Path	Description
	<p>Section L.3, “IPP Localization” [310]</p> <p>Section 16.2, “Email Templates” [230]</p> <p>Section 16.1.1.4, “Custom i18n” [230]</p>
data/	<p>Section 13.4, “SSL Passwords” [218]</p> <ul style="list-style-type: none"> • default-ssl-keystore.* <p>Section 13.6, “Encrypted Secrets” [218]</p> <ul style="list-style-type: none"> • encryption.properties
backups/	Section 4.10.13, “Backups” [148].
conf/	<p>Section 4.10.1.4, “Internal Groups” [123]</p> <ul style="list-style-type: none"> • internal-groups.txt : <p>Section 11.1.18, “User Alias” [208]</p> <ul style="list-style-type: none"> • username-aliases.txt :
docs/	Accessible via URL: Appendix E, URL Cheat Sheet [278]
email-outbox/	Outgoing email queue.
internal/	
Derby/	Section 1.2.1.3, “Database” [4].
letterheads/	<p>Section 3.6, “Letterheads” [57]</p> <p>Also see Section 9.3.2, “Alternative File Locations” [193]</p>
safepages/	Also see Section 9.3.2, “Alternative File Locations” [193]
print-jobtickets/	Chapter 5, Job Tickets Web App [174].
examples/	Section 10.2, “Printing with AirPrint” [199]
ext/	<p>Appendix M, SavaPage Plug-ins [311]</p> <ul style="list-style-type: none"> • savapage-ext-*.properties
lib/	Extension JAR files.
logs/	Extension log files.
lib/	<p>Main WAR and JAR files.</p> <ul style="list-style-type: none"> • log4j.properties
logs/	Main log files.
tmp/	Also see Section 9.3.2, “Alternative File Locations” [193].

Table F.1. SavaPage File Locations

Appendix G. Printable File Types



G.1. Standard File Types

SavaPage printer supports a number of common file types out of the box as summarized in [Table G.1, “Standard Printable File Types” \[282\]](#).

Extension	Type
pdf	Portable Document Format The PDF must not be encrypted and not password protected.
ps	PostScript DRM protected PostScript is rendered for ProxyPrinting only. See Section 10.7, “Printing Encrypted PDF” [204] .
xps	XML Paper Specification See Section G.1.1, “XPS to PDF Installation Instructions” [283] .
html	Hypertext Markup Language CSS 2.1 is fully supported.
txt	Text File
bmp	Bitmap
gif	Graphic Interchange Format For Animated GIF each image is rendered separately.
jpg, jpeg, jpe	JPEG/JIFF Image
png	Portable (Public) Network Graphic
svg	Scalable Vector Graphics The <code>librsvg2-bin</code> package is needed for this option. On Debian based systems use this command to install: <code>sudo apt-get install librsvg2-bin</code>
tiff, tif	Tagged Image Format File Multi-page tiff is supported.

Table G.1. Standard Printable File Types



Note

The Default Paper Size, as shown in [Figure 4.101](#), “Admin Web App: Options - Default Paper Size” [152], is used as the paper size for the printed document of a Printable File Type which itself does not have a document structure with a clearly defined page size. These types typically include HTML, TXT and *images*.

G.1.1. XPS to PDF Installation Instructions

XML Paper Specification (XPS) is an XML based electronic paper format originally developed by Microsoft to serve as a PDF alternative. XPS files are usually created using “Microsoft XPS Document Writer” in a Windows environment.

SavaPage uses the **xpstopdf** command from the `libgxps`¹ package to convert XPS documents to PDF format. Check if this package is installed by entering the command: `xpstopdf --help`

On Debian based systems you can install the package with the command:

```
sudo apt-get install libgxps-utils
```



Note

Before XPS to PDF can be used it must be enabled. See [Figure 4.103](#), “Admin Web App: Options - Converters” [153].

G.2. Advanced File Types

SavaPage printer supports additional file types using the PDF converter of LibreOffice as summarized in [Table G.2](#), “Advanced Printable File Types” [283]. Check if LibreOffice is installed by entering the command:

```
libreoffice --version
```

LibreOffice can easily be installed with the standard installer of the GNU/Linux host. On Debian based systems you can use the command line to install the packages needed. For example:

```
# The core package
sudo apt-get install libreoffice-core unoconv

# ... or on Debian jessie, to get the latest LibreOffice version
sudo apt-get install libreoffice-core unoconv -t jessie-backports

# To convert text, spreadsheet and presentation documents
sudo apt-get install libreoffice-writer libreoffice-calc libreoffice-impress

# Install fonts so conversion of non-embedded font sources does not fallback to DejaVu
sudo apt-get install fonts-cantarell fonts-croscore fonts-crosextra-caladea
sudo apt-get install fonts-crosextra-carlito fonts-dejavu fonts-droid
sudo apt-get install fonts-liberation fonts-noto fonts-opensymbol gsfonts

# Install even more fonts like the Ubuntu Font Family
sudo apt-get install ttf-ubuntu-font-family

# ... and Microsoft True Type Core Fonts
sudo apt-get install ttf-mscorefonts-installer
```

Extension	Type
rtf	Rich Text Format

¹`libgxps` [<https://wiki.gnome.org/libgxps>] is a library for handling and rendering XPS documents.

Extension	Type
doc	Microsoft Word 97/2000/XP/2003
xls	Microsoft Excel 97/2000/XP/2003
ppt	Microsoft PowerPoint 97/2000/XP/2003
docx	Microsoft Word 2007/2010 XML
xlsx	Microsoft Excel 2007/2010 XML
pptx	Microsoft PowerPoint 2007/2010 XML
odt	ODF Text Document
ods	ODF Spreadsheet
odp	ODF Presentation
sxw	OpenOffice.org 1.0 Text Document
sxc	OpenOffice.org 1.0 Spreadsheet
sxi	OpenOffice.org 1.0 Presentation

Table G.2. Advanced Printable File Types



Warning

PDF conversion of Microsoft documents may not give correct results in all cases.



Note

Before LibreOffice can be used it must be enabled. See [Figure 4.103](#), “Admin Web App: Options - Converters” [153].

Appendix H. Upgrading from a Previous Version



This appendix describes the SavaPage standard upgrade procedure.

H.1. Upgrading the Server

SavaPage supports upgrades using a simple *install-over-the-top* procedure. We recommend reviewing all steps prior to commencing the upgrade procedure.

1. Download the SavaPage installer for your platform. In accordance with best practice we recommend that you archive your install programs just in case you need to reinstall in the future or roll back to a previous version.
2. Take some time to read the release notes for this version as they may highlight considerations during upgrades.
3. Schedule approximately 10 minutes downtime. It is suggested to choose a time of day with minimal network activity. If there is a large volume of data in the system (for example if the system has been running for more than a year, or there are several thousands of users) the upgrade may take longer. With very large installations it may be appropriate to schedule an hour or more of downtime.
4. Take a point-in-time backup of the data by pressing the **Backup Now** located under **Options → Backups**. This will ensure you have a copy of the important data.
5. As a precaution on very large systems, we recommend backing up the whole SavaPage install directory. Existing overnight backups may have taken care of this task, however take a few moments to grab an up-to-date backup now. For example, create a zip archive of the directory `/opt/savapage`
6. Run the installer downloaded in step 1 and install into the *same* location as the existing install, like `/opt/savapage`.
7. After the install has completed allow a few minutes before accessing the system. The system may need to perform a database upgrade and this will be performed in the background. If you try to access the application while a database upgrade is in progress a message displaying the upgrade status will be displayed.



Important

Do not shutdown the application while an upgrade is in progress. Wait for the upgrade to complete.



Note

Sometimes a new SavaPage version performs changes on the database schema. In that case a database backup is performed automatically before the upgrade. The backup file is located at `/opt/savapage/server/data/backups/`. The file name is formatted as `schema-[nn]-upgrade-backup-[time-stamp].zip`, where `[nn]` is the database schema version before the upgrade.

H.2. Upgrading Client Printer Drivers

Although upgrading locally installed SavaPage Printer Drivers is not strictly required, we strongly recommend doing so. We strive to maintain backwards compatibility between versions, so in most cases these drivers will continue to function, but to take advantage of new features they must be upgraded.

H.3. Testing the Upgrade

After the install is complete, log into the system and perform some tests to ensure all is working as expected.

Appendix I. Migrating to a New Server



Migrating to a new server is a major task. Administrators should block out at a minimum two hours, and should select a time where downtime will be of minimum disruption to end-users.

This section describes how to migrate SavaPage to a new system so that all data is moved to the new system. To ensure a smooth migration it is strongly recommended that the versions of SavaPage on both the old and new servers are the same. The easiest way to achieve this is to upgrade the old server to the latest version, and then install the latest version on the new server.

Please read the sections below in full before conducting your migration.

I.1. Upgrade Old Server

Upgrade the old server to the latest version:

1. Download the latest available version available from the [SavaPage Website](https://www.savapage.org)¹
2. Install the upgrade by following the steps in [Appendix H, *Upgrading from a Previous Version* \[285\]](#).
3. After the upgrade is complete, check that everything is working as expected.

I.2. Install New Server

Install the latest version of SavaPage on the new server.

1. Make your CUPS printers on the new server identical as the ones on the old server, and test the CUPS queues before installing SavaPage. Use exactly the same names for the CUPS queues. If you deviate you might need to rename Proxy Printers after installation, as explained in [Section I.4, “Rename Printers” \[287\]](#).
2. Follow the instructions at [Chapter 2, *Server Installation* \[11\]](#). Complete the configuration steps, including the user import. Although importing the users is not strictly required, as this data will be overridden after data migration, it does confirm that your new server has the correct network connectivity. Of course you are free to synchronize with a smaller user group to proof the connectivity.
3. Compare the content of the `/opt/savapage/server/server.properties` file from the old server with the one on the new server, and change the file on the new server where needed.
4. Import your Member Card file into the new server. See [Section 4.13.3, “Community” \[165\]](#).

I.3. Migrate Data to New Server

The simplest way to migrate the data to the new server is to use the backup and restore process.

1. Backup the data from the old system using the [Admin Web App](#), or see the instructions at [Section C.3.4, “db-export and db-export-to” \[271\]](#) for the command-line variant.
2. Copy the backup zip file created in the backup step onto the new server.
3. Stop the SavaPage Server by running the stop script. See [Section C.4, “Stopping and Starting the Server” \[272\]](#).
4. Restore the data from the old system into SavaPage on the new server by following the [Database Import](#) instructions. The import commands need to be run as the savapage user.

¹ <https://www.savapage.org>

5. If present, migrate the **Google Cloud Print Service** parameters by copying the `gcp.properties` file at location `/opt/savapage/server/` from the old server to the same location on the new server. Make sure this file is owned by `savapage`, and restrict access by executing:

```
sudo chown savapage:savapage gcp.properties
sudo chmod 600 gcp.properties
```

6. Migrate the database **Encryption Keys** by copying the `encryption.properties` file at location `/opt/savapage/server/data/` from the old server to the same location on the new server. Overwrite the existing file on the new server. Make sure this file is owned by `savapage`, and restrict access by executing:

```
sudo chown savapage:savapage encryption.properties
sudo chmod 600 encryption.properties
```

7. If present, migrate the **User Name Aliases** and **Internal Groups** by copying the `username-aliases.txt` and `internal-groups.txt` files at location `/opt/savapage/server/data/conf/` from the old server to the same location on the new server. If the alias file depends on users from a **User Source on the local Unix system**, be sure that these users also exist on the new server.
8. If present, migrate any messages in the email outbox by copying the files at location `/opt/savapage/server/data/email-outbox/` from the old server to the same location on the new server.
9. Migrate all customization files to the new server. See **Chapter 16, Customization** [227] and **Appendix K, PPD Extensions** [292].
10. Start the SavaPage Server by running the start script. See **Section C.4, “Stopping and Starting the Server”** [272].
11. Check that all data has been migrated correctly and the system works as expected by comparing **Users** and **Documents** data in the old and new **Admin Web App**.

I.4. Rename Printers

If you changed the CUPS printer names on the new server, you may want to rename the existing **Proxy Printer** entries in SavaPage so that the printing history and settings are maintained. See **Figure 4.47, “Admin Web App: Proxy Printer - Rename”** [110] for details about the rename action.

I.5. Update SavaPage Printers

If the server’s name and/or IP address has changed then it is necessary to update the connection details for SavaPage Printers on user workstations. See **Section 10.1.2, “SavaPage Printer Installation”** [195] for details.

Appendix J. Advanced LDAP Configuration



SavaPage supports the following LDAP server types out-of-the-box:

- OpenLDAP
- Apple Open Directory
- Novell eDirectory
- Microsoft Active Directory

The basic configuration options for these types are discussed at [Section 4.10.1.2, “LDAP” \[121\]](#). However, other server/schema types can be supported by defining the fields to query and the LDAP searches to perform. These options are configured by adjusting entries in the [Config Editor](#) of the Admin Web App. The following configuration properties are available:

Configuration property	Description
ldap.schema.user-name-field	The LDAP field that contains the user's username.
ldap.schema.user-full-name-field	The LDAP field that contains the user's full name.
ldap.schema.user-email-field	The LDAP field that contains the user's email address.
ldap.schema.user-department-field	The LDAP field that contains the user's department.
ldap.schema.user-office-field	The LDAP field that contains the user's office location.
ldap.schema.user-name-search	<p>The LDAP search to retrieve the user. The {0} in the search is replaced with * when listing all users, and [username] when searching for a specific user.</p> <p>If no search is defined the default is ([userNameField]={0}).</p> <p>IMPORTANT: The search must include the {0} value.</p>
ldap.schema.group-name-field	The LDAP field that contains the group's name.
ldap.schema.group-member-field	The LDAP field that contains the group members.
ldap.schema.group-search	<p>The LDAP search to retrieve the group. The {0} in the search is replaced with * for all group searches.</p> <p>If no search is defined, the default is ([groupMemberField]={0}), which means get all entries with at least one member.</p> <p>IMPORTANT: The search must include the {0} value.</p>
ldap.schema.posix-groups	If Y, then the group member field contains the user's username. If N, then the group member field contains the user's DN.

Table J.1. LDAP Configuration Properties

J.1. LDAP Server Default Configuration

When a particular LDAP server type is selected (e.g. Novell eDirectory), SavaPage uses the following defaults to query the LDAP server. These defaults can be used as a starting point for customizing the LDAP searches or for supporting other server types.

J.1.1. OpenLDAP

If the LDAP server is configured to support OpenLDAP based authentication then this schema type can be used. The following defaults are used.

Configuration property	Default value
ldap.schema.user-name-field	uid
ldap.schema.user-full-name-field	cn
ldap.schema.user-email-field	mail
ldap.schema.user-department-field	departmentNumber
ldap.schema.user-office-field <i>This item is not set.</i>	
ldap.schema.user-name-search	(uid={0})
ldap.schema.group-name-field	cn
ldap.schema.group-member-field	member
ldap.schema.group-search	(&(cn={0})(objectClass=groupOfNames))
ldap.schema.posix-groups	N

Table J.2. OpenLDAP Default Settings

J.1.2. Apple Open Directory

If the LDAP server is configured to support Apple Open Directory based authentication then this schema type can be used. The following defaults are used.

Configuration property	Default value
ldap.schema.user-name-field	uid
ldap.schema.user-full-name-field	cn
ldap.schema.user-email-field	mail
ldap.schema.user-department-field	departmentNumber
ldap.schema.user-office-field <i>This item is not set.</i>	
ldap.schema.user-name-search	(uid={0})
ldap.schema.group-name-field	cn
ldap.schema.group-member-field	memberUid
ldap.schema.group-search	(memberUid={0})
ldap.schema.posix-groups	Y

Table J.3. Apple Open Directory Default Settings

J.1.3. Novell eDirectory Defaults

If the LDAP server is a Novell eDirectory then the following defaults are used¹.

Configuration property	Default value
ldap.schema.user-name-field	cn
ldap.schema.user-full-name-field	fullName
ldap.schema.user-email-field	mail
ldap.schema.user-department-field	OU
ldap.schema.user-office-field	1
ldap.schema.user-name-search	(& (cn = { 0 }) (objectClass = person))
ldap.schema.group-name-field	cn
ldap.schema.group-member-field	member
ldap.schema.group-search	(& (member = { 0 }) (objectClass = groupOfNames))
ldap.schema.posix-groups	N

Table J.4. Novell eDirectory Default Settings

J.1.4. Microsoft Active Directory Defaults

If the LDAP server is a Microsoft Active Directory then the following defaults are used².

Configuration property	Default value
ldap.schema.user-name-field	sAMAccountName
ldap.schema.user-full-name-field	displayName
ldap.schema.user-email-field	mail
ldap.schema.user-department-field	department
ldap.schema.user-office-field	physicalDeliveryOfficeName
ldap.schema.user-name-search	(& (sAMAccountName = { 0 }) (objectCategory = person) (objectClass = user) (sAMAccountType = 805306368) { 1 }) The extra { 1 } in the search is replaced with an optional filter to fetch enabled users only (see ldap.allow-disabled-users).
ldap.schema.group-name-field	sAMAccountName
ldap.schema.group-member-field	member
ldap.schema.group-search	(& (sAMAccountName = { 0 }) (objectCategory = group))
ldap.schema.posix-groups	N

Table J.5. Microsoft Active Directory Default Settings

¹ The list of standard Novell eDirectory user fields can be found on NDK: Novell eDirectory Schema Reference [https://www.novell.com/documentation/developer/ndslib/schm_enu/data/h4q1mn1i.html#h4q1mn1i].

² The list of standard Active Directory user fields can be found on the Microsoft web site <https://msdn2.microsoft.com/en-us/library/ms683980.aspx>.

Configuration property	Default value / Description
ldap.disabled-users.allow	N If Y, then disabled users are accepted in user name searches. If N, they are ignored.
ldap.schema.dn-field	distinguishedName The LDAP field that contains the Distinguished Name (DN).
ldap.schema.user-name-group-search	<code>(&(memberOf={0}))(objectCategory=person)(objectClass=user)(sAMAccountType=805306368){1})</code> This is the LDAP search to retrieve the users from a group. The {0} in the search is replaced with the DN of the user. The {1} in the search is replaced with an optional filter to fetch enabled users only (see ldap.allow-disabled-users). IMPORTANT: The search must include the {0} and {1} value.
ldap.schema.nested-group-search	<code>(&(memberOf={0}))(objectCategory=group)</code> This is the LDAP search to retrieve the nested groups from a group. The {0} in the search is replaced with the DN of the group. IMPORTANT: The search must include the {0} value.

Table J.6. Microsoft Active Directory Custom Settings



Important

Active Directory field names must be in the `Ldap-Display-Name` format. For example, if you want to use the *Employee-Number* field, then the field name entered should be *employeeNumber* as shown on the Employee-Number attribute page <https://msdn2.microsoft.com/en-us/library/ms675662.aspx>.

Appendix K. PPD Extensions

Vendor specific PPD option keywords are generally not mapped to IPP attributes by CUPS. That's why we do not get IPP attributes for finishings (staple, punch, fold, booklet) or collating delivered, when we ask CUPS for an IPP printer description. To bridge this gap we built our own mapping by means of a so-called PPD Extension `.ppde` file. With this mapping we are able to identify printer capabilities based on IPP and feed CUPS the vendor specific PPD options as IPP attributes when sending a print job. When printing, these IPP disguised PPD options are neatly applied by CUPS in the context of the PPD driver, so the right PostScript / PCL snippets are injected in the spool file.

PPD Extension files reside in the `/opt/savapage/server/custom/cups/` directory. An annotated type-model-version.ppde.template file is installed there for your convenience. The `.ppde` file can be linked to a Proxy Printer. See [Section 4.8.2, “Edit Proxy Printer”](#) [105]. When linked, the mapped PPD options will appear in the [Printer Settings Dialog](#).



Warning

PPD Extensions is an advanced feature. Please consult your SavaPage Community Representative before implementing.

K.1. PPD to IPP Mappings

The PPDE file holds mappings of original PPD file options to their IPP attribute and values counterparts. Mapped IPP attributes and values can either be IANA registered, or [Internal SavaPage Extensions](#). This is the syntax:

```
*VENOption IppAttribute      ❶  
*VENOption *VENOptionValue IppValue  ❷
```

- ❶ `VENOption` must be replaced by its PPD equivalent (the `VEN` prefix stands for *vendor* specific).
- ❷ A `VENOption` / `VENOptionValue` pair, whose values must be copied from the vendor PPD.

`IppValue` is the IPP value equivalent. The optional asterisk `*` prefix of `VENOptionValue` tells if the value is the default.

The `IppValue` *must* be unique in the `VENOption` set, while the `VENOptionValue` can be used more than once. *In this way different IPP values can be mapped to the same PPD value.*

K.1.1. Mapping PPD to IPP

IANA IPP attributes available for mapping are presented in the sections below.

K.1.1.1. media-source

```
*VENMediaSource media-source  
*VENMediaSource *VENAuto auto  
*VENMediaSource VENTop top  
*VENMediaSource VENMiddle middle  
*VENMediaSource VENBottom bottom  
*VENMediaSource VENBypassTray by-pass-tray  
*VENMediaSource VENManual manual  
*VENMediaSource VENTray1 tray-1  
# tray-2 ... tray-10
```

K.1.1.2. media-type

```
*VENMediaType media-type
```

```
*VENMediaType *VENPaper paper
*VENMediaType VENTransparency transparency
*VENMediaType VENLabels labels
*VENMediaType VENLetterhead letterhead
```

K.1.1.3. output-bin

```
*VENOutputBin output-bin
*VENOutputBin *VENAuto auto
*VENOutputBin VENBottom bottom
*VENOutputBin VENCenter center
*VENOutputBin VENTop top
*VENOutputBin VENFaceDown face-down
*VENOutputBin VENFaceUp face-up
*VENOutputBin VENLargeCap large-capacity
*VENOutputBin VENLeft left
*VENOutputBin VENMiddle middle
*VENOutputBin VENRear rear
*VENOutputBin VENSIDE side
*VENOutputBin VENStacker1 stacker-1
# stacker-2 ... stacker-5
*VENOutputBin VENTray1 tray-1
# tray-2 ... tray-5
```

K.1.1.4. print-color-mode

```
*VENPrintColorMode print-color-mode
*VENPrintColorMode *VENMonochrome monochrome
*VENPrintColorMode VENColor color
```

K.1.1.5. print-scaling

Since CUPS does not map the IPP `print-scaling` attribute to vendor PPD values, SavaPage falls back to the CUPS `fit-to-page`¹ boolean attribute to scale documents by default. Value `true` scales the document up or down to fit the selected media. Value `false` preserves the physical size of the printed document and crops any content outside the selected media.

You can override this behavior by a custom vendor mapping, as shown below.

```
*VENPrintScaling print-scaling
*VENPrintScaling *VENFit fit
*VENPrintScaling VENNone none
```

See [Section 3.5.1, “Page Scaling” \[55\]](#).

K.1.1.6. sheet-collate

```
*VENCollate sheet-collate
*VENCollate *VENCollated collated
*VENCollate VENUncollated uncollated
```

When `sheet-collate` is not mapped, SavaPage generates a single PDF and applies a one-copy print. The PDF is a concatenation of the requested number of copies with pages arranged in the right collate order.

K.1.1.7. sides

```
*VENSides sides
```

¹ <https://www.cups.org/doc/options.html>

```
*VENSides VOneSided          one-sided
*VENSides *VENTwoSidedLongEdge two-sided-long-edge
*VENSides VENTwoSidedShortEdge two-sided-short-edge
```

K.1.2. Mapping PPD to IPP Extensions

PPD Options can be mapped to [Internal IPP Extensions](#). For example:

```
*VENStapleOption org.savapage-finishings-staple ❶
*VENStapleOption *VENNone          3 ❷
*VENStapleOption VENTopLeft        20 ❸
*VENStapleOption VENBottomLeft     21 ❹
```

- ❶ VENStapleOption is mapped to [org.savapage-finishings-staple](#).
- ❷ VENNone value of VENStapleOption is mapped to IPP enum value 3 (none). The asterisk * tells this options is the default.
- ❸ VENTopLeft value of VENStapleOption is mapped to IPP enum value 20 (staple-top-left).
- ❹ VENBottomLeft value of VENStapleOption is mapped to IPP enum value 21 (staple-bottom-left).

K.1.3. Restricting Standard Options

Standard CUPS/IPP options, like `number-up`, are independent of PPD, and available for all printers types. The values of these options can be restricted with PPDE syntax. Below are examples for options currently supported by SavaPage.

K.1.3.1. number-up

For example, option `number-up` can be restricted to values 1 (default), 2 and 4 like this:

```
*number-up number-up
*number-up *1 1
*number-up 2 2
*number-up 4 4
```

K.2. PPD Rules

Sometimes, simple one-to-one PPD to IPP mappings do not suffice to get the right options to the PPD driver. In that case PPD rules can be of help. PPD rules take independent IPP (or PDF document) options as input, and return dependent PPD (or CUPS) options.

K.2.1. Generic PPD Rules

Generic PPD Rules add, assign or substitute PPD option values.

K.2.1.1. SPEXtra

The `SPEXtra` rule adds one or more PPD options to a print request, depending on one or more IPP options. The rule is formatted like this:

```
*SPEXtra/<attrib>/<value>: <mnemonic> \ ❶
    <attrib>/[!]<value> ... \ ❷
    *<option>/<value> ... \ ❸
```

- ❶ The rule prefix, an IPP attribute/value pair as main independent variable, and identifying `<mnemonic>`.
- ❷ Optionally one or more extra IPP attribute/value pairs as independent variables. An optional `!` before a value negates it, and selects all other attribute values.

- ❸ One or more PPD option/value pairs as dependent variables. These option pairs are added to the print request, when all independent variables are present.

As an example, some [org.savapage-finishings-punch](#) rules for “Canon iR-ADV 8285/8295 UFR II” are shown below.

```
*SPExtra/org.savapage-finishings-punch/3: punch-none \
  *CNPunch/None

*SPExtra/org.savapage-finishings-punch/74: punch-dual-left \
  *CNPunch/Left

*SPExtra/org.savapage-finishings-punch/82: punch-quad-left \
  *CNPunch/Left
```

The rule can also be used to *replace* existing options. For example: [org.savapage-finishings-booklet](#) rules, again for “Canon iR-ADV 8285/8295 UFR II”. The snippet below *adds* extra options CNSaddleStitch and BindEdge, and *replaces* the media option.

```
*SPExtra/org.savapage-finishings-booklet/topleft-totop: booklet-a3 \
  media/iso_a3_297x420mm \
  *media/iso_a4_210x297mm \
  *CNSaddleStitch/True \
  *BindEdge/Left

*SPExtra/org.savapage-finishings-booklet/topleft-totop: booklet-a4 \
  media/iso_a4_210x297mm \
  *media/iso_a5_148x210mm \
  *CNSaddleStitch/True \
  *BindEdge/Left
```

You can even replace a single IPP option value. For example, if you want to print NA Letter as A4 (relying on default [print-scaling](#)) you can use this rule:

```
*SPExtra/media/na_letter_8.5x11in: letter-to-a4 \
  *media/iso_a4_210x297mm \
```

K.2.1.2. SPSubst

This rule is meant to assign a native PPD value to an IPP attribute, and is formatted like this:

```
*SPSubst/<attrib>/<value>: <mnemonic> \ ❶
  <attrib>/[!]<value> ... \ ❷
  *<value> \ ❸
```

- ❶ The rule prefix, an IPP attribute/value pair as main independent variable, and identifying <mnemonic>.
- ❷ Optionally one or more extra IPP attribute/value pairs as independent variables. An optional ! before a value negates it, and selects all other attribute values.
- ❸ The PPD option value as dependent variable. This value is assigned as PPD option to the main IPP attribute, when all independent variables are present.

As an example, some [sheet-collate](#) substitution rules for “Canon iR-ADV 8285/8295 UFR II” are shown below. Also see [Section L.1.1.1](#), “[org.savapage-finishings-staple](#)” [305]

```
*SPSubst/sheet-collate/collated: collate \
  org.savapage-finishings-staple/3 \
  *True

*SPSubst/sheet-collate/uncollated: group \
  org.savapage-finishings-staple/3 \
  *Group
```



```
*SPSubst/sheet-collate/collated: staple-and-collate \
  org.savapage-finishings-staple/!3 \
  *StapleCollate

*SPSubst/sheet-collate/uncollated: staple-and-group \
  org.savapage-finishings-staple/!3 \
  *StapleGroup
```

K.2.2. Custom PPD Rules

Custom PPD Rules are restricted to specific situations.

K.2.2.1. LandscapeOrientation

Most PPD files contain an attribute called `LandscapeOrientation`, with value `Plus90` (default) or `Minus90`, that describes how landscape oriented pages are rotated to fit the portrait mode "Finished Page" ².

In `Plus90` mode, landscape pages are -90 rotated to fit the "Finished Page", and the user must +90 rotate after printing, to get it into tangible landscape view again. `Minus90` is vice versa. The terms "Minus" and "Plus" are a bit confusing. They do not refer to the rotate direction before printing, but refer to the "manual" user rotation *after* printing, to get the "Finished Page" into landscape view again.

SavaPage creates IPP print jobs that are effectively printed according to Minus90. In this way, `staple-top-right` makes sense for landscape viewed pages.

As the CUPS `number-up` option still behaves according to the original PPD `LandscapeOrientation` attribute, SavaPage corrects with appropriate `number-up-layout` and `orientation-requested` options, to get the overall `Minus90` effect for `number-up` printed sheets that have landscape view.

Because SavaPage assumes that all target proxy printers behave according to the `Plus90` default, any exception must be entered in a PPDE like this:

```
*LandscapeOrientation: Minus90
```

When nevertheless `number-up` printed sheets do not turn out as expected, corrections can be made with `SPRule/number-up`.



Note

SavaPage does **not** correct with `number-up-layout` and `orientation-requested` options when a [org.savapage-finishings-booklet](#) finishing is chosen, since MFP booklet finishers are supposed to apply the correct orientation and layout.

K.2.2.1.1. SPSRule/number-up

This is a fallback rule to make corrections, in the rare case that `LandscapeOrientation`, as discussed in the previous section, does not work out as expected. Each rule is prefixed with `*SPRule/number-up:` and formatted like this:

```
*SPRule/number-up: <mnemonic> \ ❶
  pdf-orientation/<value> \ ❷
  pdf-rotation/<value> \ ❸
  pdf-content-rotation/<value> \ ❹
```

² According to PWG5100.3: a Finished Page is "One side of a sheet in a Finished Document, i.e., one side of a sheet as perceived by a person after any cutting, folding, and/or booklet making" making.

```

user-rotate/<value>          \ ❸
number-up/<value>            \ ❹
*orientation-requested/<value> \ ❺
*number-up-layout/<value>    \ ❻
*org.savapage-landscape      ❼

```

- ❶ The rule prefix and identifying <mnemonic>.
- ❷ The orientation <value> of the first PDF page to be printed: portrait, landscape .
- ❸ The rotation <value> of the first PDF page to be printed: 0, 90, 180 , 270 .
- ❹ The content rotation <value> of the first PDF page to be printed: 0, 90, 180 , 270 . When not specified, value 0 is assumed.
- ❺ The user rotation <value> on the PDF document to be printed: 0, 90. See [Section 3.2.3.2, “Rotation” \[33\]](#). When not specified, value 0 is assumed.
- ❻ The selected number-up <value> : 1, 2, 4, 6, 9.
- ❼ As the previous independent variables describe the situation “as is”, this is the first *dependent* variable telling CUPS the orientation-requested <value> : 4 (landscape), 5 (reverse landscape) , 6 (reverse portrait). Note: a value of - indicates that this CUPS attribute is *not* used.
- ❽ The second *dependent* variable telling CUPS the number-up-layout <value> : btlr, btrl, lrbt, lrtb, rlbt, rltb, tblr, tbrl. Note: a value of - indicates that this CUPS attribute is *not* used.
- ❾ The third optional *dependent* variable is used for the [Document Log](#) only, telling that the resulting n-up layout has logical landscape orientation. When not specified, portrait orientation is assumed.

As an example, some rules are shown below.

```

*SPRule/number-up: portrait-90-0-0-1 \
  pdf-orientation/portrait pdf-rotation/90 user-rotate/0 number-up/1 \
  *orientation-requested/- *number-up-layout/-

*SPRule/number-up: portrait-90-0-0-2-270 \
  pdf-orientation/portrait pdf-rotation/90 user-rotate/0 number-up/2 \
  *orientation-requested/5 *number-up-layout/tbrl

*SPRule/number-up: portrait-90-0-0-4 \
  pdf-orientation/portrait pdf-rotation/90 user-rotate/0 number-up/4 \
  *orientation-requested/- *number-up-layout/tbrl

*SPRule/number-up: portrait-90-0-0-6-270 \
  pdf-orientation/portrait pdf-rotation/90 user-rotate/0 number-up/6 \
  *orientation-requested/5 *number-up-layout/lrtb

*SPRule/number-up: portrait-90-0-0-9 \
  pdf-orientation/portrait pdf-rotation/90 user-rotate/0 number-up/9 \
  *orientation-requested/- *number-up-layout/tbrl

*SPRule/number-up: landscape-0-0-0-2-270 \
  pdf-orientation/landscape pdf-rotation/0 user-rotate/0 number-up/2 \
  *orientation-requested/5 *number-up-layout/tbrl

*SPRule/number-up: landscape-0-0-0-6-270 \
  pdf-orientation/landscape pdf-rotation/0 user-rotate/0 number-up/6 \
  *orientation-requested/5 *number-up-layout/lrtb

*SPRule/number-up: landscape-270-270-0-6-270 \
  pdf-orientation/landscape pdf-rotation/270 pdf-content-rotation/270 \
  user-rotate/0 number-up/6 \
  *orientation-requested/5 *number-up-layout/tbrl

*SPRule/number-up: landscape-270-0-90-2 \
  pdf-orientation/landscape pdf-rotation/270 pdf-content-rotation/0 \
  user-rotate/90 number-up/2 \
  *orientation-requested/- *number-up-layout/tbrl

```

```
*SPRule/number-up: landscape-270-0-90-4 \
  pdf-orientation/landscape pdf-rotation/270 pdf-content-rotation/0 \
  user-rotate/90 number-up/4 \
  *orientation-requested/- *number-up-layout/tbrl *org.savapage-landscape

*SPRule/number-up: landscape-270-0-90-6 \
  pdf-orientation/landscape pdf-rotation/270 pdf-content-rotation/0 \
  user-rotate/90 number-up/6 \
  *orientation-requested/- *number-up-layout/tbrl
```

K.2.2.1.2. Number-up semantics and limitations

- PDF input from **Web Print** can lead to unexpected number-up print results, when the semantic (perceived) orientation of a page, does not match the actual orientation/rotation of the PDF page. For instance, a mismatch occurs when a landscape oriented PDF page, has rotated portrait content. *There is no way SavaPage can identify this situation to make intelligent corrections.*
- An `SPRule/number-up` is applied when its independent variables (orientation, rotation) match the *first* page in the PDF document. When these variables differ for subsequent pages, the PPD has the final say on how page rotation and n-up layout turns out. *A PDF document with different page orientations may produce an unexpected number-up result.*

K.3. IPP Rules

IPP Rules specify relations between IPP attributes.

K.3.1. SPConstraint

An `SPConstraint` rule specifies values of two incompatible IPP attributes, for example two-sided printing on transparency media. The rule is formatted like this:

```
*SPConstraint: <mnemonic> \ ❶
  <attrib>/[!]<value> \ ❷
  <attrib>/[!]<value> ❸
```

- ❶ The rule prefix and identifying `<mnemonic>`.
- ❷ The first IPP attribute/value pair. An optional `!` before a value negates it, and selects all other attribute values.
- ❸ The second IPP attribute/value pair, that is incompatible with the first.

Some examples:

```
*SPConstraint: transparency-two-sided \
  media-type/transparency \
  sides/!one-sided

*SPConstraint: booklet-punch \
  org.savapage-finishings-booklet/!none \
  org.savapage-finishings-punch/!3
```



Note

`SPConstraint` is the IPP counterpart of the `UIConstraints` directive found in PPD files.

“CUPS does not enforce constraints when printing. Constraints must be managed and resolved by the user interface, because there is usually no way to specify preferences or intentions for automatic resolution of constraints by the driver or other filters.” From “CUPS: Common Unix Printing System” by Michael R. Sweet, Sams Publishing, 2002 (page 337).

SPConstraint rules are used by SavaPage to validate user input when specifying a print job.

K.3.1.1. Internal SPConstraint Rules

SavaPage has predefined rule sets. When activated, they are applied globally, independent of any PPDE file.

K.3.1.1.1. Booklet SPConstraint Set

This set is for [org.savapage-finishings-booklet](#) and must be activated by setting configuration property `ipp.ext.constraint.booklet.enable` to Y.

```
*SPConstraint: sp-booklet-number-up-1 \
  org.savapage-finishings-booklet/!none \
  number-up/1

*SPConstraint: sp-booklet-number-up-4 \
  org.savapage-finishings-booklet/!none \
  number-up/4

*SPConstraint: sp-booklet-number-up-6 \
  org.savapage-finishings-booklet/!none \
  number-up/6

*SPConstraint: sp-booklet-number-up-9 \
  org.savapage-finishings-booklet/!none \
  number-up/9

*SPConstraint: sp-booklet-one-sided \
  org.savapage-finishings-booklet/!none \
  sides/one-sided

*SPConstraint: sp-booklet-rotate-180-on \
  org.savapage-finishings-booklet/!none \
  org.savapage.int-page-rotate180/1
```



Important

When Media Cost is specified for Proxy Printers or Job Tickets supporting booklet finishing, activating this set is **crucial** for calculating the right cost for booklet print jobs. See [Section 4.8.2.2, “Printer Costs” \[106\]](#) and [Section K.4.1, “Job Ticket Media Options” \[300\]](#).



Caution

When this set is activated, user choices will be restricted to `number-up/2` and `side/!one-sided`. As this is correct for cost calculation, these options will probably **not** deliver the intended booklet. So, make sure to review the required IPP options for the booklet job, and use [SPExtra](#) rules to customize proxy printer instances where needed.

K.4. Job Ticket Extensions

Job Tickets are created with a dedicated Proxy Printer marked as Job Ticket Printer. As with any printer, a PPD extension file can be linked to this virtual printer.

In the PPD Extension file, special extensions are available to define Job Ticket IPP options, including *Cost Rules*. These IPP options are generic and abstracted from physical printers. When handling the ticket, the operator interprets the option values, selects a suitable printer, assures the right media are present in the target tray, and redirects the job to it.

Job Ticket IPP options can be defined for scope *Media* (single side of a page sheet), *Sheet* (single page sheet), *Copy* (sheet collection of a printed copy) and *Set* (set of collated copies). The syntax is as follows:

```
*SPJobTicket/<scope>: <attr> \ ❶
[*|+]<value> ... ❷
```

- ❶ The *SPJobTicket option with scope *Media* | *Sheet* | *Copy* | *Set*, and attribute name.
- ❷ One or more option values. An * before a value marks it as default. A + prefix signifies an *extended* value, that is available for Job Ticket Operators only, so they can ad-hoc assign this value when editing a job ticket.

Cost Rules are introduced for each option scope in the sections below.



Important

IPP Options defined in the Job Ticket context are *not* mapped to their PPD counterparts. Therefore, their chosen values will *not* be send with the print job.

K.4.1. Job Ticket Media Options

IPP attributes describing media characteristics, like *media-color* and *media-type* are supported. Each option is prefixed with *SPJobTicket/Media:

Some examples:

```
*SPJobTicket/Media: media-color *white int.colored ❶
*SPJobTicket/Media: media-type *paper transparency labels ext.letterhead-1 ❷
```

- ❶ Option to select white and colored. *white* is an IANA media-color and is the default. *int.colored* is an internal IPP value extension, denoting a non-white color. Other values could be blue, red, green, orange, etc.
- ❷ Option to select media types: *ext.letterhead-1* is an external IPP value extension.

Media Cost is charged per media *side*, and specified for a combination of IPP values for media attributes. Each cost rule is prefixed with *SPJobTicket/Media/Cost: and formatted like this:

```
*SPJobTicket/Media/Cost: <cost> <mnemonic> \ ❶
<media*>/[!]<value> ... \ ❷
<attrib>/[!]<value> ... \ ❸
```

- ❶ Decimal point <cost> and identifying <mnemonic>.
- ❷ One or more IPP <media*>/<value> pairs. An optional ! before a value negates it, and selects all other attribute values.
- ❸ Optionally one or more IPP non-media <attrib>/<value> pairs.



Note

The IPP <media*>/<value> pairs referred to in *SPJobTicket/Media/Cost must either be automatically picked up from the PPD, or be defined as **PPD to IPP Mapping** or *SPJobTicket/Media.

Some examples:

```
*SPJobTicket/Media/Cost: 0.0430 white-A4-080-S \ ❶
media-type/ext.paper-80 \
media-color/white \
```

```
media/iso_a4_210x297mm \
sides/one-sided

*SPJobTicket/Media/Cost: 0.0610 color-A4-120-S \ ❷
media-type/ext.paper-120 \
media-color/!white \
media/iso_a4_210x297mm \
sides/one-sided

*SPJobTicket/Media/Cost: 0.0390 white-A4-80-D \ ❸
media-type/ext.paper-80 \
media-color/white \
media/iso_a4_210x297mm \
sides/!one-sided

*SPJobTicket/Media/Cost: 0.0790 letterhead-A4-S \ ❹
media-type/ext.letterhead-1 \
media/iso_a4_210x297mm \
sides/one-sided
```

- ❶ Single-sided A4 print on 80 grams white paper: 0.0430 per side.
- ❷ Single-sided A4 print on 120 grams colored paper: 0.0610 per side.
- ❸ Double-sided A4 print on 80 grams white paper: 0.0390 per side.
- ❹ Single-sided A4 print on letterhead: 0.0790 per side.



Important

When *SPJobTicket/Media/Cost items are present, they acts as constraint. When a user sets print job properties, a cost rule must be present that matches chosen media options. When no rule is found, a warning message is displayed to the user.

The calculated cost of the *first* cost rule, that applies to the Job Ticket option values, is used as media cost.

K.4.2. Job Ticket Sheet Options

Job Ticket Sheet options specify *finishing* actions, performed on a single printed sheet. Each option is prefixed with *SPJobTicket/Sheet:

For example:

```
*SPJobTicket/Sheet: org.savapage-finishings-ext *none laminate ❶
```

- ❶ **org.savapage-finishings-ext** option to select an extra finishing to be performed manually by Job Ticket operator.

Sheet Cost is charged per media *sheet*, and specified for a combination of SPJobTicket/Sheet and other (media*) attribute values. Each cost rule is prefixed with *SPJobTicket/Sheet/Cost: and formatted like this:

```
*SPJobTicket/Sheet/Cost: <cost> <mnemonic> \ ❶
<sheet>/[!]<value> ... \ ❷
<attr>/[!]<value> ... \ ❸
```

- ❶ Decimal point <cost> and identifying <mnemonic>.
- ❷ One or more IPP <sheet>/<value> pairs of type SPJobTicket/Sheet. An optional ! before a value negates it, and selects all other attribute values.
- ❸ Optionally one or more other IPP <attr>/<value> pairs.

For example:

```
*SPJobTicket/Sheet/Cost: 0.5000 laminate-A4 org.savapage-finishings-ext/laminate \
media/iso_a4_210x297mm
*SPJobTicket/Sheet/Cost: 0.7500 laminate-A3 org.savapage-finishings-ext/laminate \
media/iso_a3_297x420mm
```



Important

The calculated cost of *all* sheet cost rules, that apply to the Job Ticket option values, are *accumulated* as cost per sheet.

K.4.3. Job Ticket Copy Options

Job Ticket Copy options specify *finishing* actions, performed on a single printed copy. Each option is prefixed with *SPJobTicket/Copy:

Some examples:

```
*SPJobTicket/Copy: org.savapage-cover-type *no-cover ext.printfront-1 ❶
*SPJobTicket/Copy: org.savapage-finishings-ext *none laminate bind adhesive ext.binder ❷
```

- ❶ `org.savapage-cover-type` option to select a cover type. `ext.printfront-1` is an **External IPP extension**.
- ❷ `org.savapage-finishings-ext` option to select an extra finishing to be performed manually by Job Ticket operator. `ext.binder` is an **External IPP extension**.

Copy Cost is charged per job *copy*, and specified for a combination of SPJobTicket/Copy and other (media*) attribute values. Each cost rule is prefixed with *SPJobTicket/Copy/Cost: and formatted like this:

```
*SPJobTicket/Copy/Cost: <cost> <mnemonic> \ ❶
<copy>/[!]<value> ... \ ❷
<attr>/[!]<value> ... \ ❸
```

- ❶ Decimal point `<cost>` and identifying `<mnemonic>`.
- ❷ One or more IPP `<copy>/<value>` pairs of type SPJobTicket/Copy. An optional ! before a value negates it, and selects all other attribute values.
- ❸ Optionally one or more other IPP `<attr>/<value>` pairs.

For example:

```
*SPJobTicket/Copy/Cost: 0.5000 folder-A4 org.savapage-finishings-ext/ext.folder \
media/iso_a4_210x297mm
```



Important

The calculated cost of *all* copy cost rules, that apply to the Job Ticket option values, are *accumulated* as cost per copy.

K.4.3.1. Using External IPP Extensions

Job Ticket Copy Options are not confined to regular IPP attributes and **Internal IPP Extensions**. By utilizing **External IPP Extensions**, you can fully customize your Job Tickets Copy options and cost rules. For example:

```
*SPJobTicket/Copy: org.savapage.ext-myoption \ ❶
*none ext.choice-1 ext.choice-2
*SPJobTicket/Copy/Cost: 0.1000 myrule-1 \ ❷
```

```
org.savapage.ext-myoption/ext.choice-1 media/iso_a4_210x297mm

*SPJobTicket/Copy/Cost: 0.1500 myrule-2 \
    org.savapage.ext-myoption/ext.choice-2 media/iso_a3_297x420mm
```

- ❶ Definition of External IPP extension `myoption` with two custom choices. The default choice `none` is a reserved internal value, indicating that the option is not selected.
- ❷ Cost rule for `ext.choice-1` and A4 media.
- ❸ Cost rule for `ext.choice-2` and A3 media.

K.4.4. Job Ticket Set Options

Job Ticket Set options specify actions performed on a set of *collated* copies. The term *Set* is defined as “A logical boundary between the delivered media sheets of a printed job” (RFC8011). For example, in a print request for 50 copies of a ten-page document with collated pages, the 50 printed copies of the document constitutes a Set. Each option is prefixed with `*SPJobTicket/Set:`

For example:

```
*SPJobTicket/Set: org.savapage-job-sheets none *job-sheet-start
*SPJobTicket/Set: org.savapage-job-sheets-media *iso_a4_210x297mm
```

- ❶ `org.savapage-job-sheets` option to select a job-sheet.
- ❷ `org.savapage-job-sheets-media` option to select the job-sheet media.

Set Cost is charged per *job*, and specified for a combination of `SPJobTicket/Set` and other (`org.savapage-job-sheets-*`) attribute values. Each cost rule is prefixed with `*SPJobTicket/Set/Cost:` and formatted like this:

```
*SPJobTicket/Set/Cost: <cost> <mnemonic> \
    <set>/[!]<value> ... \
    <attr>/[!]<value> ... \
```

- ❸ Decimal point `<cost>` and identifying `<mnemonic>`.
- ❷ One or more IPP `<set>/<value>` pairs of type `SPJobTicket/Set`. An optional `!` before a value negates it, and selects all other attribute values.
- ❸ Optionally one or more other IPP `<org.savapage-job-sheets-*>/<value>` pairs.

For example:

```
*SPJobTicket/Set/Cost: 0.05 banner-A4 org.savapage-job-sheets/!none \
    org.savapage-job-sheets-media/iso_a4_210x297mm
```



Important

The calculated cost of *all* cost rules, that apply to the Job Ticket option values, are *accumulated* as cost per set.

K.5. Tips and Tricks

K.5.1. Fast Print A4 and Letter to Single Tray

For **Fast Mode** printers with a single tray holding A4 media, PPDE can be used to configure an extra (virtual) IPP `media-source` entry for Letter, that maps to the same A4 tray. The two media sources will be visible in the **Proxy**

Printer Edit dialog, so A4 and Letter media size can be assigned to them, and both A4 and Letter jobs will be Fast Print candidates. With an **SPExtra** rule the Letter media is substituted with A4. And, since `fit-to-page` **print-scaling** is applied for Fast Mode printing by default, the Letter job will correctly scale to A4.

The .ppde of the Fast Mode printer looks like this:

```
#-----
# Two trays mapped to a single media-source holding A4 media.
# In SavaPage Admin Web App: assign A4 to tray-1 and US Letter to tray-2.
# -----
*VENInputSlot media-source
*VENInputSlot *Auto tray-1
*VENInputSlot Auto tray-2

# -----
# When job is US Letter, send as A4. The fit-to-page strategy
# for Fast Print will make Letter scale to A4.
# -----
*SPExtra/media/na_letter_8.5x11in: letter-to-a4 \
    *media/iso_a4_210x297mm
```

K.6. proxy-print.log

The IPP request and response details of every Proxy Print job send to CUPS is logged in the rotating log file:

`/opt/savapage/server/logs/proxy-print.log`

This file is an indispensable resource for debugging PPDE related issues. See `/opt/savapage/server/lib/log4j.properties.template` for more information.

Appendix L. IPP Extensions

Internet Printing Protocol (IPP) attributes and values are registered by IANA. See the [IANA site](https://www.iana.org/assignments/ipp-registrations/ipp-registrations.xhtml)¹ or the [PWG site](http://www.pwg.org/ipp/ipp-registrations.xml)² for a full list.

SavaPage uses two types of extensions:

- *Internal* extensions, which are intrinsic to SavaPage.
- *External* extensions, as defined in implementation specific configuration files.

L.1. Internal IPP Extensions

Internal IPP Extension attributes are *intrinsic* to SavaPage. To distinguish them from IANA registrations, their names have a `org.savapage-` prefix. Attribute value extensions with “type2 keyword” syntax are `int.` prefixed.

Attributes and values are summarized in the sections below. Attribute values are IANA registered, and the semantics can be found in the Reference documents. Attribute value `int.` extensions are described separately.

L.1.1. Internal IPP - PPD Mapping Extensions

These Internal IPP extensions are used to map vendor specific PPD options to an independent common denominator. They are *never* send to CUPS as print job descriptors.

L.1.1.1. org.savapage-finishings-staple

Staple positions are specified with respect to portrait media orientation. See [RFC3806](https://tools.ietf.org/html/rfc3806)³.

Attribute	Value	Name	Syntax	Reference
org.savapage-finishings-staple	3	none	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	20	staple-top-left	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	21	staple-bottom-left	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	22	staple-top-right	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	23	staple-bottom-right	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	24	edge-stitch-left	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	25	edge-stitch-top	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	26	edge-stitch-right	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	27	edge-stitch-left-bottom	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	28	staple-dual-left	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	29	staple-dual-top	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	30	staple-dual-right	1setOf type2 enum	RFC8011
org.savapage-finishings-staple	31	staple-dual-bottom	1setOf type2 enum	RFC8011

¹ <https://www.iana.org/assignments/ipp-registrations/ipp-registrations.xhtml>

² <http://www.pwg.org/ipp/ipp-registrations.xml>

³ <https://tools.ietf.org/html/rfc3806>

Attribute	Value	Name	Syntax	Reference
org.savapage-finishings-staple	32	staple-triple-left	1setOf type2 enum	PWG5100.1
org.savapage-finishings-staple	33	staple-triple-top	1setOf type2 enum	PWG5100.1
org.savapage-finishings-staple	34	staple-triple-right	1setOf type2 enum	PWG5100.1
org.savapage-finishings-staple	35	staple-triple-bottom	1setOf type2 enum	PWG5100.1

Table L.1. Internal IPP Attribute: org.savapage-finishings-staple

L.1.1.2. org.savapage-finishings-punch

Punch positions are specified with respect to portrait media orientation. See [RFC3806](https://tools.ietf.org/html/rfc3806)⁴.

Attribute	Value	Name	Syntax	Reference
org.savapage-finishings-punch	3	none	1setOf type2 enum	RFC8011
org.savapage-finishings-punch	70	punch-top-left	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	71	punch-bottom-left	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	72	punch-top-right	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	73	punch-bottom-right	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	74	punch-dual-left	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	75	punch-dual-top	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	76	punch-dual-right	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	77	punch-dual-bottom	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	78	punch-triple-left	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	79	punch-triple-top	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	80	punch-triple-right	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	81	punch-triple-bottom	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	82	punch-quad-left	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	83	punch-quad-top	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	84	punch-quad-right	1setOf type2 enum	PWG5100.1
org.savapage-finishings-punch	85	punch-quad-bottom	1setOf type2 enum	PWG5100.1

Table L.2. Internal IPP Attribute: org.savapage-finishings-punch

L.1.1.3. org.savapage-finishings-fold

Attribute	Value	Name	Syntax	Reference
org.savapage-finishings-fold	3	none	1setOf type2 enum	RFC8011
org.savapage-finishings-fold	90	fold-accordion	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	91	fold-double-gate	1setOf type2 enum	PWG5100.1

⁴ <https://tools.ietf.org/html/rfc3806>

Attribute	Value	Name	Syntax	Reference
org.savapage-finishings-fold	92	fold-gate	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	93	fold-half	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	94	fold-half-z	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	95	fold-left-gate	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	96	fold-letter	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	97	fold-parallel	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	98	fold-poster	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	99	fold-right-gate	1setOf type2 enum	PWG5100.1
org.savapage-finishings-fold	100	fold-z	1setOf type2 enum	PWG5100.1

Table L.3. Internal IPP Attribute: org.savapage-finishings-fold

L.1.1.4. org.savapage-finishings-booklet

Attribute	Keyword Value	Syntax	Reference
org.savapage-finishings-booklet	none	type2 keyword	RFC8011
org.savapage-finishings-booklet	toleft-totop	type2 keyword	PWG5100.3
org.savapage-finishings-booklet	toright-tobottom	type2 keyword	PWG5100.3

Table L.4. Internal IPP Attribute: org.savapage-finishings-booklet



Important

For the right booklet cost calculation, make sure to activate the [Booklet SPConstraint Set](#).

L.1.1.5. org.savapage-finishings-jog-offset

A finishing that shifts Each Set in the [output-bin](#) from the previous one by a small amount which is device dependent (PWG5100.1).

Attribute	Value	Name	Syntax	Reference
org.savapage-finishings-jog-offset	3	none	1setOf type2 enum	RFC8011
org.savapage-finishings-jog-offset	14	EachSet	1setOf type2 enum	PWG5100.1

Table L.5. Internal IPP Attribute: org.savapage-finishings-jog-offset

L.1.2. Internal IPP Job Ticket Extensions

These Internal IPP extensions are used in Job Ticketing Cost Rules. See [Section K.4, “Job Ticket Extensions”](#) [299].

L.1.2.1. org.savapage-finishings-ext

This attribute represents a finishing action, on a set of pages (a copy), executed externally by an operator, on the printed output.

Attribute	Keyword Value	Syntax	Reference
org.savapage-finishings-ext	none	type2 keyword	RFC8011
org.savapage-finishings-ext	laminate	type2 keyword	PWG5100.1
org.savapage-finishings-ext	bind	type2 keyword	PWG5100.1
org.savapage-finishings-ext	adhesive	type2 keyword	Candidate Standard PWG 5100.1-2014: "sheets are bound using glue or adhesive."

Table L.6. Internal IPP Attribute: org.savapage-finishings-ext

L.1.2.2. org.savapage-cover-type



Caution

This attribute is intended for externally printed Job Tickets only.

The same PDF document must be printed twice with different page ranges:

1. The first page(s) are printed on a single cover page. The cost for this action is calculated according to the [Job Ticket Copy Cost Rules](#).
2. The rest of the pages are printed on selected media. The cost for this action is calculated according to the [Job Ticket Media Cost Rules](#).

Attribute	Keyword Value	Syntax	Reference
org.savapage-cover-type	no-cover	type2 keyword	PWG5100.3
org.savapage-cover-type	int.printfront	type2 keyword	SavaPage Extension: a single-sided printed cover page.
org.savapage-cover-type	int.printboth	type2 keyword	SavaPage Extension: a double-sided printed cover page.

Table L.7. Internal IPP Attribute: org.savapage-cover-type

L.1.2.3. org.savapage-job-sheets

This attribute determines which Job start/end sheet(s), if any, must be printed with a Job. Contrary to the IPP `job-sheets` definition, where sheets are part of the job, `org.savapage-job-sheets` are printed as a *separate* job. In this way sheets can be printed from a different media-source with distinctive [org.savapage-job-sheets-media](#) and `media-color`.

A job sheet is a single monochrome page with job detail information.

Attribute	Keyword Value	Syntax	Reference
org.savapage-job-sheets	none	type2 keyword	RFC8011
org.savapage-job-sheets	job-start-sheet	type2 keyword	PWG5100.3: "A job sheet MUST be printed to indicate the start of the job".
org.savapage-job-sheets	job-end-sheet	type2 keyword	PWG5100.3: "A job sheet MUST be printed to indicate the end of the job".

Table L.8. Internal IPP Attribute: org.savapage-job-sheets

L.1.2.4. org.savapage-job-sheets-media

The IPP media of [org.savapage-job-sheets](#).

L.1.3. Internal IPP Marker

These Internal IPP extensions are used to add a special marker to a print job. They are *never* send to CUPS as such, but translated by SavaPage to "real" IPP or CUPS options. IPP marker names have a `org.savapage.int-` prefix.

L.1.3.1. org.savapage.int-page-rotate180

SavaPage creates IPP print jobs that are printed according to [LandscapeOrientation](#) Minus90. This is convenient most of the time. For example, when a landscape oriented PDF is 4-up printed, and `staple-top-right` and `punch-dual-right` are available as finishing. But, what if `staple-top-left` and `punch-dual-left` are the only options available? In that case a 180 degrees rotation of the "Finished-Page" is needed. This is where the `org.savapage.int-page-rotate180` boolean option comes in. This marker is added to each proxy printer as option to make the printed result "Rotate by 180 degrees", and translated by SavaPage to the proper CUPS `orientation-requested` and `number-up-layout` values.

Attribute	Keyword Value	Syntax	Description
<code>org.savapage.int-page-rotate180</code>	0	boolean	Do not rotate.
<code>org.savapage.int-page-rotate180</code>	1	boolean	Rotate by 180 degrees.

Table L.9. Internal IPP Marker Attribute: `org.savapage.int-page-rotate180`



Note

`org.savapage.int-page-rotate180` can be used in [SPConstraint](#) rules.

L.2. External IPP Extensions

External IPP Extension attributes as defined in implementation specific configuration files. To distinguish them from IANA registrations, their names have a `org.savapage.ext-` prefix.



Important

External IPP attributes can be used as IPP Job Ticketing Extension only.

The syntax is as follows:

- External *value* extensions with "type2 keyword" syntax must be `ext.` prefixed.
- Value `none` is a reserved internal value, indicating that the attribute option is not selected.
- External attribute values can be added to an [Internal IPP Job Ticket Extension](#).
 - For example: to charge cost for a finishing to wrap a printed copy into a folder, a `ext.folder` value could be added to [org.savapage-finishings-ext](#).

External IPP Extensions open the way to full customization of Job Ticketing options. Any option with any number of choices can be configured in a PPDE file as `*SPJobTicket/Copy finishing`, and thereupon be used in any cost rule. See [Section K.4.3.1, "Using External IPP Extensions" \[302\]](#) and [Section L.3, "IPP Localization" \[310\]](#).

L.3. IPP Localization

User interface localization of regular IPP attributes and **Internal IPP Extensions** are part of SavaPage.

Localization of **External IPP Extensions** must be supplied by XML files in the `/opt/savapage/server/custom/cups/i18n` directory. The default `ipp-i18n.xml` file is reserved for the English locale. Variants are created by appending the language locale to the base file name. For example: `ipp-i18n_de.xml` is the German version. An annotated `ipp-i18n.xml.template` file is available in the target directory.



Note

`ipp-i18n*.xml` files can also be used to override IPP localization that is part of SavaPage.



Warning

After creating or updating any of the `ipp-i18n*.xml` files you might need to restart the server to see the effect.

Appendix M. SavaPage Plug-ins



A *plug-in* (aka “extension”) is a software component that adds a specific feature to SavaPage. Plug-ins have a well-defined interface so partner developers can easily create isolated components that extend the application with new features. Extension interfaces are defined in the [savapage-ext](#)¹ project.

A plug-in is installed by copying its property file in `/opt/savapage/server/ext` and its jar files in the `/opt/savapage/server/ext/lib` directory. For example, the **Mollie Payment Plug-in** is installed with these two files:

```
/opt/savapage/server/ext/savapage-ext-mollie.properties
/opt/savapage/server/ext/lib/savapage-ext-mollie-<version>.jar
```



Important

Since property files contain sensitive data make sure they are protected by executing commands like:

```
sudo chown savapage:savapage savapage-ext-mollie.properties
sudo chmod 600 savapage-ext-mollie.properties
```

M.1. Web API Callback Plug-in

The Web API Callback method is used by many third-party providers who offer their services via HTTP. SavaPage supports this method with the `/callback` URL *path*. However, URL *protocol* and *authority* for the callback needs to be configured. Configuration is done by using the **Config Editor** of the Admin Web App. The following configuration property is available:

Configuration property	Description
<code>ext.webapi.callback.url-base</code>	<p>The publicly accessible base URL, i.e. <code>protocol://authority</code> <i>without the path</i>, of the Web API callback interface (no trailing slash).</p> <p>Since SavaPage is typically implemented as an intranet application to be accessed with a local URL, take care to configure proper port forwarding of the public base URL to the local SavaPage server host name or IP address in your router or firewall .</p>

Table M.1. Web API Callback Configuration Property

M.1.1. Payment Gateway Plug-in

The Payment Gateway Plug-in is based on the Web API Callback Plug-in and uses the `/callback/payment` URL *path*. The following configuration property can be set by using the **Config Editor** of the Admin Web App.

Configuration property	Description
<code>ext.webapi.redirect.url-webapp-user</code>	The User Web App URL used by the Web API to redirect to after a <i>remote</i> Web App dialog is finished.

¹ <https://gitlab.com/savapage/savapage-ext>

Configuration property	Description
	Configuration is optional. SavaPage uses the local URL from which the remote excursion started as default.

Table M.2. Payment Gateway Configuration Property

Payment Gateway events are persisted in the rotating log file:

```
/opt/savapage/server/logs/paymentgateway.log
```

This file has a tab separated value (TSV) format for easy import and manipulation into spreadsheet programs.

M.1.1.1. Generic Payment Plug-in

A Generic Payment Plug-in implements several payment methods behind a single Web API. Only one generic plug-in can be active. See [Section 3.10.3, “Financial” \[68\]](#) and [Section 3.10.6, “Transfer Money” \[69\]](#).

M.1.1.1.1. Mollie Payment Plug-in

[Mollie](#)² is a Dutch payment provider that offers a single Web API for the following payment methods:

- Creditcard
- PayPal
- Bitcoin
- paysafecard
- SOFORT Banking (Europe)
- SEPA bank transfers (Europe)
- Bancontact/Mister Cash (Belgium)
- Belfius Direct Net (Belgium)
- IDEAL (Netherlands).

Mollie supports EUR payments only.

See the README.md of the [savapage-ext-mollie](#)³ project for more information.



Note

The [Mollie Payment Plug-in](#) is shipped with the SavaPage install binary.

M.1.1.1.2. Generic Payment Pitfalls

Callback messages for generic payments return the identity of the user that started the payment in the [Transfer Money](#) dialog. So, when a payment is acknowledged, we can easily add the amount paid to the user's balance. In some very unlikely cases a user might not be found. For example, when a database export is restored or a user was deleted, all in the short lifecycle of a payment transaction.

In the rare case a user is not found, a warning message containing the user and transaction identification are written to the [Application Log](#). With this information the user balance can be updated manually, after the user has been added again, either in the [Admin WebApp](#) or with a [Server Command](#).

² <https://www.mollie.com/en/>

³ <https://gitlab.com/savapage-ext/savapage-ext-mollie>

M.1.1.2. Bitcoin Payment Plug-in

The Bitcoin Payment Plug-in supports native Bitcoin payments with the advantage of a low native Bitcoin transaction fee. Only one Bitcoin plug-in can be active. See [Section 3.10.3, “Financial” \[68\]](#) and [Section 3.10.7, “Send Bitcoins” \[69\]](#).

As a privacy and security measure a new Bitcoin address is generated for each payment⁴. Generated addresses are held in a Bitcoin *wallet*. The location and access credentials of the wallet are to be specified in the plug-in property file.



Note

When a Bitcoin payment method is active in an enabled [Generic Payment Plug-in](#), it is deactivated in favour of an enabled Bitcoin Payment Plug-in.

M.1.1.2.1. Blockchain.info Payment Plug-in

With the Blockchain.info plug-in users can send Bitcoin payments to a Blockchain.info *web-based* wallet. See the `README.md` of the [savapage-ext-blockchain-info](#)⁵ project for more information.



Note

The [Blockchain.info Payment Plug-in](#) is shipped with the SavaPage install binary.

M.1.1.2.2. Bitcoin Payment Pitfalls

Because of the anonymous nature of Bitcoin payments, a callback message with a payment confirmation only contains the Bitcoin address and transaction hash as identification.

Fortunately, we can trace the identity of the user who made the payment, either by the one-time Bitcoin address, that we generated and reserved for the user at the start of the [Send Bitcoins](#) dialog, or by the Bitcoin transaction hash, that we linked to a user payment transaction at the callback of the first confirmed payment.

When a user can *not* be traced, the payment confirmation is ignored. This can happen when a database export is restored and either the user, the reserved Bitcoin address or transaction hash is missing from the database. This case becomes more unlikely as the number of confirmations after which the payment is trusted is set lower, causing a shorter latency of a trusted payment confirmation.

When a payment confirmation arrives for a Bitcoin address for which a user payment transaction link is present with a *different* transaction hash, it is ignored. This can happen when:

- A user, against advice, reused the generated Bitcoin address, as offered in the [Send Bitcoins](#) dialog, to make an extra payment.
- A payment *from* the Bitcoin Wallet was executed which lead to a transaction with a positive satoshi remainder.

When a payment confirmation is ignored, a warning message with the Bitcoin address and transaction hash is written to the [Application Log](#). This information can be used to query the transaction history in the Bitcoin Wallet. Since the Bitcoin address is tagged in the Wallet with the user id, any transaction with a received amount can be used to trace the user. In case an extra user payment is identified, the user balance can be updated manually, either in the [Admin WebApp](#) or with a [Server Command](#).

⁴ As [Satoshi Nakamoto](#) [https://en.wikipedia.org/wiki/Satoshi_Nakamoto], the elusive creator of Bitcoin, states in his [Bitcoin whitepaper](#) [https://en.bitcoin.it/wiki/Bitcoin_whitepaper] : “... a new key pair should be used for each transaction to keep them from being linked to a common owner”. Also see [this article](#) [https://en.bitcoin.it/wiki/Address_reuse] on address reuse.

⁵ <https://gitlab.com/savapage-ext/savapage-ext-blockchain-info>

M.2. OAuth Client Plug-in

With [OAuth](#)⁶ Client Plug-ins [Login Alternatives](#) can be activated. Currently the following providers are supported:

- [Google](#)⁷
- [Smartschool](#)⁸

See the `README.md` of the [savapage-ext-oauth](#)⁹ project for more information.



Note

A special URL path is available to use OAuth as SSO for User Web App. This URL can be presented on the site of an OAuth provider, to seamlessly link to SavaPage. See [Appendix E, URL Cheat Sheet \[278\]](#).

⁶ <https://en.wikipedia.org/wiki/OAuth>

⁷ <https://www.google.com/>

⁸ <https://www.smartschool.be/>

⁹ <https://gitlab.com/savapage-ext/savapage-ext-oauth.git>

Appendix N. PaperCut Integration



PaperCut is a popular print and copy management software product developed by [PaperCut Software](https://www.papercut.com/)¹ based in Melbourne, Australia.

Some functions that are not present in PaperCut can be implemented with SavaPage as pre-processor and integrator.



Note

See [Section 4.10.5, “PaperCut Integration”](#) [132] on how to set the PaperCut Connectivity options.

N.1. Delegated Print to PaperCut

Delegated Print is integrated with PaperCut when the following conditions are met:

- PaperCut Integration is enabled: see [Section 4.10.5, “PaperCut Integration”](#) [132]
- Delegated Print and Delegated Print integration with PaperCut are enabled: see [Section 4.10.10.2, “Proxy Print Delegation”](#) [143].

The Proxy Printer must meet the following requirements:

- The Proxy Printer is managed by PaperCut.
- The Proxy Printer is configured as non-secure. See [Section 4.10.10, “Proxy Print”](#) [141]. If secure printing is required it must be configured in PaperCut and not in SavaPage.



Warning

A Delegated Print job is denied when the delegate or any of the delegators does not exist in PaperCut. This limitation can easily be solved when both systems synchronize from the same user source. See [Section 4.10.1, “User Source”](#) [120].

Tips for further reading:

- [Section A.2.3, “Delegated Print - PaperCut Scenario”](#) [249].
- [Section A.2.2, “Delegated Print - Job Ticket - PaperCut - Scenario”](#) [248].

N.1.1. PaperCut Configuration

N.1.1.1. Step 1 - Create shared account

Create a shared parent account called SavaPage. This top-level account *must* be present, since several sub-accounts will be lazy created by SavaPage.

In addition, any printer used for Delegated Print must be configured to charge to this account. See [Section N.1.1.3, “Step 3 - Configure Printers”](#) [316].



Note

The PaperCut shared account name is known by the SavaPage configuration key:

¹ <https://www.papercut.com/>

```
proxy-print.delegate.papercut.account.shared.parent
```

The value defaults to SavaPage. See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change this value.

N.1.1.2. Step 2 - Enable Multiple Personal Accounts

Enable the PaperCut “Multiple Personal Accounts” option and add the personal account SavaPage. This account *must* be present for it is used by SavaPage to charge printing costs to individual persons.



Important

The PaperCut personal account name is known by the SavaPage configuration key:

```
proxy-print.delegate.papercut.account.personal
```

The value defaults to SavaPage.

The account type for this account as determined by PaperCut in its own configuration key `multi-personal-accounts.definitions` (with values like `USER-001`, `USER-002`) can be set with this key:

```
proxy-print.delegate.papercut.account.personal-type
```

When a value is specified it is used to filter personal account transactions in JDBC queries (CSV downloads) for the Delegated Print context.

See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change these values.

N.1.1.3. Step 3 - Configure Printers

Take a moment to consider how you want the PaperCut printers that are used for SavaPage Delegated Print to act. A likely scenario is that you want these printers to be virtual hold/release queues so users can enjoy follow-me printing, and release print jobs at a series of physical printers. Or, may be you want these jobs to be released by administrators only. Consult the PaperCut User Manual on how implement the desired scenario.

There is one crucial printer configuration property though that must be addressed. Make sure that **Override user-level settings** is set, and activate **Do not show account popups and allocate jobs to: a single shared account**. Use the Shared Account SavaPage (as created in Step 1) and select the **Charge shared account** option.

Now, when a Delegated Print job is successfully printed by PaperCut, the cost will be automatically charged to the shared SavaPage account.

N.1.2. PaperCut Delegated Print Processing

SavaPage monitors the print job status in PaperCut and, when printing is successful, charges the costs, as reported by PaperCut, to the proper PaperCut accounts, as explained in the section below.



Important

In addition to PaperCut Account processing, account transactions are still added to SavaPage as explained in [Section 3.4.7, “Delegated Print Edit” \[46\]](#). However, the printing costs reported by PaperCut overrule any costs defined in SavaPage for regular printing. *For Job Tickets, SavaPage cost is leading.*

N.1.3. PaperCut Delegated Print Accounting

SavaPage uses the PaperCut cost total of the Delegated Print job to add extra PaperCut account transactions. The *comment* of each account transaction holds a combination of pipe-separated (|) job parameter fields, with the following meaning:

- *delegate* : the user (delegate) who printed the job on behalf of the target users (delegators).
- *class* : the “class” a target user belongs to. A value of “-” (minus) means the user does not belong to a class. A class can be of type:
 - *shared* : a SavaPage **Shared Account**.
 - *group* : a SavaPage **Group Account**.
- *copies* : the number of document copies printed (a *negative* number for a **Refund**).
- *pages* : number of pages within the document.
- *size* : the page size of the document (A4, A3).
- *duplex* : D for duplex, S for simplex.
- *color* : C for color, G for grayscale.
- *id* : the CUPS job ID.
- *document* : the document name.
- *comment* : any comment entered by the delegate.

As a rule SavaPage charges target users individually for print copies.

As an extra, solely for reporting purposes, dedicated PaperCut shared accounts are used to accumulate cost and job information globally (the SavaPage\Jobs account) and per User Class.



Important

No transaction appears in any PaperCut account when the cost of a print job is *zero*. For transactions to appear you need to specify page cost at the PaperCut printer configuration.

N.1.3.1. PaperCut User Accounting

SavaPage proportionally splits the cost total of the printed Delegated Print job over individual users (delegators), directly or indirectly by group membership. Costs are charged to their *personal* user account named SavaPage. The comment format of the transaction is:

group | delegate | copies | pages | size | duplex | color | id | document | comment

Copies for users not belonging to a group have dummy group “-” (minus).

N.1.3.2. PaperCut User Class Accounting

SavaPage proportionally splits the cost total of the printed Delegated Print job over “classes”. Class cost is proportional to the sum of the print copies for users belonging to the class. This cost is charged to a shared child account of the SavaPage parent account, with format:

savapage.[class].[name]

The [class] placeholder stands for the class *type* and can have value *group* or *shared*. The [name] is placeholder for the class *name*. The *name* of a shared child account is prefixed with the name of its parent, separated by a dot character (as parent.child).

PaperCut shared child accounts are *ad-hoc* created by SavaPage on first use.

The comment format of the transaction is:

```
delegate | copies | pages | size | duplex | color | id | document | comment
```

N.1.3.3. PaperCut Job Accounting

As an extra, for each Delegated Print job, SavaPage adds a transaction to the shared child account “Jobs” of the SavaPage parent account. The comment format of the transaction is:

```
delegate | copies | pages | size | duplex | color | id | delegate@class-1 |  
copies-1 | ... | delegate@class-n | copies-n | document | comment
```

The `delegate@class | copies` field group shows the printed copies per class and is repeated for each class that was printed for. Copies for users not belonging to a class are accumulated in dummy class “-” (minus).



Note

The PaperCut shared child account name is known by the SavaPage configuration key:

```
proxy-print.delegate.papercut.account.shared.child.jobs
```

The value defaults to Jobs. See [Section 4.10.14.10, “Config Editor” \[155\]](#) on how to change this value.

N.1.4. PaperCut Queries and Reports

Use the on-line queries and run the reports in the PaperCut Admin Web App to answer questions about Delegated Print jobs and transactions from the following perspectives.

N.1.4.1. User Prints

SavaPage documents printed for a user.

Users → User List → User Details → Transactions gives a quick on-line view of personal account transactions. Sort and filter to find the SavaPage transactions.

Use Reports → Transaction Reports → Transaction logs to generate a SavaPage transaction report for a user or user group over a period of time. The report shows the individual transaction details and the total amount charged.



Important

PaperCut does *not* have a Transaction Report to accumulate transaction totals per user over a period of time. Please use the [Delegator Invoicing from PaperCut](#) export function in SavaPage for that purpose.

N.1.4.2. Class Prints

SavaPage documents printed for a class.

Select Accounts → Shared Account List → Account Details → Transaction for a SavaPage class account to get a quick on-line view of account transactions. Sort and filter to find transactions.

The Reports → Shared Account Reports section contains many reports that summarize printing activity charged to shared accounts. Select the SavaPage class account to get a transaction summary report for a period of time.

N.1.4.3. Delegate Prints

SavaPage documents printed by a delegate.

Users → User List → User Details → Job Log gives a quick on-line view of the documents printed by a delegate user. Sort and filter to find the print jobs charged to the shared SavaPage account.

N.1.4.4. Delegate Class Prints

SavaPage documents printed by a delegate for a class.

Select Accounts → Shared Account List → Account Details → Transaction for the SavaPage Jobs account to get a quick on-line view of account transactions. Since the comment contains formatted information about classes, you can select transactions with the “Comment containing” filter.

Likewise you can use Reports → Transaction Reports → Transaction logs to generate a SavaPage\Jobs transaction report for over a period of time. The report shows the individual transaction details and the total amount charged.

N.1.5. Integration Pitfalls

The state of the two systems involved in the Delegated Print chain (SavaPage, PaperCut) is tightly coupled. Restoring an earlier backup of either system can break the work-flow for pending jobs and lead to unwanted results. For instance:

- *When a backup of SavaPage is restored*, it will not handle jobs that were submitted to PaperCut after the backup point. In these cases SavaPage will show a print job status that does not reflect the real situation. On the other side, jobs that were already fully processed, might be re-processed by SavaPage, leading to extra charges on the shared PaperCut accounts.
- *When a backup of PaperCut is restored*, SavaPage will not find PaperCut print status information for *pending* jobs that were submitted to PaperCut after the backup point. In these cases SavaPage will show an *error* print job status, when in real the job is *completed* or *cancelled*.

To avoid integration problems, review your backup and restore strategy carefully.

Make sure you create backups of both SavaPage and PaperCut at the same point in time. Also, be sure that at the time of backup all pending Delegated Print jobs are fully processed. When you need to restore, use backups of the same snapshot time, first restore PaperCut and then SavaPage.

N.2. PaperCut Integration Limitations



Caution

The PaperCut print status of “normal” non-delegated jobs proxy printed to a PaperCut managed queue is **not** monitored.

Appendix O. Job Scheduling

SavaPage background jobs are scheduled with **Cron Trigger Format** expressions:

- External User Synchronization
- Database Backup

O.1. Cron Trigger Format

Cron Trigger Format is similar to GNU/Linux `cron` scheduling expressions. An extensive tutorial can be found [here](http://www.quartz-scheduler.org/documentation/quartz-2.x/tutorials/crontrigger.html)¹, and a simplified introduction is presented in this section.

A Cron Trigger is an expression comprised of 6 mandatory fields separated by white space.

Field Name	Range	Special	Remark
Seconds	0-59	, - *	Not relevant in our case: can be zero.
Minutes	0-59	, - *	
Hours	0-23	, - *	
Day of month	1-31	, - * ?	
Month	1-12	, - *	Range is identical to JAN-DEC.
Day of week	1-7	, - * ?	Range is identical to SUN-SAT. Note: GNU/Linux <code>cron</code> uses range 0-6.

Table O.1. Cron Trigger Format - simplified

Special character values are:

- , : used to specify *additional values*. For example, “2,4,6” in the day-of-week field means the days Monday, Wednesday, and Friday.
- - : used to specify *ranges*. For example, “2-6” in the day-of-week field means the days Monday to Friday.
- * : used to select *all values* within a field. For example: * in the Hours field means “every hour”.
- ? : used to select *no specific value*. This is useful when you need to specify something in one of the two fields in which the character is allowed, but not the other. For example, if I want a trigger to fire on a particular day of the week (say, Sunday), but don’t care what day of the month that happens to be, you would put 1 in the day-of-week field, and ? in the day-of-month field.

¹ <http://www.quartz-scheduler.org/documentation/quartz-2.x/tutorials/crontrigger.html>

Appendix P. GNU Affero General Public License (AGPL)



GNU AFFERO GENERAL PUBLIC LICENSE Version 3, 19 November 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU Affero General Public License is a free, copyleft license for software and other kinds of works, specifically designed to ensure cooperation with the community in the case of network server software.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, our General Public Licenses are intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

Developers that use our General Public Licenses protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License which gives you legal permission to copy, distribute and/or modify the software.

A secondary benefit of defending all users' freedom is that improvements made in alternate versions of the program, if they receive widespread use, become available for other developers to incorporate. Many developers of free software are heartened and encouraged by the resulting cooperation. However, in the case of software used on network servers, this result may fail to come about. The GNU General Public License permits making a modified version and letting the public access it on a server without ever releasing its source code to the public.

The GNU Affero General Public License is designed specifically to ensure that, in such cases, the modified source code becomes available to the community. It requires the operator of a network server to provide the source code of the modified version running there to the users of that server. Therefore, public use of a modified version, on a publicly accessible server, gives the public access to the source code of the modified version.

An older license, called the Affero General Public License and published by Affero, was designed to accomplish similar goals. This is a different license, not a version of the Affero GPL, but Affero has released a new version of the Affero GPL which permits relicensing under this license.

GNU Affero General Public License (AGPL)

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU Affero General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all

the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code;

GNU Affero General
Public License (AGPL)

keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting

GNU Affero General Public License (AGPL)

any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Remote Network Interaction; Use with the GNU General Public License.

Notwithstanding any other provision of this License, if you modify the Program, your modified version must prominently offer all users interacting with it remotely through a computer network (if your version supports such interaction) an opportunity to receive the Corresponding Source of your version by providing access to the Corresponding Source from a network server at no charge, through some standard or customary means of facilitating copying of software. This Corresponding Source shall include the Corresponding Source for any work covered by version 3 of the GNU General Public License that is incorporated pursuant to the following paragraph.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the work with which it is combined will remain governed by version 3 of the GNU General Public License.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU Affero General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU Affero General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU Affero General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU Affero General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY

GNU Affero General
Public License (AGPL)

OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS