

IBM Security AppScan Standard  
Version 9.0.1

*User Guide*





---

# Contents

<b>Chapter 1. Introduction . . . . .</b>	<b>1</b>
Product overview . . . . .	1
What's new . . . . .	1
Contact and support information . . . . .	3
<b>Chapter 2. Installing . . . . .</b>	<b>5</b>
System requirements . . . . .	5
Flash Player upgrade . . . . .	7
Flash Player configuration . . . . .	7
Install . . . . .	7
Silent install . . . . .	8
Uninstall . . . . .	9
Legacy scans . . . . .	9
License . . . . .	10
Load a floating or token license . . . . .	11
Load a node-locked license . . . . .	12
Updates . . . . .	12
Temp file location . . . . .	13
<b>Chapter 3. Getting started . . . . .</b>	<b>15</b>
How an automatic scan works . . . . .	15
Scanning web services . . . . .	16
Basic workflow . . . . .	17
Workflow description . . . . .	17
Tour of the main window . . . . .	19
View selector . . . . .	19
Application Tree . . . . .	20
Result List . . . . .	22
Detail Pane . . . . .	23
Scan panels . . . . .	23
Status bar . . . . .	23
Tutorial . . . . .	24
Step 1: Configuring the scan . . . . .	24
Step 2: Running the scan . . . . .	25
Step 3: Reviewing Scan Results . . . . .	26
Step 4: Communicating results . . . . .	27
Sample scans . . . . .	27
<b>Chapter 4. Configuring . . . . .</b>	<b>29</b>
Scan configuration wizard . . . . .	29
Launching the Scan Configuration Wizard . . . . .	30
Wizard for exploring with AppScan . . . . .	31
Wizard for exploring with an external device or client . . . . .	35
Wizard for exploring with GSC . . . . .	39
Scan configuration dialog box . . . . .	42
URL and Servers view . . . . .	43
Login Management view . . . . .	46
Environment Definition view . . . . .	56
Exclude Paths and Files view . . . . .	56
Explore Options view . . . . .	62
Parameters and Cookies view . . . . .	65
Automatic Form Fill view . . . . .	75
Error Pages view . . . . .	77
Multi-Step Operations view . . . . .	79
Content-Based Results view . . . . .	82
Glass Box view . . . . .	85
Communication and Proxy view . . . . .	86
HTTP Authentication view . . . . .	86
Test Policy view . . . . .	87
Test Options view . . . . .	92
Privilege Escalation view . . . . .	95
Malware view . . . . .	96
Scan Expert view . . . . .	96
Result Expert view . . . . .	98
Advanced Configuration view . . . . .	98
Scan templates . . . . .	111
Predefined templates . . . . .	111
User-defined templates . . . . .	113
Loading scan templates . . . . .	113
Editing Scan Templates . . . . .	113
Changing the configuration during a scan . . . . .	114
<b>Chapter 5. Manual Exploring . . . . .</b>	<b>115</b>
Using AppScan . . . . .	115
Recording a Manual Explore . . . . .	116
Exporting Manual Explore data . . . . .	118
Importing Manual Explore data . . . . .	119
Using AppScan as a proxy server . . . . .	120
Using AppScan as recording proxy . . . . .	121
Exploring with the External Traffic Recorder . . . . .	121
External Login Recorder . . . . .	123
External Traffic Recorder . . . . .	123
Using GSC . . . . .	124
Exploring with GSC . . . . .	125
Scanning a site that includes a SOAP web service as part of the site . . . . .	126
<b>Chapter 6. Scanning . . . . .</b>	<b>127</b>
Starting scans . . . . .	127
Starting scans from the Scan Configuration Wizard . . . . .	127
Starting scans from the Scan menu or the toolbar . . . . .	127
Starting scans from the Welcome dialog box . . . . .	128
Starting scans from the New Scan dialog box . . . . .	128
Scan progress . . . . .	129
Pausing and continuing scans . . . . .	129
Scans stopped due to connectivity issues . . . . .	129
Scans stopped due to application issues . . . . .	130
Saving and loading scans . . . . .	131
Saving scans . . . . .	131
Automatic scan save . . . . .	131
Loading saved scans . . . . .	132
Legacy scan files . . . . .	132
Automatic scan . . . . .	133
Automatic multiphase scanning . . . . .	133
Scan Expert . . . . .	134
Scan Expert recommendations . . . . .	134
Glass box scanning . . . . .	135

Overview . . . . .	135
For Java platforms . . . . .	136
For .NET platforms . . . . .	154
Partial scans . . . . .	159
Explore Only . . . . .	159
Test Only . . . . .	160
Re-Scan . . . . .	160
Scan Multi-Step Operations Only . . . . .	160
Changing the configuration during a scan . . . . .	161
Result Expert . . . . .	161
Exporting scan results . . . . .	161
Generating scan result DB and XML files . . . . .	162
Firebird database structure . . . . .	162
<b>Chapter 7. Results: Application Data</b>	<b>167</b>
Application Data: Application Tree . . . . .	167
Application Data: Result List . . . . .	167
Requests . . . . .	168
Parameters . . . . .	168
Cookies . . . . .	169
Failed Requests . . . . .	170
Filtered URLs . . . . .	170
User Interaction Needed . . . . .	171
Comments . . . . .	171
JavaScript . . . . .	172
Application Data: Detail Pane . . . . .	172
<b>Chapter 8. Results: Security Issues</b>	<b>173</b>
Security Issues: Application Tree . . . . .	173
Exclude URL from scan . . . . .	173
Security Issues: Result List . . . . .	174
Severity levels . . . . .	174
Issue state: Open or Noise . . . . .	175
Resending tests . . . . .	176
Right-click menu . . . . .	177
Filtering Security Issues in Result List . . . . .	177
Sorting the Result List . . . . .	177
Security Issues: Detail Pane . . . . .	178
Issue Information tab . . . . .	178
Advisory tab . . . . .	183
Fix Recommendations tab . . . . .	185
Request/Response tab . . . . .	185
Report false positive test results . . . . .	187
Manual tests . . . . .	189
Non-vulnerable variants . . . . .	191
Saving all non-vulnerable variants . . . . .	191
Defining variants as non-vulnerable . . . . .	191
Non-Vulnerable Variants List . . . . .	191
Deleting variants . . . . .	192
<b>Chapter 9. Results: Remediation</b>	<b>193</b>
<b>Tasks</b> . . . . .	<b>193</b>
Remediation Tasks: Application Tree . . . . .	193
Remediation Tasks: Result List . . . . .	193
Searching Remediation Tasks in Result List . . . . .	194
Sorting Remediation Tasks . . . . .	195
Manipulating priority levels . . . . .	195
Deleting Remediation Tasks from the Result List . . . . .	195
Remediation Tasks: Detail Pane . . . . .	196
<b>Chapter 10. Reports</b> . . . . .	<b>197</b>
Report overview . . . . .	197
Configuring report layout . . . . .	197
Viewing and saving reports . . . . .	198
Creating partial reports . . . . .	198
Creating user-defined report templates . . . . .	199
Earlier versions of report templates . . . . .	203
Security reports . . . . .	204
Limiting the size of Security reports . . . . .	206
Industry standard reports . . . . .	206
Regulatory compliance reports . . . . .	208
Delta analysis reports . . . . .	209
Template-based reports . . . . .	211
Creating a custom report template . . . . .	212
Importing a custom template . . . . .	216
<b>Chapter 11. Tools</b> . . . . .	<b>217</b>
Options dialog box . . . . .	217
Scan Options tab . . . . .	217
Preferences tab . . . . .	218
Recording Proxy tab . . . . .	219
General tab . . . . .	221
Advanced tab . . . . .	221
Scan Scheduler . . . . .	222
Schedule a new scan . . . . .	222
Edit scheduled scan configuration . . . . .	223
Delete a scheduled scan . . . . .	223
Schedule a Test stage only . . . . .	223
Schedule a scan in installments . . . . .	224
Scheduled task command line parameters . . . . .	225
User-defined tests . . . . .	226
User-Defined Test Wizard . . . . .	226
Defining infrastructure tests . . . . .	228
Defining parameter modification tests . . . . .	229
Defining parameter addition tests . . . . .	231
Defining pattern search tests . . . . .	232
Creating your own advisory . . . . .	233
Completing the user-defined test wizard . . . . .	235
Managing user-defined tests . . . . .	235
PowerTools . . . . .	236
Authentication Tester . . . . .	236
Connection Test . . . . .	243
EncodeDecode . . . . .	244
Expression Test . . . . .	245
HTTP Request Editor . . . . .	247
Token Analyzer . . . . .	251
Generic Service Client . . . . .	265
Customizing the Tools menu . . . . .	266
Adjust the order of the PowerTools . . . . .	266
Add programs to the Tools menu . . . . .	266
Extensions . . . . .	267
Extension Manager . . . . .	267
Pyscan . . . . .	268
Explore Optimization module . . . . .	269
Logs . . . . .	273
Scan Log . . . . .	273
AppScan Log . . . . .	274
Update Log . . . . .	274
Traffic Log . . . . .	275
Searching Results . . . . .	275

<b>Chapter 12. Integrations . . . . .</b>	<b>277</b>
AppScan Enterprise . . . . .	277
Importing AppScan Enterprise license permissions . . . . .	277
Publishing to AppScan Enterprise . . . . .	277
Creating a job in AppScan Enterprise . . . . .	278
Rational ClearQuest . . . . .	279
HP Quality Center. . . . .	279
<b>Chapter 13. Best practices . . . . .</b>	<b>281</b>
Workflow for advanced users . . . . .	281
Initial Configuration . . . . .	283
Initial Automatic Explore . . . . .	284
Improve site coverage manually . . . . .	284
Evaluate Explore results. . . . .	286
Additional configuration . . . . .	288
Sites that use parameter-based navigation . . . . .	288
The challenge of parameter-based navigation sites . . . . .	289
Live production environments. . . . .	290
Flash content . . . . .	292
<b>Chapter 14. Troubleshooting . . . . .</b>	<b>295</b>
License troubleshooting . . . . .	295
Troubleshooting features. . . . .	295
Reporting false positive results . . . . .	296
Troubleshooting the Report False Positive feature . . . . .	296
Extended Support Mode. . . . .	296
Changing the default browser . . . . .	297
Login troubleshooting . . . . .	298
Action-based login troubleshooting . . . . .	300
Request-based login troubleshooting. . . . .	300
Long or never-ending Explore stage . . . . .	301
Flash movie troubleshooting . . . . .	302
Some Flash movies are not scanned . . . . .	303
Restore Adobe Flash Player settings . . . . .	304
Scan Log messages . . . . .	305
AppScan Log messages . . . . .	314
Flash Log messages . . . . .	319
Glass box troubleshooting . . . . .	320
<b>Chapter 15. CLI . . . . .</b>	<b>321</b>
Command structure . . . . .	321
Commands . . . . .	321
Exec command . . . . .	321
Report command . . . . .	324
Help command. . . . .	325
Exit Status codes . . . . .	325
Launching AppScan from the command line . . . . .	325
<b>Chapter 16. Menus, toolbars and keyboard shortcuts . . . . .</b>	<b>327</b>
File menu . . . . .	327
Edit menu . . . . .	327
View menu . . . . .	328
Scan menu . . . . .	328
Tools menu . . . . .	329
Help menu . . . . .	330
Main toolbar . . . . .	331
Browser toolbar . . . . .	332
Keyboard shortcuts . . . . .	332
Accessibility controls . . . . .	333
<b>Chapter 17. Glossary . . . . .</b>	<b>337</b>
<b>Chapter 18. Notices for IBM Security AppScan Standard V9.0.1 documentation . . . . .</b>	<b>349</b>



---

# Chapter 1. Introduction

A product overview, summary of what's new in this version and contact information.

---

## Product overview

IBM Security AppScan® Standard is a security vulnerability testing tool for web applications and web services. It features the most advanced testing methods to help protect your site from the threat of cyber-attack, together with a full range of application data output options.

AppScan Standard employs three distinct testing techniques that complement and enhance each other:

### **Dynamic Analysis ("black-box scanning")**

This is the primary method, testing and evaluating application responses during run-time.

### **Static Analysis ("white-box scanning")**

This is a unique technology that analyzes JavaScript code in the context of the full web page.

### **Interactive Analysis ("glass box scanning")**

The dynamic test engine can interact with a dedicated glass-box agent which resides on the web-server itself, enabling AppScan to identify more issues, and with greater accuracy, than by conventional dynamic testing alone

AppScan's advanced capabilities include:

- General and regulatory compliance reporting, with over 40 different templates available out-of-the-box
- Customization and extensibility through the AppScan eXtension Framework, or by direct integration into existing systems using the AppScan SDK
- Link categorization capabilities that go beyond application security to identify risks posed to users from links to malicious or other unwanted sites

AppScan Standard helps you decrease the risk of web application attacks and data breaches both before site deployment and for ongoing risk assessment in production.

---

## What's new

This section describes new product features and enhancements in this version.

A complete list of fixes can be found at: <http://www.ibm.com/support/docview.wss?uid=swg27021374>

### **New in IBM Security AppScan Standard 9.0.1**

The main release included a variety of fixes, as well as the following new features:

#### **Updated wizard now includes exploring web services with AppScan as recording proxy**

The new feature introduced in version 9.0.0.1, for exploring web services using AppScan as a recording proxy, is now incorporated into the Scan Configuration wizard, enabling a simpler workflow.

**Note:** For web services with a WSDL file, you can still explore using Generic Service Explorer (GSC).

#### **Manual Explore Export/Import**

Exported Manual Explore data now includes responses, improving AppScan's testing ability. In

most cases importing the responses will improve scan accuracy, unless the site has changed since the Manual Explore was performed (in which case a new Manual Explore is recommended).

**Note:** When a legacy Manual Explore file (without responses) is imported, a dialog box recommends re-recording the Manual Explore, but also offers a "Continue anyway" option.

**Note:** When a legacy Manual Explore file is imported as part of a scan run using the CLI, the requests are sent again as part of the Explore stage, and the new responses used to analyze tests for the Test stage.

### Defining sequence variables

It is now possible to define the format of sequence variables found in multi-step operations, from the Configuration dialog box (Multi-Step Operations view).

For example, if the sequence registers a new user, and a new email address must be entered each time (so that the site will initiate the registration process rather than an "Existing User" page), the right-click menu now lets you define the variable as a "Random email address". Then, each time a test request is sent that includes the variable, a different email address will be used, enabling AppScan to test the registration pages.

### Multiple parameter exclusion

In Data view you can now select multiple parameters and right-click to exclude them from testing ("Do not test selected parameter(s)").

### Smart card and other client-side certificates

AppScan now supports multiple client-side certificates from the user's Windows Personal Certificates Store, including certificates from Smart Cards and US DoD Common Access Cards (CAC).

### Glass box .NET agent now supports invisible parameters

This enables AppScan to identify HTTP parameters that are not visible to black box scanners, improving scan coverage. No special configuration is needed. Until now, invisible parameters were supported only for Java platforms.

### JavaScript execution

Improved Explore stage stability and performance. Supports multi-threading and additional JavaScript capabilities.

### Improved Issue Information tips

Additional validation and other informational tips were added for a number of security issues.

### Improved server-down detection

### Security Reports now include CVSS scores

### Explore Optimization extension behavior

Additional configuration options (Tools > Extensions > Explore Optimization: Configure), and AppScan now automatically suggests activating it when the Explore stage reaches more than 1,000 links.

### WebSphere® portal support

If your site is a WebSphere Portal site, you can now configure AppScan to decode the URLs automatically, for a more efficient scan (Explore Options view).

### Improved report creation

Improved stability, reduced report creation time, and AppScan will now warn you if a report is going to be very large, so you can change the settings to define a more manageable one.

## Deprecated in version 9.0.1

- Support for logging defects found by AppScan directly to defect tracking systems (Rational ClearQuest and HP Quality Center) is deprecated as of this version.
- Generating security reports in RTF format is deprecated as of this release. The functionality will be removed altogether in a future release.

## Contact and support information

AppScan contact information for technical support, to report false positive test results, and for technical, sales and general information.

Item	Details
Documentation	The AppScan Standard Publications Library links to all online user documentation, including: <ul style="list-style-type: none"><li>• PDF version of this User Guide</li><li>• Readme file, containing any last minute information that could not be included in this Help</li><li>• Fix List, detailing APARs fixed by version</li><li>• System requirements</li><li>• Known issues of general interest in the current version (updated as issues are discovered and as they are resolved in fix packs)</li><li>• AppScan Standard download instructions</li></ul> <a href="http://www.ibm.com/support/docview.wss?uid=swg27024868">http://www.ibm.com/support/docview.wss?uid=swg27024868</a>
Support portal	<a href="http://www.ibm.com/support/entry/portal/product/software/security_systems/ibm_security_appscan_standard">http://www.ibm.com/support/entry/portal/product/software/security_systems/ibm_security_appscan_standard</a>
To open a service request	<a href="http://www.ibm.com/support/entry/portal/Open_service_request">http://www.ibm.com/support/entry/portal/Open_service_request</a>
To report "false positive" results	<a href="http://www.ibm.com/support/docview.wss?uid=swg21295428">http://www.ibm.com/support/docview.wss?uid=swg21295428</a> For more details see: "Report false positive test results" on page 187
AppScan eXtensions framework	<a href="http://www.ibm.com/developerworks/rational/downloads/08/appscan_ext_framework/">http://www.ibm.com/developerworks/rational/downloads/08/appscan_ext_framework/</a> For more details see: "Extensions" on page 267
Sales and general information	<a href="http://www.ibm.com/software/rational/offering/testing/webapplicationsecurity/">http://www.ibm.com/software/rational/offering/testing/webapplicationsecurity/</a>

**Note:** When calling or submitting a problem to AppScan Support about a particular service request, have the following information ready:

- Customer Number
- The machine type/model/serial number (for software maintenance calls)
- Company name
- Contact name
- Preferred means of contact (voice or email)
- Telephone number where you can be reached (if voice contact requested)
- Related product and version information
- Related operating system and database information
- Detailed description of the issue

- Severity of the issue with reference to your business needs

---

## Chapter 2. Installing

Installation and license procedures.

---

### System requirements

A summary of the minimum hardware and software required to run AppScan Standard.

**Important:** A more complete list, which may include updates added after the product was released, can be found online at: <http://www.ibm.com/support/docview.wss?uid=swg27024155>

#### Hardware requirements

Hardware	Minimum Requirement
Processor	Core 2 Duo 2 GHz (or equivalent)
Memory	3 GB RAM
Disk Space	30 GB
Network	1 NIC 100 Mbps for network communication with configured TCP/IP

#### Operating system and software requirements

Software	Details
Operating System	Supported operating systems (both 32-bit and 64-bit editions): <ul style="list-style-type: none"><li>• Microsoft Windows Server 2012: Essentials, Standard and Datacenter</li><li>• Microsoft Windows Server 2012 R2: Essentials, Standard and Datacenter</li><li>• Microsoft Windows Server 2008: Standard and Enterprise, SP2</li><li>• Microsoft Windows Server 2008 R2: Standard and Enterprise, with or without SP1</li><li>• Microsoft Windows 8.1: Pro and Enterprise</li><li>• Microsoft Windows 8: Standard, Pro and Enterprise</li><li>• Microsoft Windows 7: Enterprise, Professional and Ultimate, with or without SP1</li></ul>
Browser	Microsoft Internet Explorer 8, 9, 10, 11
License Key Server	Rational® License Key Server 8.1.1, 8.1.2, 8.1.3, 8.1.4
Other	Microsoft .NET Framework 4.5  (Optional) Adobe Flash Player for Internet Explorer, Version 10.1.102.64 or higher is required for Flash execution (and for viewing instructional videos in some of the advisories). Earlier versions are not supported, and some versions may require configuration. For details, see "Flash content" on page 292.  (Optional) Microsoft Word 2007, 2010, 2013 for custom report templates.

**Important:** Customers without a local license on their machine require a network connection to their licensing server when using AppScan.

**Important:** A personal firewall running on the same computer as AppScan can block communication and result in inaccurate findings and reduced performance. For best results do not run a personal firewall on the computer that runs AppScan.

## Glass box server requirements

The glass box scanning feature requires a glass box agent to be installed on the application server. For more details, see “[Installing the glass box agent](#)” on page 136.

**Java platforms:** On Java platforms the following server platforms and technologies are supported.

Software	Details
Operating System	Supported Microsoft Windows systems (both 32-bit and 64-bit editions): <ul style="list-style-type: none"><li>• Microsoft Windows Server 2012</li><li>• Microsoft Windows Server 2012 R2</li><li>• Microsoft Windows Server 2008 SP2</li><li>• Microsoft Windows Server 2008 R2</li></ul> Supported Linux systems: <ul style="list-style-type: none"><li>• Linux RHEL 5, 6, 6.1, 6.2, 6.3, 6.4</li><li>• Linux SLES 10 SP4, 11 SP2</li></ul> Supported UNIX systems: <ul style="list-style-type: none"><li>• UNIX AIX® 6.1, 7.1</li><li>• UNIX Solaris (SPARC) 10, 11</li></ul>
Java™ EE container	JBoss AS 6, 7; JBoss EAP 6.1; Tomcat 6.0, 7.0; WebLogic 10, 11, 12; WebSphere 7.0, 8.0, 8.5, 8.5.5

**.NET platforms:** On .NET platforms the following systems and technologies are supported:

Item	Details
Operating System	Supported operating systems (both 32-bit and 64-bit editions): <ul style="list-style-type: none"><li>• Microsoft Windows Server 2012</li><li>• Microsoft Windows Server 2012 R2</li><li>• Microsoft Windows Server 2008 SP2</li><li>• Microsoft Windows Server 2008 R2</li></ul>
Other	Microsoft IIS 7.0 or later  Microsoft .NET Framework 4.0 or 4.5 must be installed, and IIS must be configured at the root level to work with this version of ASP.net

**Note:** User must have administrator privileges when running the application on the server.

**Note:** The agent should be installed *after* the application you want to test is successfully installed on the server.

## Cookie usage

Although AppScan tracks cookies set by the application it is scanning, it does not set cookies of its own for authentication or any other purpose.

## Supported languages

The AppScan user interface can run in the following languages: Chinese (Simplified), Chinese (Traditional), English (United States), French, German, Italian\*, Japanese, Korean, Portuguese (Brazil), Spanish (Spain). To change the user interface language go to **Tools > Options > General tab**

**Note:** For the Italian user interface, the documentation is in English.

## Flash Player upgrade

### About this task

In order for AppScan to be able to execute Adobe Flash content during scanning, you must have a supported version of the Adobe Flash Player for Internet Explorer installed. Versions 9.0.124.0 and higher are supported. If you have an earlier version, or none, you must install a supported version in order to execute Flash files during scanning.

**Note:** Until you upgrade Flash Execution will not run even though the Flash Execution check box (Scan Configuration > Explore Options) is selected.

### Procedure

1. Close AppScan and any Microsoft Internet Explorer windows.
2. To download and install the latest Flash Player, go to <http://get.adobe.com/flashplayer/>

## Flash Player configuration

### About this task

In order for AppScan to execute Adobe Flash content during scanning, you must have a supported version of Adobe Flash Player for Internet Explorer installed. In some cases Flash Player Version 10.1 or higher may require configuration to work with AppScan. If you get a message that your Flash Player requires configuration, follow the procedure below.

**Note:** Without configuration Flash Execution will not run even though the Flash Execution check box (Scan Configuration > Explore Options) is selected.

**Note:** This procedure requires Administrator permissions.

### Procedure

1. Close AppScan.
2. *With Administrator permissions*, open the folder containing the Flash installation files.
  - For 32-bit systems the path is usually:  
`C:\WINDOWS\System32\Macromed\Flash`
  - For 64-bit systems the path is usually:  
`C:\WINDOWS\SysWow64\Macromed\Flash`
3. Look for a file named `mms.cfg` in the Flash folder. If there is no such file, create an empty TXT file with this name.
4. Open `mms.cfg` with a text editor such as Microsoft Notepad, and search for the entry  
`FullFramerateWhenInvisible`
  - If it exists, set its value to 1
  - If it does not exist, add the following line, *as a separate line*, after any existing content in the file:  
`FullFramerateWhenInvisible = 1`
5. Save.

The Flash Player is now configured for AppScan Flash Execution.

---

## Install

The installation wizard guides you through the fast and simple process.

## Procedure

1. If you have on your machine a version of AppScan Standard later than 8.5, you must uninstall it before installing this version.

**Note:** You can keep version 8.5 on the same machine as this version, for the purpose of opening or converting legacy scans (see "Legacy scan files" on page 132).

2. Close any Microsoft Office applications that are open.

3. Start AppScan setup.

The InstallShield Wizard starts, and checks that your workstation meets the minimum installation requirements. Then the AppScan installation wizard welcome screen appears.

4. Follow the wizard instructions to complete AppScan installation.

**Note:** You will be asked to install or download GSC (Generic Service Client). This is needed for exploring web services in order to configure a web services scan, but is not needed if you will not be scanning web services.

---

## Silent install

Instructions for unattended installation, using the command line.

You can install AppScan "silently", using the command line and the following parameters:

`AppScan_Setup.exe /l"LanguageCode" /s /v"/qn INSTALLDIR=\"InstallPath\""`

Parameter	Function
/l	Language code. Options are: <ul style="list-style-type: none"><li>• English: <b>1033</b></li><li>• Chinese (Traditional): <b>1028</b></li><li>• Chinese (Simplified): <b>2052</b></li><li>• French: <b>1036</b></li><li>• German: <b>1031</b></li><li>• Italian: <b>1040</b></li><li>• Japanese: <b>1041</b></li><li>• Korean: <b>1042</b></li><li>• Portuguese: <b>1033</b></li><li>• Spanish: <b>1034</b></li></ul>
/s	Activates "Silent Mode" (otherwise the regular installation will be launched). <b>Note:</b> Must be used in conjunction with "/v"/qn" (see next row)
/v	Sets additional MSI properties such as UI mode and the path where AppScan will be installed.  <b>UI Mode:</b> For "Silent Mode", include /qn as a parameter (enclosed in quotes).  <b>Path:</b> If you do not define an install path, installation uses the default path: ...Program Files\IBM\AppScan Standard\  To define a different install path, add <code>INSTALLDIR=\"InstallPath\"</code> as a parameter (enclosed in quotes). The path may include spaces.  <b>Example:</b> <code>/v"/qn INSTALLDIR=\"D:\Program Files\AppScan\""</code>

### **Examples:**

- To silently install an English version of AppScan in the default directory enter:  
`AppScan_Setup.exe /s /v"/qn"`
- To silently install Japanese versions of AppScan in the default directory enter:  
`AppScan_Setup.exe /l"1041" /s /v"/qn"`
- To silently install a Korean version of AppScan in D:\Program Files\AppScan\ enter:  
`AppScan_Setup.exe /l"1042" /s /v"/qn INSTALLDIR=\"D:\Program Files\AppScan\""`

---

## **Uninstall**

Instructions for uninstalling the program.

### **About this task**

Use the Add/Remove Programs facility in the Windows Control Panel to remove AppScan from your computer. The exact procedure may vary depending on which version of Windows is installed.

### **Procedure**

1. Go to **Control Panel > Add/Remove Programs**
2. Select **IBM Security AppScan**.

**Note:** If you have two versions of AppScan on your computer, make sure you select the correct one.  
3. Click **Remove** and follow the instructions.

**Note:** The uninstall wizard does not delete scan files or reports created with AppScan. If you want to delete these files, do so manually.

---

## **Legacy scans**

This section describes how the current version of AppScan deals with scans from versions earlier than 8.6 ("legacy" scan files).

In AppScan version 8.6 the scan file format was redesigned. If you have a version of AppScan earlier than 8.6 installed on your machine, it need not be removed when you install the current version. The two versions can exist on the same machine, and even be open together.

Due to the new design, scans saved in the earlier version (legacy files) cannot be opened in the newer version. It is possible to convert the Explore stage data from a legacy scan and import it into a new scan with the same configuration, or to use the configuration only to create a new scan, but the Test stage data cannot be converted.

Therefore, if you try to open a legacy scan in the newer version, you are given the following options.

### **If an earlier version of AppScan is installed on the same machine:**

- **Convert:** Convert Explore stage data and import into a new scan. This may take some time. Note that data from the Test stage is not included, and if you need Test data you will need to run the Test stage on the new scan.
- **New Scan:** Import the configuration only into a new (empty) scan. You can then run a full scan in the new version of AppScan.
- **Open in legacy version of AppScan:** Open the original scan in the older version of AppScan.

### **If an earlier version of AppScan is *not* installed on the same machine:**

- Import the configuration only into a new (empty) scan. You can then run a full scan in the new version of AppScan.
- Download AppScan Standard 8.5 to view results or convert Explore stage data.

## **Limitations**

The Convert and Import functionality for legacy files has the following limitations:

- For the conversion process to start, the starting URL of the scan being imported must be covered by your license.
- Because of the difference in logic and architecture between the older and newer versions of AppScan, Explore stage results converted to the new format will not look identical to the original results, nor to the results of an Explore stage run in the new version.
- Unexplored links do not appear in the converted scan. If you need these in your results you should run a new scan using the original configuration.
- Scans saved in a version of AppScan earlier than 8.0 cannot be converted or imported to the new format directly. They must first be opened and saved in version 8.0/8.5 and can then be converted or imported into the current version.

## **Program file locations**

The current default path for AppScan is:

...(Program Files Folder)\IBM\AppScan Standard

The default path for versions of AppScan earlier than 8.6 is:

...(Program Files Folder)\IBM\Rational AppScan

## **License**

A description of license types, installation and management.

The AppScan installation includes a default license that allows you to scan IBM's custom designed AppScan testing website ([demo.testfire.net](http://demo.testfire.net)), but no other sites. In order to scan your own site you must install a valid license supplied by IBM®. Until this is done AppScan will load and save scans and scan templates, but it will not run new scans on your site.

### **Rational licenses**

From Version 7.8 onwards, AppScan licenses are downloaded from the Rational License Key Center. There are three types of license:

#### **"Floating" licenses**

These are installed onto the IBM Rational License Server (which can be the same as the machine on which AppScan runs). Any server on which AppScan is used must have a network connection with the license server. Each time a user opens AppScan a licence is checked out, and when AppScan is closed the license is checked back in.

#### **"Token" licenses**

These are installed onto the IBM Rational License Server (which can be the same as the machine on which AppScan runs). Any server on which AppScan is used must have a network connection with the license server. Each time a user opens AppScan the required number of tokens are checked out, and when AppScan is closed they are checked back in.

#### **"Node-locked" licenses**

These are installed onto the machine on which AppScan runs. Each license is assigned to a single machine.

## License status

To view license status:

- Click **Help > License**. The License dialog box opens, showing license status and the following options:

<b>Load IBM Rational License</b>	If you have an IBM Rational license (either on your computer or on a different network server), click here to open the AppScan License Key Administrator, from where you can load and manage your licenses. The program can also be opened from <code>..\IBM\RationalRLKS\common\licadmin8.exe</code>
<b>Add AppScan Enterprise License</b>	If your organization has an AppScan Enterprise license that allows scanning additional sites to those allowed by your local AppScan Standard license, you can import these permissions to use on your local machine in addition to your existing license. <b>Note:</b> This option is available only when a full AppScan Standard license (not a demo license) is loaded. See "Importing AppScan Enterprise license permissions" on page 277.
<b>View License Agreement</b>	Click here to see the license agreement.

**Note:** You can refresh the license information displayed in the dialog box by clicking 

**Note:** If a floating or token license has been verified, but the license server later becomes unavailable, AppScan can run in "Disconnected Mode" for up to three days. During this time you can scan your application as usual.

*See also:*

"Load a floating or token license"

"Load a node-locked license" on page 12

"License troubleshooting" on page 295

"Importing AppScan Enterprise license permissions" on page 277

## Load a floating or token license

How to load a floating or token license for use with AppScan Standard Edition.

### About this task

In order to install a floating or token license you must first have a license server with Rational License Server version 8.1.2 or later installed on it. The license server can be a different machine or the same machine that Rational AppScan is installed on. The procedure below describes setting up the license server and then loading a floating license.

### Procedure

- Download Rational License Server version 8.1.2 (or later) from Passport Advantage.
- Install the Rational License Server. This can be on the same computer as AppScan, or on your central network license server.
- Download your license files from Rational License Key Center, and save them to the machine on which you installed Rational License Server.
- Click **Start > Programs > IBM Rational > License Server**, and use the Import a Rational license file wizard to import the license file to the license server.

5. In AppScan, click **Help > License > Load IBM Rational License** to open the IBM Rational License Key Administrator.
6. If the License Key Wizard doesn't open automatically, click **License Keys > License Key Wizard** to open it.
7. Select **Point to a Rational License Server**, and click **Next**.
8. Type in the name of the server on which you installed the license in Step 2, click **Show Licenses**, then click **Finish**.
9. Close the License Key Administrator window.
10. In the AppScan license dialog box, click  to load your license.

## Load a node-locked license

How to load a node-locked license.

### About this task

"Node-locked" licenses are installed on the machine on which Rational AppScan runs, not a separate server.

### Procedure

1. Download your license file from Rational License Key Center, and save it on your machine.
2. In AppScan, click **Help > License > Load IBM Rational License** to open the IBM Rational License Key Administrator.
3. If the License Key Wizard doesn't open automatically, click **License Keys > License Key Wizard** to open it.
4. Select **Import a Rational License File**, and click **Next**.
5. Click **Browse**, browse to the location of the file, click **Import**, then click **Finish**.
6. Close the License Key Administrator window.
7. In the AppScan license dialog box, click  to load your license.

---

## Updates

Keeping your installation up-to-date.

### About this task

Subscription updates include new types of web application exploitation techniques and bug fixes. It is recommended that you install these files as soon as you receive notification of their availability.

AppScan periodically checks for updates on the IBM website and notifies you when new updates are available. You may also initiate an update search.

After AppScan has detected new available updates you are given the opportunity to download and install the new update files on your machine.

### Procedure

1. On the Toolbar, click 
- AppScan checks for updates. If an update is available the **Install** button becomes active. (If your version of AppScan is up-to-date the button remains grayed out.)
2. To install the update, click **Install**.

## **What to do next**

You can check update status in the “Update Log” on page 274.

---

## **Temp file location**

Describes where AppScan saves its temporary files during normal operation, and how to change the location.

By default AppScan stores its temporary files in:

C:\Documents and Settings\All Users\Application Data\IBM\AppScan Standard\temp

If you need to override this default location for any reason, edit the path for the environment variable APPSCAN\_TEMP as required. (Environment variables are accessed by right-clicking **My Computer**, and then selecting **Properties > Advanced > Environmental Variables**.)

**Restriction:** There must be no Unicode characters anywhere in the path to the new location.



---

## **Chapter 3. Getting started**

This section provides a short tour of basic product features and procedures.

---

### **How an automatic scan works**

This topic explains the difference between the "stages" and "phases" of a scan.

An AppScan Full Scan consists of two stages: Explore and Test. It is useful to understand the principal behind this, even though most of the scan process is in fact seamless to the user, and little user input is required until the scan is complete.

#### **Explore stage**

During the first stage, AppScan explores the site (web application or web service) by simulating a web user clicking on links and completing form fields. This is the Explore stage.

AppScan analyzes the responses to each request it sends, looking for any indication of a potential vulnerability. When AppScan receives a response that may indicate a security vulnerability, it automatically creates a test (or tests) based on the response, as well as noting the validation rules needed to determine which results constitute vulnerability, and the level of security risk involved.

Before sending the site-specific tests created, AppScan sends several malformed requests to the application to determine the manner in which it generates error responses. This information is then used to increase the precision of AppScan's automatic test validation process.

#### **Test stage**

During the second stage, AppScan sends thousands of custom test requests that it created during the Explore stage. It records and analyzes the application's response to each test using the custom validation rules. These rules both identify security problems within the application and also rank their level of security risk.

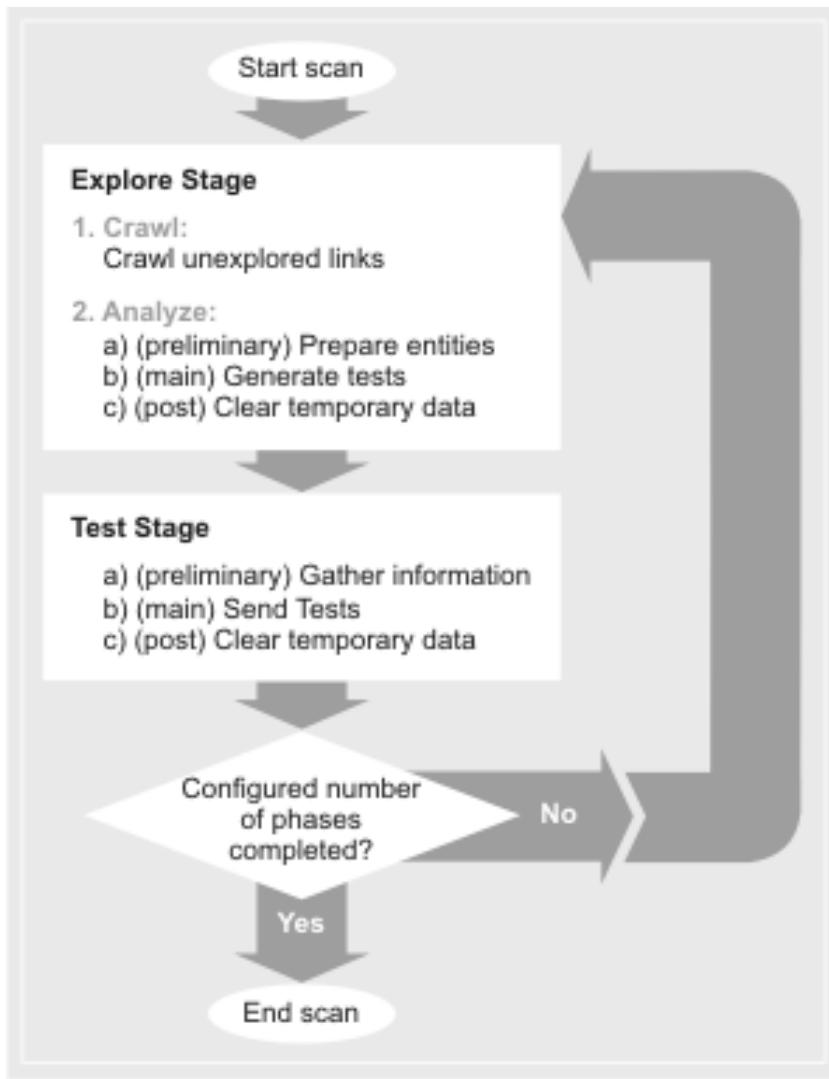
#### **Scan phases**

In practice, the Test stage frequently reveals new links within a site, and more potential security risks. Therefore, after completing the first "phase" of Explore and Test, AppScan automatically begins a second "phase" to deal with the new information. If new links are discovered during the second phase, a third phase is run, and so on.

After completing the configured number of scan phases (user configurable; default four), scanning stops and the completed results are available to the user.

#### **Illustration of automatic scan flow**

The following diagram illustrates the stages and phases of automatic scan flow. Note that this process requires no action from the user, but you may come across them referred to in the AppScan log.



## Scanning web services

This topic explains the different methods available for exploring sites, before AppScan tests them.

A site is scanned by first exploring it, and then, based on the Explore stage responses, testing it. There are different ways of gathering this "Explore data". In all cases, once the data is gathered AppScan is used to send tests to the site.

### Exploring sites without web services

- In the case of sites without web services it is often sufficient to supply AppScan with the start URL and login authentication credentials for it to be able to test the site.
- If necessary you can (also) manually explore the site *through AppScan*, in order to get access to areas that can only be reached through specific user input.

### Exploring web services

- In order to scan a web service, you can set up AppScan as a recording proxy for the device (such as a mobile phone or simulator) you use to explore the service. That way AppScan can analyze the Explore data collected, and send appropriate tests. See "Using AppScan as recording proxy" on page 121

- If you have a WSDL file for your web service (such as a SOAP web service), the AppScan installation optionally includes a separate tool that lets users view the various methods incorporated in the web service, manipulate input data, and examine feedback from the service. You first need to give AppScan the URL of the service. The integrated "Generic Service Client (GSC)" uses the WSDL file to display the individual methods available in a tree format, and create a user-friendly GUI for sending requests to the service. You can use this interface to input parameters and view the results. The process is "recorded" by AppScan and used to create tests for the service when AppScan scans the site. See "Using GSC" on page 124

## Basic workflow

A diagram showing a simple AppScan workflow using the scan configuration wizard.



For more details of the basic workflow, see "Workflow description"

Users with experience in the field of web security, see "Workflow for advanced users" on page 281

## Workflow description

AppScan provides a comprehensive assessment of your web application. It runs thousands of tests based on all levels of typical user techniques as well as unauthorized access and code injections.

When you run a scan on your application, the tests are sent by AppScan to your web application. The results of the tests are provided by AppScan's site-smart engine and result in expansive reports and fix recommendations, available for enhanced review and manipulation.

AppScan is an interactive tool: you decide on the configuration of the scan and determine what is to be done with the results.

The AppScan workflow includes the following stages:

1. **Select a Template:** A predefined scan configuration is a *scan template*. You can load the Regular Scan template, another predefined template, or a template that you previously saved. (You can later adjust the configuration as required for the current scan.)
2. **Application or web Service Scan:** Scanning web services requires some manual input by the user, using GSC (Generic Service Client), to show AppScan how to *use* the service.
  - **web Application Scan:** If you are *not* scanning a web service, or if you want to scan parts of the application *other* than its web services, leave this default option selected.
  - **web Service Scan:** Select this option if you want to scan a service. GSC (Generic Service Client) will later open to let you send requests to the service, and collect results, for AppScan to analyze and use to create tests.
3. **Scan Configuration:** Configure the scan, taking into account details of your site, your environment, and other requirements.
4. (Optional) **Manual Explore:** Log in to the site, and click links and fill in forms as a user would. This is a good way of "showing" AppScan how a typical user might browse the site, ensuring that important parts of the site are scanned, and providing data for filling forms.
5. (web services only) **Send requests using GSC:** Open GSC and send some valid requests to the service.
6. (Optional) **Run Scan Expert:** This is a short pre-scan of your site to evaluate the configuration. Scan Expert may suggest changes to increase the efficiency of the main scan.
7. **Scan the Application or Service:** This is the main scan, and consists of Explore and Test stages.

**Explore Stage:** AppScan crawls your site, visiting links as a regular user would and records the responses. It creates a hierarchy of the URLs, directories, files, and so on, that it finds on your application. This list is displayed in the Application Tree (see "Application Tree" on page 20).

The Explore stage can be done automatically, manually, or as a combination of both. You can also import an Explore Data File (see "Exporting Manual Explore data" on page 118), which consists of a previously recorded manual explore sequence. AppScan then analyzes the data it has collected from the site, and based on it, creates tests for the site. These tests are designed to reveal weaknesses both in infrastructure (such as security weaknesses in commercial, 3rd Party products or Internet systems), and the application itself.

**Test Stage:** During the Test stage, AppScan tests your application, based on the responses it received during the Explore stage, to reveal vulnerabilities and assess their severity.

An up-to-date list of all tests included in your current version of AppScan can be seen in the Scan Configuration dialog box (see "Test Policy view" on page 87).

You can also create user-defined tests in addition to the tests that AppScan automatically creates and runs (see "User-defined tests" on page 226). Your tests can supplement those generated by AppScan and can verify the results that it found.

Test results are displayed in the Result List, from where you can view and modify them. Full details of the results are displayed in the Detail Pane.

8. (Optional) **Result Expert:** Processes scan results and adds information to the Issue Information tab (in the Detail pane), including CVSS Metric scoring and relevant screen captures.
9. (Optional) **Run Malware Test:** This analyzes pages and links found on your site for malicious and otherwise unwanted content.

**Note:** Although a Malware Test can in principle be performed at this stage (in which case it will use the Explore stage results of the main scan), in practice a Malware Test is usually run on a *live* site, whereas a regular scan is usually run on a *test* site (because of the risk of disrupting a *live* site by scanning it).

## 10. Review Results to evaluate the security status of the site. Result Expert can help you with this.

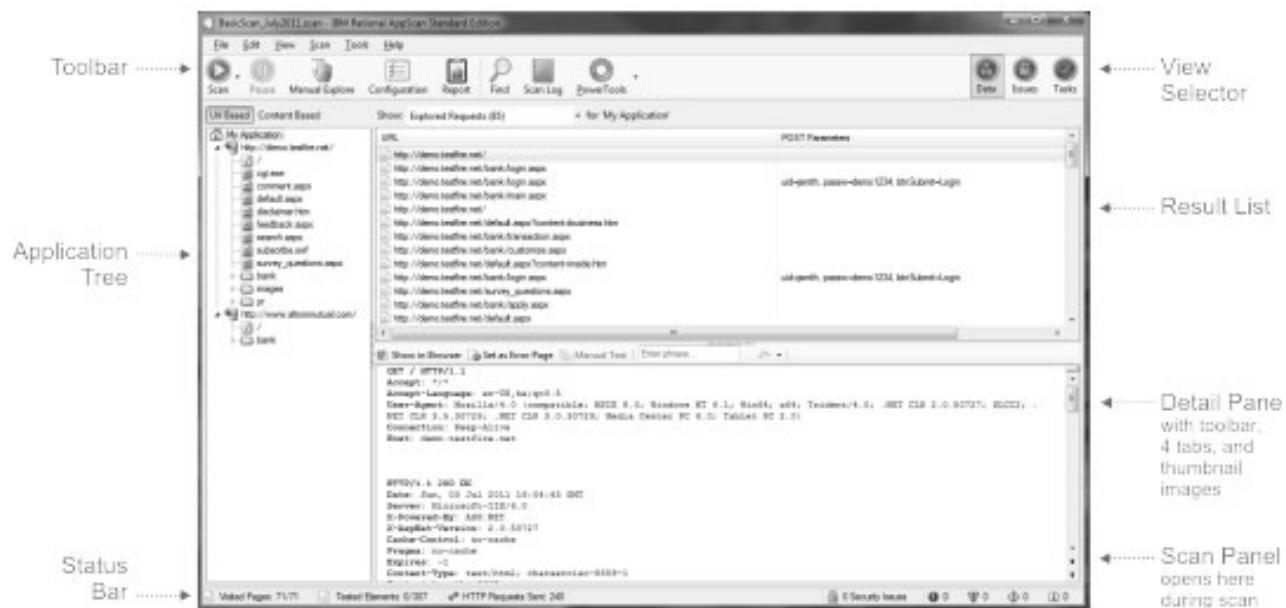
You may also want to:

- Explore additional links manually
- Review Remediation Tasks
- Print Reports
- Adjust the scan configuration, if necessary based on your review of the results, and scan again

**Note:** For a simplified illustration of this workflow, see “Basic workflow” on page 17.

## Tour of the main window

Describes the components of the AppScan main window, and all menus and toolbars.



You can resize the panes by clicking-and-dragging on the dotted section of any splitter bar (the bar between panes).

You can arrange the two right-hand panes vertically or horizontally, by clicking **View > Layout > Vertical/Horizontal**.

See also: Chapter 16, “Menus, toolbars and keyboard shortcuts,” on page 327.

## View selector

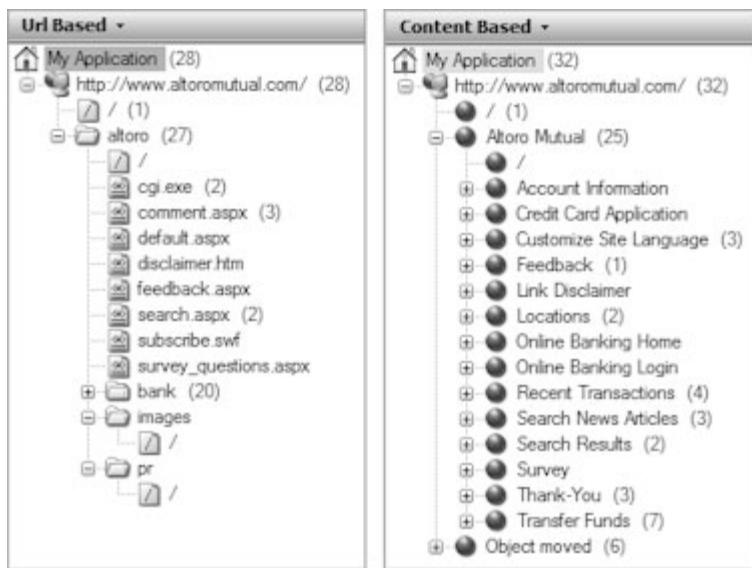
The View Selector icons on the right-hand side of the toolbar toggle between the different result views.

When you select a different view in the View Selector, the information displayed in the Application Tree, Result List and Detail Pane changes. The three parts of the screen are summarized in the table following.

	<p><b>Data view</b></p> <p>Shows script parameters, interactive URLs, visited URLs, broken links, filtered URLs, comments, JavaScripts and cookies from the Explore stage.</p> <p><b>Application Tree:</b> Complete application tree.</p> <p><b>Result List:</b> Select a filter from the pop-up list at the top of the Result List, to determine which information is displayed.</p> <p><b>Detail Pane:</b> Filtered list of: script parameters, interactive URLs, visited URLs, broken links, filtered URLs, comments, JavaScripts and cookies. Unlike the other two views, Application data view is available even if AppScan has only completed the Explore stage. Use the pop-up list at the top of the Result list to filter the data.</p> <p>Keyboard shortcut: <b>F2</b></p> <p>Chapter 7, “Results: Application Data,” on page 167</p>	
	<p><b>Issues view</b></p> <p>Shows the actual issues discovered, from overview level down to individual requests/responses. This is the default view.</p> <p><b>Application Tree:</b> Complete application tree. Counters next to each item show the number of issues found for the item.</p> <p><b>Result List:</b> Lists issues for the selected note in the application tree, and the severity of each issue.</p> <p><b>Detail Pane:</b> Shows advisory, fix recommendations and request/response (including all variants used) for the issue selected in the Result List.</p> <p>Keyboard shortcut: <b>F3</b></p> <p>Chapter 8, “Results: Security Issues,” on page 173</p>	
	<p><b>Tasks view</b></p> <p>Provides a To Do list of specific remediation tasks to fix the issues found by the scan.</p> <p><b>Application Tree:</b> Complete application tree. Counters next to each item show the number of fix recommendations for that item.</p> <p><b>Result List:</b> Lists remediation tasks for the selected node in the application tree, and the priority of each task.</p> <p><b>Detail Pane:</b> Shows details of the remediation task selected in the Result List, and all the issues that this remediation will solve.</p> <p>Keyboard shortcut: <b>F4</b></p> <p>Chapter 9, “Results: Remediation Tasks,” on page 193</p>	

## Application Tree

The Application Tree is a tree-view display of the folders, URLs, and files that AppScan found on your application.



**Note:** In the case of applications without a hierarchical URL structure, such as single entry-point applications (e.g. MVC), or applications where the hierarchical structure is not logical, you can create a "content-based" application tree, by defining a set of regular expressions which extract the logical path from the pages. (See "Content-Based Results view" on page 82.)

## Application Tree icons

Each of the various node types in the Application Tree is indicated by its own icon.

Icon	Indicates
	My Application, root node.
	Scanned host. If you set up Additional Servers or Domains ("URL and Servers view" on page 43), or if you have a server on different ports, you may have multiple nodes of this type.
	Folder (path) found in your application.
	Slash; results of tests for the parent folder.
	File found in your application.
	A red X on any icon in the Application Tree (file icon illustrated here) indicates that the node and all its child nodes have been excluded from scans by the User. (To re-include such a node in future scans, right-click and select <b>Include in Scan</b> .)  Note: A child node can be included even if its parent node is excluded.

**Note:** If a URL in the application tree contains *only* error responses, the URL is displayed with strike through formatting (with a line through it).

## Application Tree counters

The Application Tree counters (the numbers in parentheses next to each node of the tree) change according to the view you select in the View Selector:

- **Security Issues:** Counters indicate the number of issues relevant to the node and all its child nodes. (The total number of issues is shown at the top of the “Result List.”)
- **Remediation Tasks:** Counters indicate the number of remediation tasks relevant to the node and all its child nodes.
- **Application Data:** No counters.

## Application Tree right-click menu

If you right-click an item (other than My Application) in the Application Tree, the context-sensitive menu offers some or all of the following options:

Menu Item	Function
Show in Browser	Opens the embedded browser to the selected URL.
Manual Explore	Starts a Manual Explore of the selected item. See “Using AppScan” on page 115 for details.
Manual Test	Opens the Manual Test dialog box. See “Manual tests” on page 189 for details
Record Multi-Step Operation	Record a multi-step operation sequence for testing parts of the site that can only be reached by clicking links in a specific order. See “Multi-Step Operations view” on page 79 for details.
Copy URL	Copies the selected URL to the clipboard. (Where there are more than one sub-nodes the URL of the first sub-node is copied.)
Report for this node	Create a report for the currently selected node in the application tree (including all its child nodes). See “Creating partial reports” on page 198 for details.
Exclude from Scan	Excludes the selected URL (or all URLs under the selected node) from scans. (Toggles with “Include in Scan”) See “Exclude URL from scan” on page 173 for details.

## Result List

The results of the scan on your web application are displayed in the Result List. The total number of issues and variants appears at the top of the list.

The screenshot shows a software interface for managing security issues. At the top, a toolbar has a dropdown labeled "Arranged By: Severity | Highest on top". Below this is a header bar with a folder icon and the text "17 Security Issues (66 variants) for 'My Application'". The main area displays a hierarchical list of issues, each preceded by a small icon indicating the type of vulnerability. The issues listed are: Directory Listing (5), SQL Injection (2), Cross-Site Scripting (3), TRACE and TRACK HTTP Methods Enabled (1), Blind SQL Injection (1), Session Identifier Not Updated (1), HTML Comments Sensitive Information Disclosure (2), and Insecure HTTP Methods Enabled (2).

## Result List counters

The Result List counters (the numbers in parentheses next to each node) change according to the view you select in the View Selector:

- **Security Issues:** Counters indicate the number of issues relevant to the node and all its child nodes.

**Note:** The total number of security issues (at the top of the Result List) is a measure of the vulnerable locations in the site, and depends in part on how site is structured. If you define a content-based

structure, the total number of issues in the application tree may not be the same as it is for the URL-based application tree (for the same results). When site structure is content-based (rather than URL-based), and content-based view is configured correctly, the issue count in content-based view represents more accurately the number of "vulnerable locations" that exist in the site. The total number of *variants* (at the top of the Result List in parentheses) is independent of site structure, and does not change between content-based and URL-based views.

- **Remediation Tasks:** Counters indicate the number of remediation tasks relevant to the node and all its child nodes.
- **Application Data:** No counters.

See also: "Content-Based Results view" on page 82

## Detail Pane

This pane shows details of the item selected in the Result List.

The information available in the Detail Pane depends on the selected item and the selected view:

- "Application Data: Detail Pane" on page 172
- "Security Issues: Detail Pane" on page 178
- "Remediation Tasks: Detail Pane" on page 196

## Scan panels

The Progress and Notice panels

### Progress panel



The Progress panel appears below the Result List during a scan. It appears when you start a scan and displays progress information while the scan continues.

### Notice panel

The Notice panel appears in place of the Progress panel, if the AppScan is unable to connect with your web server or if the scan was stopped before it finished. It displays brief information (such as "Server Down", or "Scan is Incomplete"), with a link to open a dialog with details.

## Status bar

The Status Bar at the bottom of the main window shows information for the scan that is currently running or loaded.

Icon	Indicates
	<b>Visited Pages:</b> Number of pages visited / Total number of pages to be visited  The second number may increase and then decrease during the scan, as pages are discovered and then rejected as not requiring scanning. By the end of the scan the two numbers should be equal.

Icon	Indicates
	<p><b>Tested Elements:</b> Number of elements tested / Total number of elements to be tested</p> <p>The <i>second</i> number will increase during the Explore stage, as elements for testing are discovered. During the Test stage the <i>first</i> number will increase. By the end of the scan the two numbers should be equal.</p>
	<p><b>HTTP Requests Sent</b></p> <p>This number represents all requests sent, including in-session detection requests, server-down detection requests, login requests, multi-step operations and test requests. During the scan it is therefore an indicator that AppScan is working, but the actual number is not of any particular significance either during or after the scan.</p>
	<p><b>Security Issues</b></p> <p>Total number of security issues found, followed by the number in each category: High, Medium, Low, and Informational.</p>

## Tutorial

This simple tutorial goes through the steps of configuring a simple application scan using the Scan Configuration wizard, running the scan, and reviewing the results.

This simple tutorial uses the Scan Configuration wizard to scan IBM's "AltoroMutual Bank" website, which has been created for demonstration purposes.

Experienced users may prefer to follow the more advanced workflow (also using the wizard) described in "Workflow for advanced users" on page 281, or to configure in greater detail using the "Scan configuration dialog box" on page 42.

- "Step 1: Configuring the scan"
- "Step 2: Running the scan" on page 25
- "Step 3: Reviewing Scan Results" on page 26
- "Step 4: Communicating results" on page 27

You may like to work through this tutorial using IBM's "AltoroMutual Bank" website, which has been created for demonstration purposes:

URL	http://demo.testfire.net/
Username	jsmith
Password	demo1234

**Note:** If you are using an evaluation copy of AppScan, the AltoroMutual Bank website is the *only* site you can scan.

**Note:** This tutorial is only a quick view of the basic steps for running a scan. For explanations and full instructions, see Chapter 4, "Configuring," on page 29. See also "Workflow for advanced users" on page 281.

### Step 1: Configuring the scan

Step one of the tutorial.

## About this task

The Scan Configuration Wizard provides an easy way to configure a scan, when you do not need to change many of the default configuration settings.

### Procedure

1. Launch AppScan, to open the Welcome Screen or, if AppScan is open, click **File > New** for a similar dialog box.
2. Verify that the **Launch Scan Configuration Wizard** check box is selected, then select the **Regular Scan** template.

The Scan Configuration Wizard Welcome opens.

3. Select the **web Application Scan** radio button, then click **Next**.

The URL and Servers step of the wizard (1/3) appears.

**Note:** If you select the web *Services* Scan option, the flow is slightly different, and when the wizard closes the Generic Service Client (GSC) opens, enabling you to input parameters that AppScan will use for the Test stage of its scan. (For details, see “Wizard for exploring with GSC” on page 39.)

4. Enter the URL of your application in the text box, then click **Next**.

The Login Management step (2/3) appears.

5. Click **Record Login**.

The AppScan browser opens, to the starting URL that you set up in the previous step. Your browsing is now being recorded by AppScan.

6. Log into your application with an authorized username and password.

7. When you have successfully logged-in, close the browser.

The “Login Sequence” (the sequence of links that achieved the logged-in state) is displayed (see “Recording a login” on page 47 for more details), and the gray key icon turns green, indicating that in-session detection is active.

This:  changes to this: 

8. Click **Next**.

The Test Policy step (3/3) appears.

9. Click **Next**.

The final step of the wizard appears. You are now ready to run the scan (see “Step 2: Running the scan”).

**Note:** Although it is possible to start the automatic scan at this stage, in many cases a better result will be achieved by manually exploring the application first, as a regular user would (see “Using AppScan” on page 115).

## Step 2: Running the scan

Step two of the tutorial.

### About this task

When configuration is complete, you can run the scan.

### Procedure

1. Select **Start a full automatic scan**, then click **Finish**.

The wizard closes and Scan Expert starts to evaluate the effectiveness of the current configuration for your site. When the evaluation is complete, a checklist of suggested configuration changes appears.

**Note:** If there are changes that require user input, their check boxes are grayed out and deselected. To provide the required input for these changes, click the link for that change.

## 2. Click **Apply Recommendations**.

The selected configuration changes are applied and scan starts. The Progress Panel opens, and Application Data and Issues are updated in real-time.

During the Explore stage, AppScan crawls your application to discover its pages and content. The **Application Data** tree is updated as this happens, and eventually shows a complete tree of the site. Then, during the Test stage, AppScan runs thousands of tests on the site and reports the issues found and fix recommendations. During this part of the scan, Security Issues view is selected automatically and the Result List displays a dynamically updated list of discovered issues.

A scan can have multiple phases (a phase is a cycle of Explore followed by Test). This happens when AppScan discovers new links during the test stage, and needs to a scan, it will create new tests based on these links, and run an additional scan phase. The subsequent phase is usually shorter than the previous one, as only the new links are scanned. AppScan will add phases as long as it finds new URLs, or until it has reached the configured "scan limit". The default limit is four phases.

When the scan is complete, the Progress Panel closes, and you can review the results (see "Step 3: Reviewing Scan Results").

## Step 3: Reviewing Scan Results

Step three of the tutorial.

### About this task

When the scan is complete, the results are displayed in the main window, in three areas: the Application Tree, the Result List, and the Detail Pane. The type of information presented in each area depends on the View selected (default is Security Issues View).

### Procedure

To access a view, click the relevant icon in the View Selector on the left side of the screen.

View	Description
	<p><b>Data View</b> provides lists of content items found in your application. This can be a useful way of verifying that the scan is configured in a way that will cover your application, <i>before</i> you begin the Test stage.</p> <ul style="list-style-type: none"><li>• <b>Application Tree:</b> Shows the URL and folder nodes.</li><li>• <b>Result List:</b> Shows application data, sorted by scan results (see "Application Data: Result List" on page 167). For example, you can choose to see a list of broken links, of JavaScript, of cookies, and so on.</li><li>• <b>Detail Pane:</b> Shows the request sent to a page and the response that was received.</li></ul> <p>To learn more about the Application Data view, see Chapter 7, "Results: Application Data," on page 167.</p>

View	Description
	<p><b>Issues View</b> provides comprehensive data of the security issues discovered by the scan.</p> <ul style="list-style-type: none"> <li>• <b>Application Tree:</b> Shows the folders, URLs, and files that AppScan discovered in your application are listed. The number next to each node in the tree indicates how many issues were found.</li> <li>• <b>Result List:</b> Shows an icon per issue, indicating the severity value that is assigned to this issue (see “Security Issues: Result List” on page 174). Each issue also holds a consolidation of the URLs that are vulnerable to this issue. Under each URL, is a listing of the vulnerable data.</li> <li>• <b>Detail Pane:</b> Displays information to enable you to understand why this is an issue, what it can do if gone untreated, fix recommendations, the variants of the test requests that were sent, the response of your application for which AppScan marked the test as an issue, and more.</li> </ul> <p>To learn more about the Issues view, see Chapter 8, “Results: Security Issues,” on page 173.</p>
	<p><b>Tasks View</b> displays design remediations to address and to prevent security issues. It provides an easy and efficient way for you to communicate with personnel about the issues that your application has today and, in concise and accurate language, how to fix the design of your application.</p> <ul style="list-style-type: none"> <li>• <b>Application Tree:</b> Shows the number next to each node in the tree indicates how many remediation tasks are relevant for each item.</li> <li>• <b>Result List:</b> Shows an icon per remediation task, indicating the priority value that is assigned to this task (see “Remediation Tasks: Result List” on page 193).</li> <li>• <b>Detail Pane:</b> Shows details of the remediation task and lists the issues that will be addressed with this task.</li> </ul> <p>To learn more about the Remediations view, see Chapter 9, “Results: Remediation Tasks,” on page 193.</p>

See “Step 4: Communicating results.”

## Step 4: Communicating results

Step four of the tutorial.

### About this task

There are two basic ways of communicating scan results to your teams:

#### Reports:

You can choose a report template based on the intended report: make one report for QA, one for executives, one for developers, and so on. As you get further along the development lifecycle of your application, the compliance reports will become more and more essential. You can get a full report of how your application stands up to selected government and industry standards and regulations. See Chapter 10, “Reports,” on page 197 for details.

#### Individual issues:

If there is a specific issue that is to be addressed by a specific team or individual, you can send the tests and their results to the relevant party. See “Report false positive test results” on page 187

---

## Sample scans

The sample scans can help give you a feel for using AppScan and what scan results look like.

Four sample scans are saved to your machine when AppScan is installed. You can open them to see how they are configured, and how the results are displayed in AppScan. They can be found in your main AppScan Standard folder, whose default location is:

C:\Program Files (x86)\IBM\AppScan Standard

The scans are:

**demo.testfire.net.scan**

This is a scan of the AppScan demonstration test site. You can review the configuration and results. You can also send additional requests to the site and continue the scan with the new data.

**Glass\_Box\_DotNet\_Demo.scan and Glass\_Box\_Java\_Demo.scan**

These two scans are examples of glass box scans using a .NET application server and a Java server respectively. You can review the configurations and drill down to individual issues. For issues discovered by glass box the information includes the actual problematic source code.

**Note:** Glass box scanning requires access to an agent located on the server of the application being scanned, and you do not have access to the agent that was used for this scan, you cannot continue the scan.

**GSC\_demo.testfire.scan**

This is a web services scan of the AppScan demonstration test site. You can review the configuration and results. If you have installed GSC (Generic Service Client) you can use it to send additional requests to the site and continue the scan with the new data.

---

## Chapter 4. Configuring

You configure a scan by choosing the settings that best enable AppScan to access to your application and run tests on it.

There are two methods of configuring regular scans:

- “Scan configuration wizard” uses a wizard format and is the simplest method when:
  - You are new to AppScan, or
  - Most of the settings you require are standard, or
  - You do not need to change settings between scans, and have already saved a scan template with your own specifications
- “Scan configuration dialog box” on page 42 is the simplest method when:
  - You need to customize many settings for a scan
  - You want to create and save a custom scan template

In both cases you start with a scan template (see “Scan templates” on page 111) and can optionally change settings as necessary. (It should be noted that a number of advanced settings cannot be changed using the Scan Configuration wizard, but only using the Scan Configuration dialog box.)

You can base your scan on the **Regular Scan template**, or you can load a previously saved template of your own. (A “*scan template*” is simply a set of scan configuration settings that have been saved.) Saving and loading your own scan templates is the most efficient method when you regularly use the same settings for your scans.

**Note:** After you have configured the scan, but before you start it, you can utilize “Scan Expert” on page 134 to evaluate the configuration for your particular application, and suggest changes to optimize it.

See also: “Workflow description” on page 17

---

### Scan configuration wizard

You can quickly configure basic scans using the wizard.

The **Scan Configuration Wizard** is the simplest way to configure and start a scan when you do not need to change many of the default options. However, if you need to change advanced options, you may prefer to use the “Scan configuration dialog box” on page 42.

The wizard gives you access to the most common configuration options, the main difference being the way the options are arranged. (In both the table below and the sections following, cross-references to options that are the same in both the wizard and the dialog box point to the dialog box section of this guide.)

Explore with	Description
AppScan (automatically or manually)	Select this option for most web application scans. The application is explored manually and/or automatically with requests sent from AppScan to the application.
External device/client (with AppScan as recording proxy)	Select this option to use AppScan’s External Traffic Recorder as a recording proxy, and manually explore RESTful or other non-SOAP web services - or SOAP services that do not require security envelopes - using a mobile phone, simulator, or emulator. AppScan displays the domains and requests in its External Traffic Recorder, and create appropriate tests from the input.

Explore with	Description
Generic Service Client (WSDL)	Select this option for web services with a WSDL file. Generic Service Client (GSC) uses your web service's WSDL file to display a simple interface showing the services available, and lets you input parameters and view the results. Use the GSC interface to explore your web service manually, so that AppScan can use your input to create appropriate tests. <b>Note:</b> This option is only available if GSC (Generic Service Client) is installed on your machine. You can click the blue link to download GSC to your computer, and then double-click on the .EXE file to install it.

## Launching the Scan Configuration Wizard

### About this task

You can launch the **Scan Configuration wizard** from the AppScan Welcome Screen (which appears when you launch AppScan), or by clicking the New Scan icon on the toolbar

**Note:** If the **Welcome** screen does not appear when AppScan starts, your **Preference** settings have been changed. To change them back, see “General tab” on page 221. (You can also open the Welcome Screen at any time from the *View* menu.)



### Procedure

- To launch the wizard do one of the following:
  - On the Welcome Screen, click **Create New Scan**.
  - If AppScan is already open, click the **New Scan** icon  on the toolbar.
- In the New Scan dialog box, verify that the **Launch Scan Configuration Wizard** check box is selected.
- In the Predefined Templates list, select a scan template by clicking on it. If you do not need a *specific* template, select the **Regular Scan** template (for details see “Predefined templates” on page 111).  
The Scan Configuration Welcome appears.
- In the **Scan Configuration Wizard** Welcome screen, select the required scanning option.

Explore with	Description
AppScan (automatically or manually)	Select this option for most web application scans. The application is explored manually and/or automatically with requests sent from AppScan to the application.

Explore with	Description
External device/client (with AppScan as recording proxy)	Select this option to use AppScan's External Traffic Recorder as a recording proxy, and manually explore RESTful or other non-SOAP web services - or SOAP services that do not require security envelopes - using a mobile phone, simulator, or emulator. AppScan displays the domains and requests in its External Traffic Recorder, and create appropriate tests from the input.
Generic Service Client (WSDL)	<p>Select this option for web services with a WSDL file. Generic Service Client (GSC) uses your web service's WSDL file to display a simple interface showing the services available, and lets you input parameters and view the results. Use the GSC interface to explore your web service manually, so that AppScan can use your input to create appropriate tests.</p> <p><b>Note:</b> This option is only available if GSC (Generic Service Client) is installed on your machine. You can click the blue link to download GSC to your computer, and then double-click on the .EXE file to install it.</p>

## Wizard for exploring with AppScan

Lists the steps of this wizard.

1. "URL and Servers"
2. (Optional:) "Communication and Proxy" on page 32
3. "Login Management" on page 32
4. (Optional:) "Login Management Details" on page 33
5. "Test Policy" on page 34
6. "Complete" on page 35

### URL and Servers

Define the starting URL for the scan and any additional servers and domains that need to be included.

#### About this task

If you selected **Web Application Scan** in the Welcome step, the first step of the wizard is to define the Starting URL and server settings for the scan.

#### Procedure

1. **Starting URL:** Type the URL of your application. The scan will start at this URL.
2. To check that you have entered the Starting URL correctly, you can view the URL you have entered in the AppScan browser:
  - a. Click the **Show in Browser** icon  (This is enabled only after you have entered the URL in the text box.) The embedded browser opens to the URL you entered. (For a summary of the toolbar icons, see "Browser toolbar" on page 332.)
  - b. If necessary you can browse to a different URL and use the toolbar to set the new page as the Start URL for the scan.
  - c. Close the browser to continue with scan configuration.
3. **Case Sensitive Path:** When this check box is selected (default), links that differ from each other by case only are treated as different pages. For example, "ReadMe.as" would be considered different to "readme.as". In most cases you should check the check box for Unix-based servers, and uncheck the check box for Windows-based servers.
4. **Additional Servers and Domains:** If your application includes servers or domains other than those of the Starting URL, and your AppScan license includes them, you must add them here so that they will be included in the scan:



- a. Click the plus icon to open the Add Servers and Domains dialog box
- b. Type in the hostname or IP address of the server, or the domain name (e.g. sitetoscan.com), then click **OK**.

The new item is added to the list.

5. **I need to configure additional connectivity settings:** By default AppScan uses Internet Explorer proxy settings. Select this check box only if you want AppScan to use a different proxy. This will open an additional step, "Communication and Proxy," when you click the Next button.
6. Click **Next** to continue to the next step of the wizard.

## What to do next

"Login Management"

If you selected I need to configure additional connectivity settings", continue with "Communication and Proxy"

## Communication and Proxy

(Optional:) Configure proxy settings and server-level authentication if needed.

## About this task

By default AppScan uses Internet Explorer proxy settings. This step opens only if you select the Additional Connectivity checkbox in the URL and Servers step of the wizard, and lets you configure AppScan to use different proxy settings, or none at all.

## Procedure

1. Select one of the three proxy options:
  - **Use Internet Explorer proxy settings:** (Default) When selected AppScan will use the address and port of the Internet Explorer connection when connecting to the application.
  - **Don't use proxy:** Select this if you do not want AppScan to use proxy settings at all.
  - **Use custom proxy settings:** Select this if you want to use an address and port other than those used by Internet Explorer.
2. If you selected one of the options that uses a proxy, you can optionally set the Username, Password and Domain for AppScan to use when connecting. Click the **Configure** button to do this.
3. If your platform requires server-level authentication, set the Username, Password and Domain in the **Platform Authentication** area.
4. Click **Next**.

## What to do next

"Login Management"

## Login Management

Configure the login method, and record the login procedure if needed.

## About this task

The Login Management step of the wizard lets you select one of three methods for AppScan to use when it encounters login pages during a scan:

- **Recorded Login:** If you select this option, AppScan uses a login procedure that you record, completing fields and clicking on links like a real user. This is the recommended login method.  
For more details see "Recording a login" on page 47.

If login requires human interaction each time (such as Two-Factor Authentication, One-Time Passwords, or CAPTCHA), select the **Prompt** option.

- **Prompt:** In this case you must still record a login procedure. Although AppScan will not attempt to log in using the procedure you record, it needs the procedure as a reference to know when it is logged out.
- **Automatic Login:** If AppScan will be able to log in to the site using a name and password only, without a special procedure, select this option and enter the Username and Password.
- **No Login:** Select this option only if the application does not require logging in, or if for some other reason you do not want AppScan to log in.

## Procedure

1. Select the radio button for the required Login method.
2. Do one of the following:
  - For Recorded Login or Prompt, click **Record** or **Import** set the a login sequence (see “Login / Logout tab” on page 46 for details).
  - For Automatic Login, simply type in **Username** and **Password**.

**Note:** If you record a login, when you finish recording a dialog box may open asking you to confirm that the login data AppScan extracted is correct. Fill in or correct the parameters and values as necessary, then click **OK**.

If you have recorded a valid login sequence *and In-Session Detection is active*, the key icon turns green, indicating that the an in-session page has been identified.

This:  changes to this: 

**Note:** If the key icon turns red , AppScan attempted to but was unable to identify any pattern in the in-session page that it can use during scanning to verify that it has not been logged out. To remedy this you should open the extra step of the wizard and either provide AppScan with an identifier manually, or deactivate In-Session Detection. (See next step.)

3. If you select the **I want to configure In-Session Detection Options** check box, an additional wizard step, “Login Management Details,” will open when you click **Next**. Select this only if you need to edit the login sequence or activate/deactivate In-Session detection (see note to previous step).
4. Click **Next**.

## What to do next

“Test Policy” on page 34

If you selected “I want to configure In-Session Detection Options”, continue with “Login Management Details”

## Login Management Details

(Optional:) Review and edit the login sequence you recorded.

## About this task

The additional Login Management settings dialog box lets you review and edit the login sequence that you recorded, and the pattern that AppScan uses during scanning to verify that it is logged in (“in-session detection pattern”).

## Procedure

1. Review and edit the login sequence using the buttons:
  - To Import a saved sequence, click **Import**.

- To view any URL in the AppScan browser, select it and click
  - To discard this sequence and record a new one, click
  - To delete redundant URLs from the login sequence, select a URL and click
2. To activate/deactivate In-Session Detection, select/deselect the check box (see “Changing the in-session detection pattern” on page 54).
  3. If you have activated In-Session Detection, you should verify that the In-Session pattern AppScan chose is correct, and if it is not, select a new one (see “Changing the in-session detection pattern” on page 54).
  4. Click **Next**.

## What to do next

“Test Policy”

### Test Policy

Limiting the scan to the specific types of tests you want can reduce scan time.

### About this task

The number of tests which AppScan sends during a scan can reach the thousands. Sometimes it is preferable to reduce scan time by limiting the scan to certain types only. This is Test Policy.

AppScan comes with a Default Test Policy, and with some additional Test Policy configurations that you can select. You can also use your own User-Defined Test Policies.

The Test Policy step of the wizard shows the name of the Test Policy that the current policy is based on, and its description.

### Procedure

1. Check that the Test policy is appropriate for your needs. (If you are in doubt, leave the Default Test Policy.)
2. To load a different Test Policy, click on one of the Pre-Defined Policies or Recent Policies in the **Policy Files** pane. For details see “Test Policy view” on page 87.
3. **Send tests on login and logout pages:** By default, AppScan will test your login and logout pages along with the rest of the application. You should leave this default configuration, unless:
  - Your application has safeguards that lock out users who provide illegal input on these pages, or
  - Your application flow would be altered if these pages were tested
 If you are unsure how your application will respond to these tests, leave this option selected.
4. **Do not send session identifiers when testing login pages:** (This check box is active, and selected by default, only if the previous check box is selected.) It is recommended to leave this check box selected, since session identifiers could limit test success when testing login pages. Clear it only if you are sure that valid session tokens are necessary to test your login pages.
   
If you are unsure how your application will respond, leave this option selected.
5. Click **Next** to proceed to the final stage of the wizard.

## What to do next

“Complete” on page 35

## **Complete**

Having completed the steps of the wizard, you now decide how and when to start the scan that you have configured.

### **About this task**

If you select one of the options that starts a scan right away, you can also elect to run *Scan Expert* before the main scan starts. Scan Expert logs into your application and performs a short, preliminary scan to evaluate the settings you have configured. It then suggests configuration changes if necessary.

Incorporating these suggestions (automatically or manually) can greatly increase the efficiency of the main scan.

### **Procedure**

1. Select one of the options:

- **Start a full automatic scan:** Start a full scan of the application (Explore followed immediately by Test).
- **Start with automatic Explore only:** Explore the application, but do not proceed to the Test stage. (You can run the Test stage later.)
- **Start with Manual Explore:** The browser will open and you can manually explore your site by clicking on links and completing fields. AppScan will record the results to use in the Test stage.
- **I will start the scan later:** Close the wizard without starting a scan. The next time you start a scan, this template will be used.

2. **Start Scan Expert when Scan Configuration is Complete:** (Active only if one of the first three scan options has been selected.) Select this check box in you want Scan Expert to evaluate your configuration before starting the main scan.

### **What to do next**

*See:* “Scan progress” on page 129

*See also:*

“Starting scans” on page 127

Chapter 8, “Results: Security Issues,” on page 173

## **Wizard for exploring with an external device or client**

Lists the steps of this wizard.

1. “Recording Proxy”
2. (Optional:) “Communication and Proxy” on page 36
3. “SSL Certificate” on page 36
4. “Login Management” on page 37
5. (Optional:) “Login Management Details” on page 38
6. “Test Policy” on page 38
7. “Complete” on page 39

### **Recording Proxy**

Configure proxy port and client type.

## Procedure

1. **Recording Proxy port:** Configure the port on which AppScan will receive the Manual Explore requests you send from your remote device or external client. You can let AppScan choose an available port automatically.

**Tip:** It is better to choose the port yourself, as the automatically chosen port may change from session to session. However, you must not specify a port if you will be opening more than one instance of AppScan at the same time; if you need to use more than one instance, configure AppScan to choose the port automatically.

2. **Record from:** Define whether you will be recording from an external client on the same machine as AppScan (such as a simulator or emulator), or a remote device (such as a mobile phone).
3. Configure your remote device or external client to use the IP and port shown in the information pane.
4. **I need to configure additional connectivity settings:** By default AppScan uses Internet Explorer proxy settings. Select this check box only if you want AppScan to use a different proxy. This will open an additional step when you click the Next button.
5. Click **Next** to continue to the next step of the wizard.

## What to do next

"SSL Certificate"

If you selected "I need to configure additional connectivity settings", continue to "Communication and Proxy"

## Communication and Proxy

(Optional:) Configure proxy settings and server-level authentication if needed.

## Procedure

1. Select one of the three proxy options:
  - **Use Internet Explorer proxy settings:** (Default) When selected AppScan will use the address and port of the Internet Explorer connection when connecting to the application.
  - **Don't use proxy:** Select this if you do not want AppScan to use proxy settings at all.
  - **Use custom proxy settings:** Select this if you want to use an address and port other than those used by Internet Explorer.
2. If you selected one of the options that uses a proxy, you can optionally set the Username, Password and Domain for AppScan to use when connecting. Click the **Configure** button to do this.
3. If your platform requires server-level authentication, set the Username, Password and Domain in the **Platform Authentication** area.
4. Click **Next**.

## What to do next

"SSL Certificate"

## SSL Certificate

If your server uses HTTPS, you need to add the AppScan SSL root certificate (so that requests sent using AppScan as proxy will be accepted).

## Procedure

1. Click **Install AppScan certificate locally**, and then click **Yes** on the Windows dialog box that appears to allow the change.

**Note:** Once the certificate is installed, the button is replaced with a confirmation message, and the button will not appear the next time you use the wizard (unless the certificate is uninstalled).

**Note:** To uninstall, go to **Tools > Options > Recording Proxy**, and click **Remove**.

2. If you will be exploring from a remote device (mobile phone or emulator), you must also install the certificate on that device:
  - a. On the device, browse to: `http://appscan`
  - b. On the device, click **Install AppScan SSL Certificate**.

You are now ready to record the login from your device, with AppScan as recording proxy.

3. Click **Next**.

## What to do next

“Login Management”

### Login Management

Record the login procedure.

### Before you begin



The orange key icon indicates that the login has not yet been recorded.

### Procedure

1. On your device, browse to your application's login page.
2. In AppScan, click The External Login Recorder opens, and should indicate that it is connected to your external device.
3. On your device, log into the application.  
The login requests are listed in the External Login Recorder.
4. Once you are logged in to your application, click **Stop recording** (in the External Login Recorder).
5. Optionally review the list for unnecessary requests (for example, requests to a different domain), and delete them by selecting and clicking
6. Click **OK** to close the recorder.



The green key icon indicates that in-session status has been detected.

7. If you select the **I want to configure In-Session Detection Options** check box, an additional wizard step, “Login Management Details” on page 38, will open when you click **Next**. Select this only if you need to edit the login sequence or activate/deactivate In-Session detection (see note to previous step).
8. In the wizard, click **Next**.

Related topic:

“External Login Recorder” on page 123

## What to do next

“Test Policy” on page 38

If you selected “I want to configure In-Session Detection Options”, continue to “Login Management Details” on page 33

## Login Management Details

(Optional:) Review and edit the login sequence you recorded.

### Procedure

1. Review and edit the login sequence using the buttons:
  - To Import a saved sequence, click **Import**.
  - To view any URL in the AppScan browser, select it and click 
  - To discard this sequence and record a new one, click 
  - To delete redundant URLs from the login sequence, select a URL and click 
2. To activate/deactivate In-Session Detection, select/deselect the check box (see “Changing the in-session detection pattern” on page 54).
3. If you have activated In-Session Detection, you should verify that the In-Session pattern AppScan chose is correct, and if it is not, select a new one (see “Changing the in-session detection pattern” on page 54).
4. Click **Next**.

### What to do next

“Test Policy”

### Test Policy

Limiting the scan to the specific types of tests you want can reduce scan time.

### About this task

The number of tests which AppScan sends during a scan can reach the thousands. Sometimes it is preferable to reduce scan time by limiting the scan to certain types only. This is Test Policy.

AppScan comes with a Default Test Policy, and with some additional Test Policy configurations that you can select. You can also use your own User-Defined Test Policies.

The Test Policy step of the wizard shows the name of the Test Policy that the current policy is based on, and its description.

### Procedure

1. Check that the Test policy is appropriate for your needs. (If you are in doubt, leave the Default Test Policy.)
2. To load a different Test Policy, click on one of the Pre-Defined Policies or Recent Policies in the **Policy Files** pane. For details see “Test Policy view” on page 87.
3. **Send tests on login and logout pages:** By default, AppScan will test your login and logout pages along with the rest of the application. You should leave this default configuration, unless:
  - Your application has safeguards that lock out users who provide illegal input on these pages, or
  - Your application flow would be altered if these pages were testedIf you are unsure how your application will respond to these tests, leave this option selected.
4. **Do not send session identifiers when testing login pages:** (This check box is active, and selected by default, only if the previous check box is selected.) It is recommended to leave this check box selected, since session identifiers could limit test success when testing login pages. Clear it only if you are sure that valid session tokens are necessary to test your login pages.  
If you are unsure how your application will respond, leave this option selected.

5. Click **Next** to proceed to the final stage of the wizard.

## What to do next

“Complete”

### Complete

AppScan is now configured as a recording proxy for your device to send requests to the application. You are now ready to start your Manual Explore of the application.

### Procedure

1. Click **Finish**.

The External Traffic Recorder opens with a status message indicating that it is connected to your device.

2. When the External Traffic Recorder opens with status "Waiting for incoming connections, manually Explore the web service from your device/application:

- a. Using your device or application, explore the web service.

As you explore, domains detected are listed in the left pane of the recorder, and URLs are listed in the right pane.

- b. When finished, in AppScan click **Stop Recording**.

3. Review and edit the Manual Explore data:

#### Domains detected

All domains to which requests were sent are listed, and by default selected for adding to the list of Additional Servers and Domains (Configuration > URLs and Servers > Additional Servers and Domains) so they can be included in the scan. You can deselect any you do not want included in the scan.

**Tip:** You should deselect any domains that belong to other companies.

#### Requests sent

All requests sent by the device to domains *that are selected* (in the left pane), are listed. If you select/clear domains in the left pane, the requests list is updated. You can delete specific requests if they are not needed.

**Tip:** If the total number of filtered requests is more than 200, deleting some of them may produce a more efficient scan.

**Note:** At this stage you can click **Export** to save the Explore data for use on another machine.

4. Click **OK** to close the recorder.

AppScan takes a few moments to process and display the data.

5. To start the Test stage, click **Scan > Test Only**

The Test stage starts and when complete the scan results are displayed.

Related topics:

- “Recording Proxy tab” on page 219
- Chapter 8, “Results: Security Issues,” on page 173

## Wizard for exploring with GSC

Lists the steps of this wizard.

1. “URL and Servers” on page 40
2. (Optional:) “Communication and Proxy” on page 40

3. "Test Policy" on page 41
4. "Complete" on page 41

## URL and Servers

Supply the URL of the WSDL file for the service, and define any additional servers and domains to include in the scan.

### Procedure

1. **WSDL URL:** Type in the URL of the WSDL file for the service.  
Example: <http://www.sitetoscan.com/Service1.asmx?wsdl>
2. To check that you have entered the URL correctly, you can open the URL you have entered in the AppScan browser:
  - a. Click  (This button is enabled only after you have entered the URL in the text box.) The embedded browser opens to the URL you entered.
  - b. If necessary correct the URL.
  - c. Close the browser to continue with scan configuration.
3. **Case Sensitive Path:** When this check box is selected (default), links that differ from each other by case only are treated as different pages. For example, "ReadMe.as" would be considered different to "readme.as". In most cases you should check the check box for Unix-based servers, and uncheck the check box for Windows-based servers.
4. **Additional Servers and Domains:** If your application includes servers or domains other than those of the Starting URL, and your AppScan license includes them, you must add them here so that they will be included in the scan:
  - a. Click  icon to open the *Add Servers and Domains* dialog box.
  - b. Type in the hostname or IP address of the server, or the domain name (e.g. sitetoscan.com), then click **OK**

The new item is added to the list.

5. **I need to configure additional connectivity settings:** By default AppScan uses Internet Explorer proxy settings. Select this check box only if you want AppScan to use a different proxy. This will open an additional step when you click the Next button.
6. Click **Next** to continue to the next step of the wizard.

### What to do next

"Test Policy" on page 41

or, if you selected the **I need to configure...** check box:

"Communication and Proxy"

## Communication and Proxy

(Optional:) Configure proxy settings and server-level authentication if needed.

### About this task

By default AppScan uses Internet Explorer proxy settings. The additional URL and Servers dialog box lets you configure AppScan to use different proxy settings, or none at all.

### Procedure

1. Select one of the three proxy options:

- **Use Internet Explorer proxy settings:** (Default) When selected AppScan will use the address and port of the Internet Explorer connection when connecting to the application.
  - **Don't use proxy:** Select this if you do not want AppScan to use proxy settings at all.
  - **Use custom proxy settings:** Select this if you want to use an address and port other than those used by Internet Explorer.
2. If you selected one of the options that uses a proxy, you can optionally set the Username, Password and Domain for AppScan to use when connecting. Click the **Configure** button to do this.
  3. If your platform requires server-level authentication, set the Username, Password and Domain in the **Platform Authentication** area.
  4. Click **Next** to proceed to the Test Policy Settings step.

## What to do next

"Test Policy"

### Test Policy

Define which kinds of tests AppScan will send to the service when your Manual Explore is complete.

#### Procedure

1. In the **WSDL Scan Configuration Wizard > Platform Authentication** dialog box, click **Next**.  
The **WSDL Test Policy** step of the wizard appears (same as "Test Policy" on page 34).
2. Check that the Test Policy is appropriate for your needs. (Unless you have defined a custom policy, the web Services Test Policy is recommended.)
  - To load a different Test Policy, click **Load**.  
The **Load Test Policy** dialog box opens, letting you select one of the pre-defined or user-defined policies.
  - To edit the current Test Policy, click **Edit**.  
The **Test Policy Manager** opens, letting you configure exactly which tests are included in the scan.  
For details see "Importing a Test Policy" on page 90
3. If necessary you can click the **Advanced** Test Settings button to configure additional options. For details see "Importing a Test Policy" on page 90. Otherwise, click **Next** to proceed to the final stage of the wizard.

## What to do next

"Complete"

### Complete

You are now ready to start manually exploring the service using GSC.

#### Procedure

Click **Finish**.

A prompt that Generic Service Client (GSC) is about to open may appear (if so, click **OK**), and then the program opens.

## What to do next

"Using GSC" on page 124

# Scan configuration dialog box

## About this task

The Scan Configuration dialog box provides many options for configuring your scans. The main options are also available via the Scan Configuration Wizard, but using the dialog box is recommended when you want to change many of the default settings, or tweak an existing configuration.

## Procedure

To open the Scan Configuration dialog box, click the **Configuration** icon  on the toolbar (or press **F10**).

The Scan Configuration dialog box has different views, divided into four groups, that are accessed by clicking the relevant item in the view selection pane, on the left-hand side.

**Note:** If you stop a scan and change the configuration, changes will not affect requests already sent. To apply changes to the entire scan you must start a new scan.

**Tip:** You can change configuration options in more than one of the views, and then click **OK** to save all changes. (Your changes are kept as you navigate between the views, but only saved when you click **OK**.)

View	Select to configure:
<i>Explore</i>	
"URL and Servers view" on page 43	Starting URL, system type, and additional servers
"Login Management view" on page 46	Set login method, record a login sequence (optional), and configure in-session detection
"Environment Definition view" on page 56	Provide information about the application environment
"Exclude Paths and Files view" on page 56	Paths and file types to exclude from the scan
"Explore Options view" on page 62	Scan limits, link extraction methods and general Explore method
"Parameters and Cookies view" on page 65	Identify session IDs and list parameters to exclude from the scan
"Automatic Form Fill view" on page 75	Provide AppScan with valid parameter values for filling forms
"Error Pages view" on page 77	Add strings, regexps and URLs to identify custom error pages
"Multi-Step Operations view" on page 79	Record and manage multi-step operations that are required to reach parts of the application
"Content-Based Results view" on page 82	For applications that do not have a hierarchical URL structure, such as single entry-point applications, define how AppScan arranges the site tree.
"Glass Box view" on page 85	If the AppScan glass box agent is installed on your application server, configure glass box scanning here.
<i>Connection</i>	
"Communication and Proxy view" on page 86	Configure communication timeout and proxy server settings
"HTTP Authentication view" on page 86	Add server-level authentication and client-side certificates, if required by the application

<b>View</b>	<b>Select to configure:</b>
<i>Test</i>	
"Test Policy view" on page 87	Define and edit test policy (which tests are sent to the application)
"Test Options view" on page 92	Additional test options
"Privilege Escalation view" on page 95	Refer AppScan to scans run using different user privileges, to discover privileged resources that are available to users with insufficient privileges
"Malware view" on page 96	Test for malicious links.
<i>General</i>	
"Scan Expert view" on page 96	Configure Scan Expert behavior and modules
"Result Expert view" on page 98	Configure Result Expert behavior and modules
"Advanced Configuration view" on page 98	Configure advanced scan options

## URL and Servers view

URL and Servers view of the Scan Configuration dialog box.

You *must* define the URL from which to start the scan. The remaining settings are optional.

<b>Setting</b>	<b>Details</b>
Starting URL	<p>Provide the URL of your application to be the starting point of the scan.</p> <p>You can view the URL in the browser, to verify that it is the correct one, or that AppScan can successfully connect to your application. To do this click the <b>Show in Browser</b> icon .</p> <p>For details see "Starting URL" on page 44</p>
Scan only links in and below this directory.	<p>When selected, the scan is limited to pages below the starting URL. Links to other URLs will not be scanned, even if they are included in the list of Additional Servers and Domains (below).</p> <p>For details see "Limiting scan to the Starting URL folder" on page 44</p>
Case Sensitive Path	<p>When selected, links that differ from each other by case will be considered different pages. For example, "ReadMe.as" would be considered different to "readme.as".</p> <p>When deselected, all URLs appear in lowercase.</p> <p>Select this check box if the file system on your application host is case sensitive. (In most cases, you will want to check the check box for Unix-based servers, and uncheck the check box for Windows-based servers.)</p>
Additional Servers and Domains	<p>If your application includes links to domains other than the domain of the Starting URL, you must add them here in order for them to be included in the scan.</p> <p><b>Note:</b> You can only add servers/domains that are included in your AppScan license.</p> <p>For details see "Additional Servers and Domains" on page 46</p>

## Starting URL

The Starting URL is the location at which AppScan starts to scan.

“Viewing the Starting URL in a browser”

“Changing the Starting URL”

### Viewing the Starting URL in a browser:

#### About this task

You can view the URL in a browser, to verify that it is the correct one, or that AppScan can successfully connect to your application.

#### Procedure

Next to the URL field, click  (enabled only after you have entered the URL in the text box.) The embedded browser appears. (For a summary of the browser icons see “Browser toolbar” on page 332.)

### Changing the Starting URL:

#### About this task

If the displayed page is not the URL you want to use as the starting point of the scan, you can choose a different URL.

#### Procedure

1. Browse to the page you want the scan to start from and click the **Set as Starting URL**  icon.
2. Close the browser.

#### Results

The new starting URL appears in the text box of the **Scan Configuration** dialog box.

## Limiting scan to the Starting URL folder

You can easily limit a scan to within and below the folder of the Starting URL.

#### About this task

The check box under the Starting URL field automatically creates the necessary filters to limit your scan to within and below a particular directory.

#### Procedure

1. Open **Scan Configuration > URL and Servers**.
2. Type or paste-in the URL of the directory you want to restrict the scan to.
3. Select the **Scan only links in and below this directory** check box.

The scan will now be limited to paths under this URL. Links outside this range will not be scanned.

#### Example

If the Starting URL is defined as `http://main/bank/`

- Links to the following will be scanned:
  - `http://main/bank/transfer.aspx/`

- [http://main/bank/transfer/page\\_1.aspx/](http://main/bank/transfer/page_1.aspx/)
- Links to the following will be ignored:
  - <http://main/transfer.aspx/>
  - [http://main/transfer/page\\_1.aspx/](http://main/transfer/page_1.aspx/)

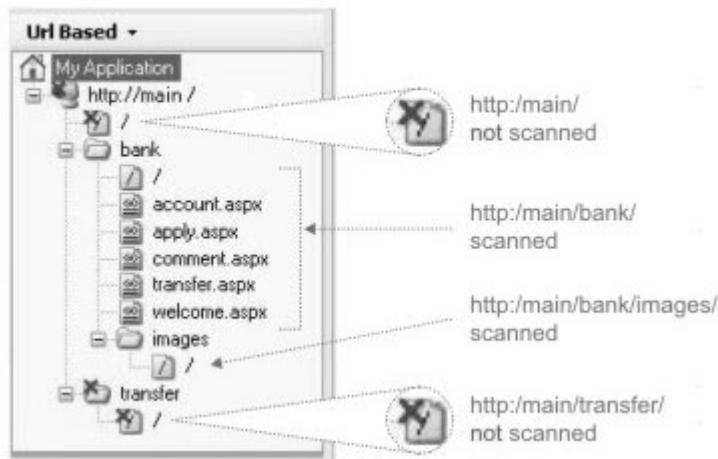
When you configure the Starting URL, and select this check box, the following two items are added at the top of the Exclude Paths and Files table (Scan Configuration > Exclude Paths and Files):

Item Type	Path	Matching	Behavior
Exclude (Start URL)	.*	Regular Expression	Always first in the table. Cannot be demoted. Cannot be edited or deleted. However, if the next item (Exception) is deleted, this is deleted too.
Exception (Start URL)	http://main/bank	Full Path	Always second in the table. Cannot be promoted or demoted. <i>Can be edited. (This is to allow for editing the exception in rare cases where AppScan misidentifies the Starting URL directory.)</i> If deleted, the previous item (Exclude) is also deleted, and the Starting URL check box is deselected.

**Note:** Unlike other exclusions and exceptions, these two have a gray background, indicating their special status.

When the scan starts, and the Application Tree fills in the left pane, links to parts of the application that are not below the Starting URL are shown with a red X, indicating that they were not scanned.

Starting URL: <http://main/bank/>



## What to do next

You can verify that the Exception added is correct, or edit it, from **Exclude Paths and Files** view (see "Limiting scan to a specific folder" on page 60).

## Additional Servers and Domains

### About this task

If your application includes links to domains other than the domain of the Starting URL, you must add them here in order for AppScan to include them in the scan.

**Note:** You can only add servers/domains that are included in your AppScan license.

### Procedure

- To add a server, click  and type in the server/domain name.  
Example: demo.testfire.net or 65.61.137.117  
The server is added to the list.
- To edit a server in the list, select it and click 
- To delete a server from the list, select it and click 

## Login Management view

Login Management view of the Scan Configuration dialog box.

**Login Management** view of the **Scan Configuration** dialog box is used for configuring how AppScan logs in to your application, and how it recognizes when it has been logged out.

AppScan can automatically detect login requests and fills in the username and password parameters. If your application has a non-standard login sequence of actions, you can record these actions for AppScan to use.

Login Management view contains three tabs:

- Login / Logout
- Details
- Session IDs

### Login / Logout tab

Scan Configuration > Login Management > Login / Logout tab.

The **Login Management** view of the **Scan Configuration** dialog box lets you define how AppScan logs in to your application, and how it recognizes when it has been logged out.

AppScan can automatically detect login requests and fills in the username and password parameters. If your application has a non-standard login sequence of actions, you can record these actions for AppScan to use.

Login Management view contains three tabs: Login/Logout, Details, and Session IDs. This section deals with the Login/Logout tab. For the other tabs, see “Details tab” on page 49, and “Session IDs tab” on page 55.

Setting	Details
Recorded (Recommended)	<p>(Default method) Select this method to open the browser and record a login sequence (both HTTP requests and user actions are recorded). AppScan will use this sequence whenever it needs to login to the application.</p> <p> <b>Record</b> is used to record the sequence. For web applications, see “Recording a login”; for RESTful (and other) web services, see “Exploring with the External Traffic Recorder” on page 121</p> <p>You can also <b>Import</b> a sequence that was previously saved (see “Importing a Login Sequence” on page 53), <b>Edit</b> the current sequence (see “Viewing URL Request/Response details” on page 53), and <b>Export</b> the sequence to use with other scans (see “Exporting a Login Sequence” on page 53).</p>
Prompt	<p>Select this method if login requires human interaction each time (such as Two-Factor Authentication, One-Time Passwords, or CAPTCHA).</p> <p>Note that even if you select this option, <b>you must record a login sequence</b>. This is to provide AppScan with an in-session page that it can later use to verify that it is logged-in. For details see “Recording a login”</p>
Automatic Login	Select this method to let AppScan automatically detect the login form of your application and use the username and password you supply. (This method can be less reliable than the Recorded Login method.)
None	Select this option if the application does not require users to log in.
Key icon	<p>The key icon indicates the status of In-Session Detection:</p> <p> Enabled and configured. (An in-session page has been identified in login sequence, either automatically or by the user.)</p> <p> Enabled but not fully configured.</p> <p> Enabled but not configuration failed.</p> <p> Disabled.</p> <p>See “Changing the in-session detection pattern” on page 54 for details.</p>
Concurrent Logins	(Selected by default) To save scan time, AppScan sends multiple login requests without logging out in between. Deselect this check box only if your application does not allow this.
Logout Page Detection	<p>AppScan uses a regexp to identify log out pages. This helps AppScan to scan more efficiently by trying to avoid getting logged out too often, and having to log in again. It is also used to identify logout pages when you configure the scan to not test login/logout pages (see “Test Options view” on page 92), and to logout when needed as a part of some security tests.</p> <p>You can edit this regexp as required.</p> <p>See “Logout page detection” on page 49</p>

## Recording a login:

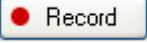
## Before you begin

Before you can record a login sequence the Starting URL must be defined (either in Configuration > URL and Server view - see "Starting URL" on page 44 - or in the previous step of the wizard).

## About this task

Recorded Login lets you teach AppScan the procedure for logging in to your site: which links to click, which text to input in forms, and the order in which to do them. As soon as you have recorded this, AppScan will attempt to identify an in-session pattern that it can use in future to verify that it is logged in. Once this is done, AppScan can use the login sequence to log itself back in during the scan, whenever it detects that it has been logged out.

### Procedure

1. In Scan Configuration > Login Management > Login / Logout view, select Recorded.
2. Click  **Record** > Use AppScan browser

The AppScan browser opens to the scan's Starting URL. The browser's **Record** button  is pressed (grayed out), indicating the recording has begun. All actions you now take, until you close the browser are recorded by AppScan.

**Note:** If the Starting URL has not yet been defined you are warned that you must define it before you can proceed (see "URL and Servers view" on page 43).

**Note:** If a login procedure has previously been recorded, you are warned that the new recording will overwrite the existing one.

**Note:** If you have configured AppScan to use an external browser for scanning (Tools > Options > Use External Browser > Select Browser), you will be given the choice of recording the login with either the AppScan browser or the external browser. If possible it is recommended to use the AppScan browser for the login recording (even if using a different browser for scanning), as it records extra information that improves login success during scanning. If recording the login with the AppScan browser does not work for your application, use the external browser.

**Note:** If your application does not support Internet Explorer you *must* configure AppScan to work with an external browser.

3. Perform the login procedure, completing forms and clicking on links as necessary, to log AppScan into the site.

**Tip:** By default, the page you reach when you have logged in will be used by AppScan as the in-session URL. AppScan sends this URL every few seconds during the scan, to check that it is still logged in. If the page sends a large response, or if it includes tracked parameters or cookies, you can improve scan performance by clicking on one or more additional links until you reach a page with a smaller response (while still logged in) and without tracked parameters or cookies. Then, after you close the browser, go to the **Details tab** and select the *later* page as the "in-session URL". For more details see "In-Session Detection" on page 53.

4. When you have successfully logged in to the site close the browser.

**Note:** When you close the browser, AppScan attempts to extract the login information from your login request, for use during scanning. If the login mechanism manipulates the login data using JavaScript, a dialog box may open asking you to confirm that the login data AppScan extracted is correct. Fill in or correct the parameters and values as necessary, then click **OK**.



The Session Information dialog box opens displaying the login requests you recorded, and



changes to , indicating that in-session detection is active.



**Note:** If the key icon turns red , AppScan attempted but was unable to identify any pattern in the in-session page that it can use during scanning to verify that it has not been logged out. If this happens, you need to identify the "in-session pattern" for AppScan. See "Changing the in-session detection pattern" on page 54 for details. In some cases a more specific message may appear, with a link to a page in this Help for troubleshooting the problem.

5. To make changes to the recorded sequence (for example to remove unnecessary steps), refer to "Details tab."

**Note:** Generally speaking the URL which logs the user in (and whose response is the first to include an in-session pattern), should be the one marked In-Session. However, sometimes you may want to select a later URL, that also includes the in-session pattern, but which has the advantage of being a smaller page or of not including tracked parameters or cookies. Additionally, sometimes the POST request with the user credentials is the request which logs you in and first contains the in-session pattern, this is a poor choice for the in-session page, since the in-session check would send the credentials each time, leading to a false positive in session response. See "Optimizing In-Session Detection" on page 54

6. To save the new login sequence (overwriting any previously recorded login sequence), click **OK**.

**Note:** If you are sure that the in-session page contains no tracked parameters or cookies, you can improve scan performance by changing the **Advanced Configuration > Session Management: Parse in-session page** setting to "False". See "Advanced Configuration view" on page 98.

**Logout page detection:** In order to verify that it is logged in to the application, AppScan checks for logout links in the responses it receives. By default AppScan uses the following regular expression:  
(logout|signout|logoff|signoff|exit|quit|invalidate)

as its *Logout Page Detection Pattern*. If any of the indicators in this regexp appear in the URL, AppScan assumes the page is a logout page, and therefore that it is currently logged in to the application.

The Logout Page Detection Pattern appears in the lower half of the **Login/Logout** view of the **Scan Configuration** dialog box. Additional indicators can be added to this regular expression as necessary. Be sure to follow the regular expression syntax rules.

**Note:** The Expression Test PowerTool (**Tools > Expression Test**) can be useful to verify the syntax of your regular expressions. If you need additional help you may find the following link useful:  
<http://www.regular-expressions.info/quickstart.html>

## Details tab

Scan Configuration > Login Management > Details tab.

When you record a login sequence, AppScan records both the actions and the requests. These are shown on the two tabs of the Details tab: Actions and Requests. When replaying the login AppScan attempts (by default) to reproduce the action-based login; if this is unsuccessful it uses the request-based login.

This tab is used to:

- Review and edit the action-based and requests-based versions of the login sequence.
- Activate/deactivate In-Session Detection

Setting	Details				
Use action-based login when possible	(Selected by default) When selected, AppScan will attempt to log in using action-based login, replaying the clicks and keystrokes of the user. If this method fails for any reason, AppScan will use the request-based version, which re-sends the raw HTTP requests from the login recording.				
Login Sequence	<p>Shows the sequence in two tabs: Actions and Requests. Each tab also indicates whether the current recorded version is valid  or invalid  (sometimes AppScan may be unable to validate one of the versions).</p> <p>URLs of the login sequence, and their type. Options are: Login, In-Session, Ignore.</p> <p> <b>Records a new sequence</b>, overwriting the previous one.</p> <table border="1"> <thead> <tr> <th>Actions tab</th> <th>Requests tab</th> </tr> </thead> <tbody> <tr> <td> <p>A list of recorded actions.</p> <ul style="list-style-type: none"> <li>Double-click on input items to view passwords and edit input.</li> </ul> <p>For details, see “Actions” on page 51</p> <p> <b>Opens the action-based player</b> and plays through the recorded actions. The left pane of the player shows the actions as a list, as the player plays them, and the right pane shows the browser view.</p> <p><b>Note:</b> If one of the login pages is very slow, it may be more efficient to use request-based login, as many logins are typically required during a scan.</p> </td> <td> <p>A list of recorded requests.</p> <ul style="list-style-type: none"> <li>Double-click on individual requests to open them in the browser.</li> <li>Right-click on the Type to change. Options are Login (part of the process), In-session (the request which results in an in-session response), and Ignore (requests after the in-session request).</li> </ul> <p>For details, see “Requests” on page 52, and “Changing the in-session detection pattern” on page 54</p> <p> <b>Opens the browser</b> on the selected page of the sequence.</p> </td></tr> </tbody> </table> <p>Generally speaking the URL which logs the user in (and whose response includes an in-session pattern), should be In-Session. However, sometimes you may want to select a later URL that also includes the in-session pattern but has the advantage of being a smaller page.</p> <p> <b>Validate</b> becomes active if you make any changes to the sequence. Click to validate the new sequence.</p> <p> <b>Exports</b> the sequence for use with other scans.</p> <p> <b>Imports</b> a sequence that was previously exported.</p> <p> <b>Edits</b> input such as username and password in the sequence.</p> <p> <b>Deletes</b> the selected page from the sequence.</p>	Actions tab	Requests tab	<p>A list of recorded actions.</p> <ul style="list-style-type: none"> <li>Double-click on input items to view passwords and edit input.</li> </ul> <p>For details, see “Actions” on page 51</p> <p> <b>Opens the action-based player</b> and plays through the recorded actions. The left pane of the player shows the actions as a list, as the player plays them, and the right pane shows the browser view.</p> <p><b>Note:</b> If one of the login pages is very slow, it may be more efficient to use request-based login, as many logins are typically required during a scan.</p>	<p>A list of recorded requests.</p> <ul style="list-style-type: none"> <li>Double-click on individual requests to open them in the browser.</li> <li>Right-click on the Type to change. Options are Login (part of the process), In-session (the request which results in an in-session response), and Ignore (requests after the in-session request).</li> </ul> <p>For details, see “Requests” on page 52, and “Changing the in-session detection pattern” on page 54</p> <p> <b>Opens the browser</b> on the selected page of the sequence.</p>
Actions tab	Requests tab				
<p>A list of recorded actions.</p> <ul style="list-style-type: none"> <li>Double-click on input items to view passwords and edit input.</li> </ul> <p>For details, see “Actions” on page 51</p> <p> <b>Opens the action-based player</b> and plays through the recorded actions. The left pane of the player shows the actions as a list, as the player plays them, and the right pane shows the browser view.</p> <p><b>Note:</b> If one of the login pages is very slow, it may be more efficient to use request-based login, as many logins are typically required during a scan.</p>	<p>A list of recorded requests.</p> <ul style="list-style-type: none"> <li>Double-click on individual requests to open them in the browser.</li> <li>Right-click on the Type to change. Options are Login (part of the process), In-session (the request which results in an in-session response), and Ignore (requests after the in-session request).</li> </ul> <p>For details, see “Requests” on page 52, and “Changing the in-session detection pattern” on page 54</p> <p> <b>Opens the browser</b> on the selected page of the sequence.</p>				

Setting	Details
In-Session / Out-of-Session Detection Pattern	<p>Shows a pattern in the login sequence which indicates that the user is in-session (or out-of-session). It is used only when the Activate Session Detection checkbox is selected.</p> <p><b>Note:</b> It is usually preferable to use an in-session (rather than an out-of-session) pattern. However, in rare cases where the in-session pattern is not always returned following an in-session request, or where it is complicated to define an in-session pattern, you can define an out-of-session pattern instead.</p> <p>Whenever the defined pattern is found, AppScan will know that it is out-of-session.</p> <p><b>RegExp:</b> Select this check box to enter a regular expression as the pattern.</p> <p> opens the Expression Test PowerTool to verify the regular expression.</p>
Activate Session Detection	<p>(Activated by default if an In-session response has been detected in the login sequence.) The field shows the text that indicates the In-session (or out-of-session) status. You can edit this pattern by entering text in the Response Pattern field, or clicking <b>Select In-session pattern</b> to select the pattern in the browser or from the raw HTML.</p> <p><b>Important:</b> Disabling in-session detection when the site contains a web login can lead to very big gaps in scan coverage, and cause inaccurate results.</p> <p><b>Note:</b> If you deselect this checkbox, the entire login sequence will be played by default whenever replaying multi-step sequences. To avoid this happening you can set <b>Advanced Configuration &gt; Session Management: In-session before multi-step operations</b> to False.</p>

## Actions:

The Actions tab shows the login sequence you recorded, as a sequence of actions.

The screen capture below shows an action-based login sequence from the demonstration website.

Actions		Requests	
ID	Type	Value	Element
1	Click		Sign In
2	Set	jsmith	uid
3	Set	*****	passw
4	Click		btnSubmit
5	Click		SMALL BUSINESS

The final click (after submitting the credentials) is not part of the login process and can be deleted.

In this tab you can:

- Click  to replay the actions in the action-based player.
- Click on set items such as username and password to edit their values.
- Delete unnecessary steps to increase efficiency.

*Action-based player:*

Environment Definition view of the Configuration dialog box.

 **Opens the action-based player** and plays through the recorded actions. The left pane of the player shows the actions as a list, as the player plays them, and the right pane shows the , a browser with two panes: in the left pane is the sequence of clicks, followed by an internal verification step for the last action; in the right pane is a browser showing the current page.

## Requests:

The Requests tab shows the login sequence you recorded, as a sequence of requests.

The URLs that are part of the login procedure should be marked "Login", and the last one, after successful login, should be "In-Session". In the example below an additional request was sent after the login, and this is highlighted in yellow and marked "Ignore" as it is not needed as part of the sequence.

		Actions	Requests
URL	Method	Type	
<a href="http://demo.testfire.net/">http://demo.testfire.net/</a>	GET	Login	▼
<a href="http://demo.testfire.net/bank/login.aspx">http://demo.testfire.net/bank/login.aspx</a>	GET	Login	▼
<a href="http://demo.testfire.net/bank/login.aspx">http://demo.testfire.net/bank/login.aspx</a>	POST	Login	▼
<a href="http://demo.testfire.net/bank/main.aspx">http://demo.testfire.net/bank/main.aspx</a>	GET	In-Session (detected)	▼
<a href="http://demo.testfire.net/default.aspx?content=business.htm">http://demo.testfire.net/default.aspx?content=business.htm</a>	GET	Ignore	▼

AppScan attempts to automatically identify an "In-session pattern" that occurs on this page that it can use during the scan to verify that it is still logged in. During the scan, AppScan sends the in-session request repeatedly, and checks that the response contains the In-Session Detection Pattern, to verify that it is still logged in. If AppScan does not find the pattern in the page's response, AppScan assumes it has been logged out, and attempts to log in again by replaying the login sequence. It follows that the login sequence is typically played many times during a scan. It is therefore best that it contains as few steps as possible. It is also helpful if the In-Session page is a small page, and does not contain tracked parameters or cookies, since these can also increase scan time significantly.

In this tab you can:

- See the sequence of requests you sent when logging in
- See the in-session page highlighted in green

**Note:** The page marked "In-Session" should be the first page to be highlighted. If an earlier, "Login", page is highlighted then either the in-session pattern is wrong, or the wrong page is marked as "In-Session".

- Select a URL and click  to see the response in a browser.
- Select a different request as the in-session request
- See later steps that contain the in-session pattern highlighted in yellow and marked "Ignore"
- Delete unnecessary steps *before* the "In-session" URL, to save AppScan repeating these unnecessary steps many times during a scan

*Viewing URL Request/Response details:*

#### About this task

From the Details tab of the Login Management view of the Scan Configuration dialog box you can view details of any request in the login sequence.

#### Procedure

1. Click on the URL to select it, then click .

The URL is displayed in the browser with two tabs: Browser and Request/Response. The Browser tab is uppermost, showing the URL as a web page.

2. To see the actual HTML, click on the **Request/Response** tab to bring it to the front.

*Exporting a Login Sequence:*

#### About this task

When you record a login sequence it is saved as part of the scan. If you save the scan as a template, the login sequence is saved as part of the template. You can also export the login sequence by itself, to use in future scans. These files are saved with the **.login** extension.

#### Procedure

1. Record a login sequence (see "Recording a login" on page 47).
2. At the bottom of the **Session Information** dialog box, click **Export**.
3. Give the file an easily recognizable name and click **Save**.

The file is saved with the LOGIN extension.

*Importing a Login Sequence:*

#### About this task

You can load a previously recorded login procedure that was saved as a LOGIN file (filename ends in **.login**) to use for the current scan.

#### Procedure

1. In the **Login/Logout** view of the **Scan Configuration** dialog box, select **Recorded Login**.
2. Click **Import**.  
If there is a current login sequence, you are warned that loading the saved sequence will discard the current one.
3. Click **OK**.
4. Browse to the LOGIN file you require and click **Open**.

#### In-Session Detection:

The lower part of the Configuration > Login > Details tab shows the pattern (if any) used by AppScan to verify that it is still logged in to the application it is scanning.

When you record a login procedure, AppScan attempts to identify a pattern in the in-session response page that indicates that the user is in fact in-session. For example, it may contain a "Log Out" link, which is only available because the user is logged in. When it finds such a pattern it displays it in the **In-session pattern** field. When scanning, AppScan sends the in-session URL, and checks the response it receives for the in-session pattern, every few seconds, to confirm that it is still logged in to the site.

Sometimes AppScan is unable to identify an in-session pattern (for example if in-session status is indicated by a graphic with an unusual filename). Also, even if AppScan has correctly identified an in-session pattern, you may sometimes want to use a different in-session pattern (for example a regular expression (regexp) that includes both the pattern AppScan defined and other possible variations of it).

To change the Response Pattern you can:

- Type text directly into the **In-session pattern** text box, or
- Select the pattern using the **Select In-session pattern** button.

"*Changing the in-session detection pattern*"

*Changing the in-session detection pattern:*

You can change the definition AppScan uses during scanning to verify that it is still in-session.

#### About this task

You can manually select the in-session pattern using the **Select In-session pattern** button.

#### Procedure

1. Verify that the URL marked In-Session in the Login Sequence (in the upper half of the Session Information dialog box) is an in-session URL. (If necessary, click the Show in Browser link to view the page.)
2. Click **Select In-session pattern**.  
The browser opens in a special mode with two tabs: Browser and Request Response.
  - **In the Browser tab:** Different areas are highlighted in pale blue as you move the mouse over the page.
  - **In the Request/Response tab:** The source HTML is shown, with the current In-Session Detection pattern highlighted in yellow.
3. Select a new pattern by doing one of the following:
  - **In the Browser:** Move the mouse until the text or image that you want is highlighted, and click.
  - **In the Request/Response tab:** Select the required text in the HTML and right-click.

**Note:** In the case of images the highlights may not show up, however the method is the same: Click on the required image.

The selected text or image is highlighted in yellow.

4. Close the browser.  
The selected pattern appears in the **In-session detection** field.
5. Click **OK** to save the change.

*Optimizing In-Session Detection:*

Reviewing the login sequence in the Requests tab can help troubleshoot and optimize in-session detection.

## About this task

AppScan attempts to identify automatically an "In-Session Detection Pattern" that occurs on the in-session page, that it can use during the scan to verify that it is still logged in. This should be a pattern that occurs in the HTML only when you are logged in. An example might be text that reads: Click here to log out".

During the scan, AppScan sends the in-session request repeatedly, and checks that the response contains the In-Session Detection Pattern. If AppScan does not find the pattern in the page's response, AppScan assumes it has been logged out, and attempts to log in again by replaying the login sequence. It follows that the login sequence is typically played many times during a scan. It is therefore best that it contains as few steps as possible. It is also helpful if the In-Session page is a small page, and does not contain tracked parameters or cookies, since these can also increase scan time significantly.

When the defined in-session pattern is detected in the in-session request, (the request immediately following the POST request), it is highlighted in green. Requests after the in-session request are highlighted in yellow. They are not needed to log in and should be deleted.

URL	Method	Type
http://demo.testfire.net/	GET	Login
http://demo.testfire.net/bank/login.aspx	GET	Login
http://demo.testfire.net/bank/login.aspx	POST	Login
<b>http://demo.testfire.net/bank/main.aspx</b>	<b>GET</b>	<b>In-Session (detected)</b>
http://demo.testfire.net/default.aspx?content=business.htm	GET	Ignore

## Procedure

1. Verify that the In-Session Detection Pattern that was automatically selected is in fact an indication that the user is logged in. If necessary, change it.
2. Verify that there are no unnecessary steps in the login procedure. If there are, delete them.
3. Verify that the in-session response is not a large one, and - if possible - does not include tracked parameters or cookies. If necessary, add one or more steps till you reach a smaller page or one without tracked items.
4. If you succeeded in selecting an in-session page without tracked parameters or cookies, there is no need for AppScan to check for these each time it logs in. Go to **Advanced Configuration > Session Management: Parse In-Session Page**, and change the setting to False.
5. If none of these succeed, you can try to identify an out-of-session pattern instead, and then change the detection method.

## Session IDs tab

Scan Configuration > Login Management > Session IDs tab.

The **Session IDs** tab of **Login Management** view is used to review and manage the tracking of variables (session IDs) received during the recorded login.

The tab lists all variables received during the login sequence, and automatically designates Session IDs as Tracked (a checkmark appears next to the variable in the Tracking column).

You can select variables in this list and use the **Track** and **Stop Tracking** buttons to change their status. All "tracked" parameters are added to the global list of parameters and cookies whose status is tracked during scanning (see "Parameters and Cookies view" on page 65).

**Tip:** If you are not sure which items are session identifiers and your scan is going out-of-session, contact your application's developers to see if they can supply a list of parameters and cookies the application uses to maintain session.

## Environment Definition view

Environment Definition view of the Configuration dialog box.

Environment definition is not essential, but enables AppScan to safely refrain from sending non-relevant tests during the scan, resulting in a faster and more accurate scan.

**Note:** Where relevant, some of the list boxes allow you to select more than one option by pressing the **Ctrl** key when selecting items in a list.

Metric	Comments
Operating System	Operating System of application being scanned.
Web Server	Select <i>all</i> applicable answers.
Application Server (if any)	Select <i>all</i> applicable answers.
Type of Database (if any)	Select <i>all</i> applicable answers.
Third-Party Component (if any)	Select <i>all</i> applicable answers.
Location of Site	
Type of Site	
Deployment Method	
Collateral Damage Potential	The potential for damage or theft if the application is vulnerable.
Target Distribution	The proportion of systems in the environment that are potential targets.
Availability Requirement	The relative importance of availability (of information).
Confidentiality Requirement	The relative importance of confidentiality (of user information).
Integrity Requirement	The relative importance of integrity (accuracy) of information.

**Note:** The last five items are the Environmental CVSS metrics for the site. If you define the relative importance of these metrics in your application environment, AppScan will take these definitions into account when assigning severity values to vulnerabilities it finds during the scan. These definitions are global. (You can adjust the Environmental Metrics for a *specific* issue from the Detail Pane > Severity > Open CVSS Panel, see “CVSS settings” on page 180.)

## Exclude Paths and Files view

Exclude Paths and Files view of the Configuration dialog box.

You can configure AppScan to ignore certain paths in the application, or specific types of file. However, you should apply exclusions with caution, as they may have important issues.

Setting	Details
Exclude Paths	You can filter the scope of the automatic Explore stage by adding URLs (full paths, which may include queries) or Regular Expressions to the <b>Exclude or Include Paths</b> list.  For details see “Exclude Paths” on page 57

Setting	Details
Exclude File Types	You can configure AppScan to ignore specific types of file during the scan. For example, you could make the scan run faster if you exclude graphics files. However, you should exclude files with caution, as they may contain important issues.  For details see “Excluding file types” on page 61

“Limiting scan to the Starting URL folder” on page 44

## Exclude Paths

Exclude paths and files view of the Configuration dialog box.

You can filter the scope of the automatic Explore stage by adding URLs (full paths, which may include queries) or Regular Expressions to the **Exclude Paths** list. Possible reasons for doing this may be:

- Because they are still under development and you know they are problematic and do not want to scan them right now
- Because you know they are *not* problematic and you want to reduce scan time
- To reduce scan time by restricting the scan to certain parts of the application

For any path defined, you can optionally limit the filter to one or more specific parameters in the path. A possible reason for doing this may be:

- For megascript applications (applications contained in a URL, and controlled by its parameters) where you want to exclude certain parameters (such as the login or log out parameter) from scans.

There are two types of entry in this table:

- **Exclude:** Listed paths are excluded from the scan.

Any link that matches a path configured to be an exclusion is filtered out of the scan.

**Note:** You can also exclude a path by right-clicking on it in the Application Tree, and selecting **Exclude from Scan**.

- **Exception:** Use to include a specific directory within a path that was excluded *higher up in the list*.

**Note:** The **Exception** function is only needed if you want to include a directory that is within a path that you have excluded. For example, if you have excluded: `http://demo.testfire.net/bank` you could then add `http://demo.testfire.net/bank/transfer.aspx` as an Inclusion, lower down in the list, to include that subdirectory in scans.

**Note:** If you add Exclusions between the Explore and Test stages of scanning, AppScan will not test the excluded paths even though they were explored.

“Adding new exclusions or exceptions”

### Adding new exclusions or exceptions:

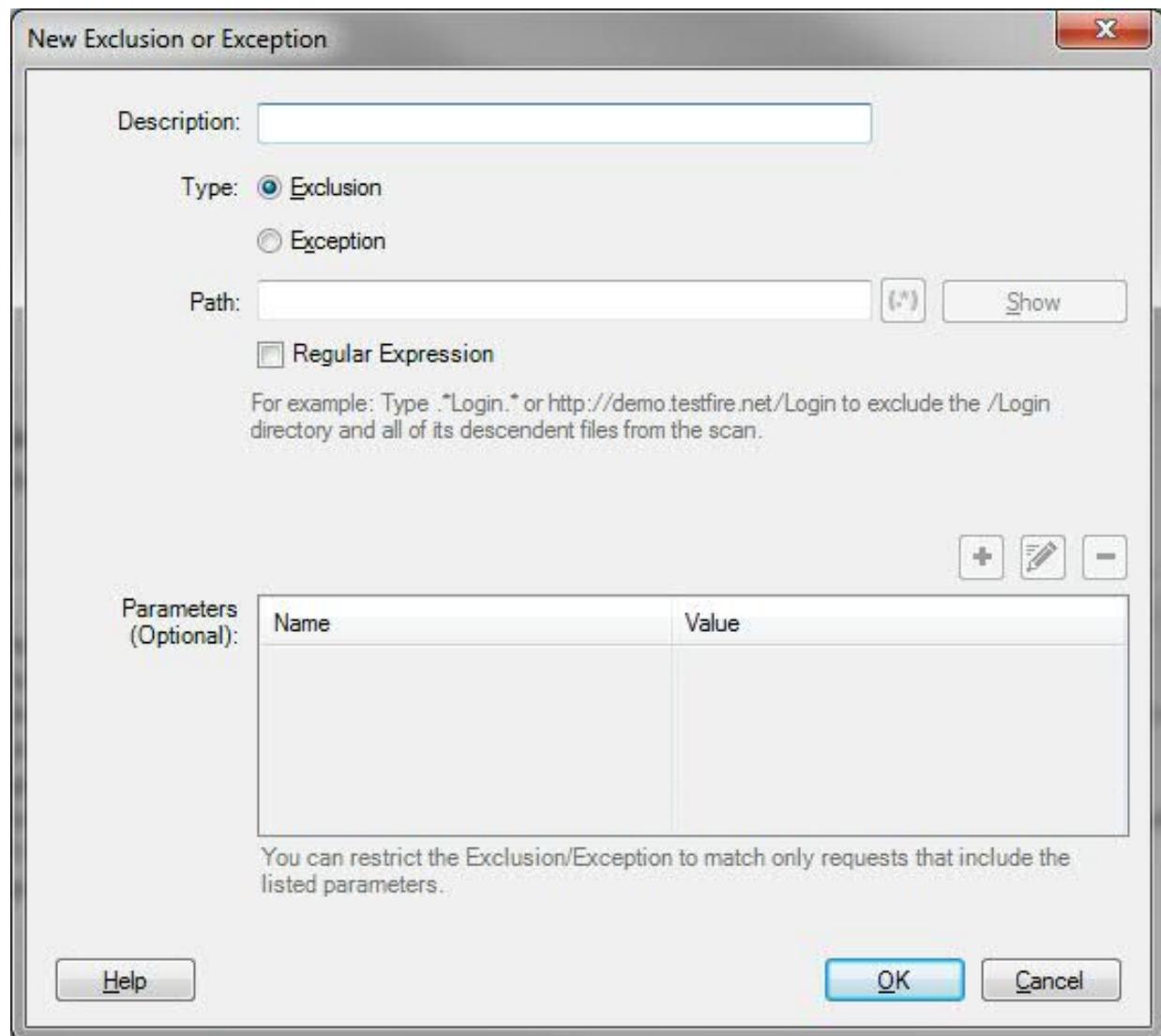
#### Procedure

1. In the Exclude Paths area of the Exclude Paths and Files tab of the Configuration dialog box, click



, select Exclusion or Exception, enter the full path or Regexp., and click **OK**. (In the **Scan Configuration Wizard > Starting URL > Advanced**, click **Explore**.)

The **New Exclusion or Exception** dialog box opens.



2. Select the radio button for the filter you want to create:
  - **Exclusion:** Do not scan URLs matching this item.
  - **Exception:** Include URLs matching this item even though they are excluded by an Exclusion higher up on the list.

**Note:** The **Exception** function is only needed if you want to include a directory that is within a path that you have excluded. For example, if you have excluded: `http://demo.testfire.net/bank` you could then add `http://demo.testfire.net/bank/transfer.aspx` as an Inclusion, lower down in the list, to include that subdirectory in scans.

**Note:** If you add Exclusions between the Explore and Test stages of scanning, AppScan will not test the excluded paths even though they were explored.

3. Optionally add a description that will appear in the Exclude Paths list.
4. Type a path or regular expression that will match a set of directories into the Path field (see examples in table following), and for a regular expression select the check box.

**Note:** A regular expression (regexp.) is a string that describes a set of strings, according to certain syntax rules. Click  to open the Expression Test PowerTool, which can help you verify the syntax of your regular expressions.

If you need additional help writing regular expressions you may find the following link useful:  
<http://www.regular-expressions.info/quickstart.html>

5. To apply the exclusion or exception to paths that include specific parameters only, click  to add one or more parameters to the lower pane of the dialog box.

**Note:** This feature is designed for "megascript" applications, where the entire application is contained in a URL and controlled by its parameters. Filtering out the URL would disable the scan, but you can filter out specific parameters, or even specific parameter values (such as those that login or logout).

6. Click **OK**.

The new item is added to the bottom of the list.

**Note:** Where there is a conflict between two items in the list, the *lower* item takes priority. Use the **Up/Down** buttons to adjust the order of items as required. If an exclusion or inclusion is made redundant due to another exclusion or inclusion higher in the list, the redundant item will be deleted from the list when you click **OK**.

### Example

Type	Example and Function
Exclude	<b>http://demo.testfire.net/transfer</b> or <b>http://demo.testfire.net/transfer/</b> Filters out the specified URL and all sub-directories and files
Exclude	<b>.*private.*</b> Exclude all URLs containing the string <b>private</b>
Exclude	<b>.*_bk.aspx</b> Exclude all URLs ending in <b>_bk.aspx</b>
Exception	<b>http://demo.testfire.net/transfer/customize.aspx</b> When an earlier Exclusion (such as the first one in this table) excludes sub-directories and files, this Exception includes this particular path in the scan. Note that the Exception must appear <i>below</i> the Exclusion to take effect

### Editing exclusions or inclusions:

#### Procedure

1. Select an item in the **Exclude or Include Paths** list.
2. Click **Edit**.  
The **Edit Exclusion or Inclusion** dialog box appears, displaying the properties of the selected item.
3. Change as required and click **OK**.

## What to do next

See also: "Limiting scan to a specific folder"

"Limiting scan to the Starting URL folder" on page 44

### Limiting scan to a specific folder

Using Exclusions and Exceptions to limit the range of a scan.

#### About this task

You can limit automatic scanning to a specific folder or folders by excluding all URLs, and then including the directory or directories required.

#### Procedure

1. Open **Scan Configuration > Exclude Paths and Folders**.
2. Add an **Exclude** item with path of your site, such as <http://www.mysite.com/> (see "Exclude Paths" on page 57).
3. Add one or more **Exception** items *below* the **Exclude** item, with the path(s) you want to scan.

**Note:** The **Exception** item(s) must appear *below* the **Exclude** item in order to take effect. If necessary, use the **Up/Down** buttons to adjust the order before closing.

4. Verify that the starting URL (see "URL and Servers view" on page 43) is not within one of the excluded paths. (If it is the scan will not be able to start!)
5. Click **OK** to save changes and close the list.

#### Example

"Example 1"

"Example 2"

See also: "Limiting scan to the Starting URL folder" on page 44

#### Example 1:

#### About this task

Let's consider a case where the Starting URL for the scan (see "URL and Servers view" on page 43) is: <http://www.mysite.com/index.aspx> and we want to restrict the scan to the folder: <http://www.mysite.com/myfolder/>

#### Procedure

1. Add an **Exclude** item with path <http://www.mysite.com/> (see "Exclude Paths" on page 57).
2. Add an **Include** item *below* the **Exclude** item, with the path of the Starting URL: <http://www.mysite.com/index.aspx>
3. Add a *second* **Include** item below the **Exclude** item, with the path of the folder to be scanned: <http://www.mysite.com/myfolder/>

#### Example 2:

#### About this task

Let's now consider a case where the Starting URL for the scan is within the folder that is to be scanned. The Starting URL is: <http://www.mysite.com/myfolder/index.aspx> and we want to restrict the scan to the folder: <http://www.mysite.com/myfolder/>

In this case the folder to be scanned and the Starting URL are added to the scan in a single **Include** item.

### Procedure

1. Add an **Exclude** item with path `http://www.mysite.com/`
2. Add an **Include** item *below* the **Exclude** item, with the path of the folder to be scanned:  
`http://www.mysite.com/myfolder/`

## Excluding file types

Excluding certain types of files from the scan.

### About this task

You can configure AppScan to ignore specific types of files, rather than complete pages, as in the Exclude Paths settings. For example, you could make the scan run faster if you exclude graphics. However, you should exclude files with caution, as they may contain important issues.

### Procedure

1. Open the **Scan Configuration** dialog box > **Explore Settings** view.
2. In the **Exclude File Types** pane, make sure the check boxes of the file types that you do not want to be scanned are selected.

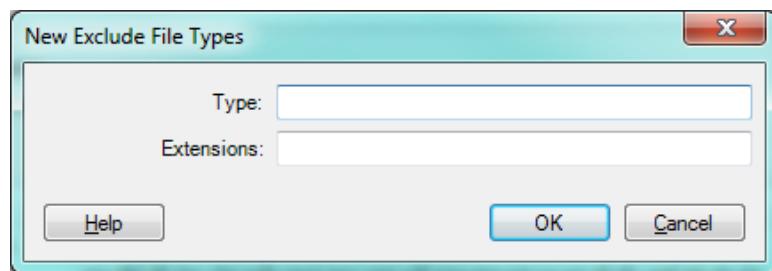
### Excluding additional file types:

#### About this task

If you have a file extension in your application that you cannot find listed in the **Exclude File Types** dialog box, you can add these extensions to the list.

### Procedure

1. In the **Exclude File Types** area, click  The **New Exclude File Types** dialog box appears.



2. In the **Type** text box, enter a word or phrase to describe a set of file extensions.
3. In the **Extensions** text box, enter the list of extensions that fall under this type of file. (Separate multiple extensions with a comma, without a space.)
4. Click **OK**.

### Editing the list of excluded file types:

#### Procedure

1. Select a row in the **Exclude File Types** dialog box.
2. Click **Edit**.

The **Edit Exclude File Type** dialog box appears, displaying the associated extensions of the selected exclusion.

3. Change the type name or extensions list as required
4. Click **OK**.

## Explore Options view

Explore Options view of the Configuration dialog box.

- The **Scan Limits** determine how deeply (or how quickly) AppScan explores your application.
- The **JavaScript** and **Flash** options determine whether AppScan should ignore or scan these scripts.
- The **Explore Method** determines whether AppScan explores all links on a page before continuing to the next page, or explores each new link as it is found.
- The **Browser Settings** are for configuring the client to recognize a specific server encoding and to send a specific user-agent header.

Setting	Details
<i>Scan Limits</i>	
Redundant Path Limit	<p>AppScan will not access the same path more than the specified number of times.</p> <p>A particular path may be visited several times if it appears with different parameters. This limit is relevant mainly for scripts.</p>
Click Depth Limit	AppScan will not scan pages that are accessed by clicking more than the specified number of links.
Total Page Limit	If selected, AppScan will access no more than the maximum number of pages defined. Note that there may be many URLs explored per page.
<i>JavaScript</i>	
Parse JavaScript code to discover URLs	AppScan will parse JavaScript code as text data to collect links.
Execute JavaScript to discover URLs and dynamic content	AppScan will actually execute JavaScript code and analyze the results to collect links, including dynamic links that may not be discovered by parsing alone. (This utilizes more system resources than parsing.)
Execute JavaScript when replaying login	If the application's login page uses JavaScript code, this check box must be selected in order for AppScan to be able to login during scanning.
<i>Explore Method</i>	
Breadth First	<p>(Default) AppScan explores page by page, exploring all links on one page before continuing to the next.</p> <p>It is recommended that you do not change the default selection of this option (Breadth First), unless you are aware of limitations in your application that demand that a user visits links in a specific order.</p>
Depth First	<p>AppScan explores link by link, exploring each new link as it is found</p> <p>If you change the Explore Method to <b>Depth First</b>, you must also change AppScan to use only one thread during the Explore; see "Explore Options view."</p>
<i>Other Settings</i>	
Encoding	<p>AppScan generally detects the application's encoding method automatically, and therefore <b>Autodetect</b> is selected by default.</p> <p>If the content of responses in the scan Results looks distorted, this may mean that the encoding method was not correctly identified. To solve this problem, select the correct encoding method from the drop-down list.</p>

Setting	Details
User-Agent	<p>The user-agent header in an HTTP request tells the server what kind of client sent the request, and this may affect the content that the server returns. For example, there may be content that is specific to mobile phones that is sent only when the user-agent is a mobile phone browser. In order for AppScan to be able to test such content, you need to configure it to send the appropriate user-agent header.</p> <p>AppScan generally detects the user agent automatically, and therefore <b>Autodetect</b> is selected by default. However, if you use a browser other than the built in browser, and you do not record a login procedure, a multi-step operation, or a manual explore, AppScan will be unable to autodetect the user agent, and you must select it manually.</p> <p>To change the user-agent, select an agent from the drop-down list.</p> <p>To enter custom content, click the <b>Edit</b> button and type in the content. When you close the dialog box the button name changes to <b>Custom User Agent</b>.</p> <p>For more details, see "User-agent headers."</p> <p><b>Note:</b> If you change the default browser, refer to the conditions listed in "Changing the default browser" on page 297</p>
WebSphere Portal	<p>If the site is a WebSphere Portal site, AppScan will need to get URL decoding information from the site for more efficient scanning and to build a useful application tree. To enable decoding, select <b>Enable WebSphere Portal scanning</b>.</p> <p>If the context root URL does not follow the default format, click <b>Add Context Root URL</b> to add one or more context root URLs.</p> <p><b>Tip:</b> If you are not sure what your portal's context root URL is, the following technote may help: Locating the Websphere portal context root.</p> <p><b>Tip:</b> When scanning a WebSphere Portal site, it is recommended to use the predefined WebSphere Portal scan template, which is configured for the purpose.</p>
<i>Flash</i>	
Parse Flash to discover URLs	AppScan will parse Flash code as text data to collect links.
Execute Flash files to discover potential vulnerabilities	<p>AppScan will actually <i>play</i> Flash files and analyze the results to collect links, including dynamic links that may not be discovered by parsing alone. (This utilizes more system resources than parsing.)</p> <p>Adobe Flash Player for Internet Explorer, Version 9.0.124.0 or higher is required. If a supported version is not installed a warning appears next to the check box when selected, and Flash Execution will not run. See "Flash content" on page 292.</p> <p>If you have Adobe Flash Player Version 10.1 or higher, you may get a message that it requires configuration to work with AppScan. See "Flash Player configuration" on page 7.</p> <p>When Flash Execution is selected, three Flash execution limits can also be configured. These are:</p> <ul style="list-style-type: none"> <li>• <b>Depth Limit:</b> The maximum number of "clicks away from the initial screen" that can be clicked before the scanning of any particular Flash movie is stopped and the scan moves on.</li> <li>• <b>Click Limit:</b> The maximum total number of clicks that can be made before the scanning of any particular Flash movie is stopped and the scan moves on.</li> <li>• <b>Screen Limit:</b> The maximum number of unique Flash states allowed before the scanning of any particular Flash movie is stopped and the scan moves on.</li> </ul>

## User-agent headers

This section describes the user-agent header sent for each user-agent, and how to work with "Auto-Detect".

The default user-agent setting is "Auto Detect", and in most cases this will be able to automatically detect the user-agent. The user-agent is detected when you record a login procedure, a multi-step operation, or a manual explore. If none of these is recorded, AppScan will use the Internet Explorer user-agent.

If necessary, you can edit the content of any user-agent header by clicking the Edit button. The updated user-agent will then appear as "Custom User Agent".

You can select a header by clicking the User-Agent button, and edit the content by clicking the Edit button. Once you edit the content, the button name changes to "Custom User Agent". Custom User Agent content is only saved with the scan (and transferred to other computers along with the scan) if it is the selected user-agent when the scan is saved.

The table below shows the user-agent header sent for each user-agent.

User-Agent	Header
Auto Detect	<p>This is the default setting.</p> <p>If you record a login procedure, a multi-step operation, or a manual explore, AppScan will automatically detect the user-agent. If none of these is recorded, AppScan will use the Internet Explorer user-agent.</p> <p><b>Important:</b> Once AppScan enters text in the Auto Detect header field, text on the button changes to "Auto Detect (Detected)", and AppScan does not check it again. So if you change the browser used for scanning after the Auto Detect content has been filled and marked as "Detected", you must click the Edit button and delete the content of the field, so that AppScan will automatically detect the new browser.</p>
Auto Detect (Detected)	If the User-Agent is set to Auto Detect and a login procedure, a multi-step operation, or a manual explore has been recorded, the setting changes to "Auto Detect (Detected)", and the content is added.
Custom User Agent	<p>If you edit the content of the selected user-agent, the setting is saved as a "Custom User Agent".</p> <p><b>Important:</b> Custom User Agent content is only saved with the scan (and transferred to other computers along with the scan) if it is the selected user-agent when the scan is saved.</p>
<i>Desktop browsers</i>	
Internet Explorer	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; Win64; x64; Trident/4.0; .NET CLR 2.0.50727; SLCC2; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; Tablet PC 2.0)
Firefox	Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.2.13) Gecko/20101203 Firefox/3.6.13
Chrome	Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US) AppleWebKit/534.10 (KHTML, like Gecko) Chrome/8.0.552.237 Safari/534.10
Opera	Opera/9.80 (Windows NT 6.1; U; zh-tw) Presto/2.7.62 Version/11.01
Safari	Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_6_7; en-us) AppleWebKit/534.16+ (KHTML, like Gecko) Version/5.0.3 Safari/533.19.4
<i>Mobile browsers</i>	
iPhone (Safari)	Mozilla/5.0 (iPhone; U; CPU iPhone OS 4_2_1 like Mac OS X; en-us) AppleWebKit/533.17.9 (KHTML, like Gecko) Version/5.0.2 Mobile/8C148 Safari/6533.18.5
iPad (Safari)	Mozilla/5.0 (iPad; U; CPU OS 4_2_1 like Mac OS X; ja-jp) AppleWebKit/533.17.9 (KHTML, like Gecko) Version/5.0.2 Mobile/8C148 Safari/6533.18.5
Google Android	Mozilla/5.0 (Linux; U; Android 2.2; he-il; GT-I9000 Build/FROYO) AppleWebKit/533.1 (KHTML, like Gecko) Version/4.0 Mobile Safari/533.1
Blackberry	BlackBerry9700/5.0.0.862 Profile/MIDP-2.1 Configuration/CLDC-1.1 VendorID/331
Opera Mini	Opera/9.80 (Series 60; Opera Mini/5.1.22784/22.394; U; en) Presto/2.5.25 Version/10.54

User-Agent	Header
Opera Mobile	Opera/9.80 (Android; Linux; Opera Mobi/ADR-1012221546; U; pl) Presto/2.7.60 Version/10.5
Microsoft IE 7 for mobile	Mozilla/4.0 (compatible; MSIE 7.0; Windows Phone OS 7.0; Trident/3.1; IEMobile/7.0) Asus;Galaxy6

## Parameters and Cookies view

Parameters and Cookies view of the Configuration dialog box.

This view is used to manage three main functions:

- Assign special treatment to specific parameters and cookies
- Define parameters and cookies that have a special format which AppScan might not recognize on its own
- Control the default treatment of parameters and cookies ("redundancy tuning")

Setting	Description	See
Parameters and Cookies tab	<p>Lets you view, add, edit and delete global parameters that require non-default treatment.</p> <p>For example, your application may have parameters and cookies whose values you do not want AppScan to manipulate during tests. To make sure that AppScan does not change these parameters and cookies, exclude them from tests. For example, your application might lock a user session if certain cookie or parameter values are changed. You should exclude these parameters from manipulation. If you do not exclude them, AppScan may not be able to successfully complete the scan, as these cookies will lock AppScan out of the application.</p> <p>During the Explore stage, AppScan automatically detects cookies and HTML parameters that are likely to be session IDs and adds them to the list in this tab. You can manually add cookies and parameters that you know to be session IDs.</p> <p>The columns in this tab are defined in the table below.</p> <p><b>Note:</b> The <b>Hide/Show template items</b> button lets you filter out items that originated in the scan template, which may not be relevant to the current scan.</p>	"Parameter definition" on page 66
Advanced: Custom Parameters tab	Lets you add, edit and delete parameters with a custom format that AppScan might not otherwise recognize as such.	"Advanced: Custom Parameters tab" on page 72
Redundancy Tuning Defaults	<p>This link (at the bottom of the Parameters and Cookies tab) lets you access and edit the <i>default</i> redundancy tuning applied to all parameters, whether discovered by AppScan or defined by the user.</p> <p><b>Note:</b> Changing the specific redundancy tuning of an individual parameter is done as part of "Parameter definition" on page 66</p> <p>Changes to the defaults are not applied retroactively to parameters that have already been defined. This must be done manually for each parameter.</p>	"Redundancy tuning" on page 69

## Parameters and Cookies tab fields

The following table summarizes the fields in this tab.

Heading	Options and description
Type	Parameter / cookie / custom parameter
Name	
Tracking	How to track this parameter/cookie: <ul style="list-style-type: none"><li>• As a login value</li><li>• As a dynamic value</li><li>• As a fixed value</li><li>• Don't track it at all</li></ul>
Test Exclude	Defines whether or not to exclude this parameter/cookie from testing during the Test stage of the scan.
Redundancy Tuning	<ul style="list-style-type: none"><li>• Default: The default redundancy tuning is applied to this item</li><li>• Custom: The redundancy tuning for this item is different to the current default</li></ul>
Source	Shows from where AppScan obtained this item: <ul style="list-style-type: none"><li>• Scan template: Originated in the scan template</li><li>• Login session ID: From the login sequence recorded by the user</li><li>• Multi-step sequence variable: From a sequence recorded by the user</li><li>• Scan Expert [module name]: From the specified Scan Expert module</li><li>• Explore Optimizer: From the Explore Optimizer extension</li><li>• User-defined</li></ul>

## Parameter definition

### Procedure

To add a new definition, click  (or, to edit an existing parameter, select it and click 

Setting	Description
Type	<p>Select a parameter type from the drop-down list:</p> <p><b>Parameter:</b> All parameters matching this name are included in the definition.</p> <p><b>Cookie:</b> All cookies matching this name are included in the definition.</p> <p><b>Custom Parameter:</b> This is a custom parameter (select one of the custom parameters from the <i>Name</i> drop-down list)</p>
Name	<p>The name of the parameter or cookie.</p> <p>Select the adjacent check box if the name you enter is a regular expression. If you do this you can also open the Expression Test PowerTool by clicking  , to help you verify the syntax of your regular expression.</p> <p>See “Parameter names” on page 68 for details.</p>
Comments	You may optionally add a comment about the parameter in this field for your own reference.
Hosts	<p><b>If a Host is specified:</b> Use this parameter for the specified host only.</p> <p><b>If left blank:</b> Use this session ID for all hosts.</p>
Path	<p>If the application supplies cookies of the same name from different parts of the application, you can differentiate between them by defining the path for each one.</p> <p>Blank or / will include all occurrences of the cookie.</p>
Test Exclude	Select this check box only if you are sure you don't want AppScan to test this parameter at all.
Tracking	This setting tells AppScan that this parameter or session ID should be updated during the scan whenever a new value is set by the application, so that a valid cookie/parameter is always sent in requests to the application.

Setting	Description
Tracking Options...	<p>(Click the link to open this optional section of the dialog box.)</p> <p>These options let you fine-tune how the tracked parameter or cookie is treated.</p> <p><b>Track Type</b></p> <ul style="list-style-type: none"> <li>• <b>Login Value:</b> (Default, and Recommended) Requests sent to the application that include this parameter use the value of the parameter received at the end of the login process.</li> <li>• <b>Dynamic Value:</b> Requests sent to the application that include this parameter use the most recent value received from the application.</li> <li>• <b>Fixed Value:</b> Requests sent to the application that include this parameter always use the value that you enter in the Value field.</li> </ul> <p>See “Session IDs” on page 69 for more details.</p> <p><b>Send cookie on all requests:</b> When selected, the cookie will be included in all requests, even if not explicitly set by the application.</p> <p><b>Treat as Group:</b> If the cookie name is a regular expression, define whether to treat different cookie names that match the regexp. as a group (and therefore update the name as well as the value, when there are changes) or as separate cookies.</p> <p><b>Response Pattern:</b> Generally, AppScan updates parameter or cookie values based on the content of links extracted from the response (parameters) or from the cookie header (cookies). If AppScan will not be able to extract the value unaided, you can supply the regexp. that AppScan can use to extract the value from the raw response. The regexp. must contain at least one group, and AppScan will extract the first match.</p> <ul style="list-style-type: none"> <li>• <b>URL Filter:</b> If you know that the parameter/cookie only appears in a specific URL, you can improve scan efficiency by defining it here.</li> <li>• <b>Encoding:</b> If the extracted value must be encoded when pasted into the request, define the method here. If you are unsure of the coding, select <b>According to context</b>; if you are sure, selecting the <i>correct</i> encoding is preferable. Options are: None, According to context, URL, XML, Json.</li> <li>• <b>Match:</b> Select <b>Header and Body or Body only</b>.</li> </ul>
Redundancy Tuning...	<p>(Click the link to open this optional section of the dialog box.)</p> <p>These four check boxes let you fine-tune how AppScan relates to changes in the parameter (or even its existence) during the Explore and Test stages of the scan. See “Redundancy tuning” on page 69</p>

## Identifiers that define a parameter or cookie

A parameter or cookie is recognized as unique on the basis of certain identifiers. It follows that you cannot define two or more parameters or cookies with the same identifiers. The table below shows the identifiers for each kind of entry.

Parameter	Parameter name, whether a regular expression, host
Cookie	Parameter name, whether a regular expression, host, path
Custom parameter	Extracted name (if one exists), reference name, host, occurrence index

## Parameter names

### Procedure

In the **Name** text box, enter the parameter or cookie name, or a regular expression that matches it:

- All text is case-sensitive; add (?i) to the regexp if you want it to be case-insensitive

- Strings in the regexp are considered partial; you do not need to add `(.*)`

A number of default definitions are provided.

Type	Value	Includes
Parameter	<code>__VIEWSTATE</code>	parameters with this string in their name
Parameter and Cookie	<code>^CFID</code>	parameters and cookies whose names begin with CFID or cfid
Parameter and Cookie	<code>Token</code>	parameters and cookies whose names include "Token"

**Note:** A regular expression (regexp) is a string that describes a set of strings, according to certain syntax rules. IBM Security's Expression Test PowerTool (**Tools > Expression Test**) can be useful to verify the syntax of your regexps.

If you need additional help writing regexps you may find the following link useful: <http://www.regular-expressions.info/quickstart.html>

## Session IDs

If the site uses time-restricted session IDs (in the form of cookies or parameters), the site will reject requests that contain expired tokens; causing the site tests to fail.

Therefore, AppScan must be able to recognize and handle the HTML parameters or cookies that are time-restricted *session IDs*. AppScan will assign session IDs the most recent value available, thus preventing the application session from expiring.

You can determine whether AppScan should automatically update the value of a session ID. Set the **Status** of a session ID:

- **Login Value:** (Recommended) When sending a test request containing this parameter, AppScan automatically updates the session ID with the value received from the application after successful login.

This status is recommended for most parameters and cookies, unless there is a specific need to set a specific value. However, when Login Value session IDs are used, the value might expire while it is in the database.

To update a tracked session ID in the database: Just before running the scan, visit the URL where the session ID is sent. A new session ID will be sent, with an updated value.
- **Dynamic:** AppScan automatically updates the session ID value during the Test stage, according to new values set by the web application in prior tests (for example, as with Shadow Cookies).

Select *Dynamic* only if you know that your web application enforces security measures that demand that a specific session ID be updated during certain usage procedures.
- **Fixed:** Retains a fixed value. Set a fixed value for a session ID if your web application security needs this session ID to always have this value.

During the Explore stage, AppScan automatically detects cookies and HTML parameters that are likely to be session IDs and adds them to a list. You can manually add the cookie and parameters that you know to be session IDs when you configure the scan.

## Redundancy tuning

Careful redundancy tuning can significantly reduce scan time.

AppScan tries to avoid sending more than one request when it is clear that no new information will be revealed. In many cases a difference in the value of a particular parameter is insignificant, and it is not necessary to send more than one request when all other parameters are the same and only that value is different.

Consider the following two requests:

`.../doAction.pl?action=buy&timestamp=14:00&n=1`

`.../doAction.pl?action=buy&timestamp=15:30&n=1`

The only difference between them is their Timestamp value.

In most cases it would be sufficient to send a single request using either of the configurations, and unnecessary to send them both. It is unlikely that the response to one of the requests would reveal any weakness that would not also be revealed by the other. You should therefore configure the Redundancy Tuning settings for the Timestamp parameter so that in cases like this only one of the requests will be sent.

*See also:*

"Redundancy tuning options"

"Redundancy tuning defaults" on page 71

"Changing redundancy tuning defaults" on page 72

"Changing redundancy tuning for a specific parameter" on page 72

#### **Redundancy tuning options:**

Redundancy tuning for parameters and cookies with special attributes, listed in Configuration > Parameters and Cookies.

check box	When selected...
Explore the URL again whenever this parameter/cookie is added or removed.	During the Explore Stage, treat URLs whose only difference is that one includes this parameter and one does not, as different URLs, and explore both.  For example, in the case of the following two URLs, <i>both</i> will be explored: <code>...page.jsp</code> <code>...page.jsp?thisParam=Value</code>  If you deselect this check box, only one request will be sent in cases like this, and the other will be discarded.
Explore the URL again whenever the value of this parameter/cookie changes.	During the Explore Stage, treat URLs whose only difference is the value of this parameter/cookie, as different URLs, and explore both.  For example, in the case of the following two URLs, <i>both</i> will be explored: <code>...page.jsp?thisParam=Value1</code> <code>...page.jsp?thisParam=Value2</code>  If you deselect this check box, only one request will be sent in cases like this, and the other will be discarded. <b>Note:</b> This option is irrelevant when the parameter or cookie is tracked.

check box	When selected...
Repeat all adjacent parameter/cookie tests whenever this parameter/cookie is added or removed.	<p>During the Test Stage, treat URLs whose only difference is that this parameter was added or removed, as different URLs, and test adjacent parameters again.</p> <p>For example, in the case of the following two URLs, <i>two full sets</i> of tests will be generated for the adjacent parameter, one set for each URL.</p> <pre>...page.jsp?adjacentParam=&lt;test_this&gt; ...page.jsp?adjacentParam=&lt;test_this&gt;&amp;thisParam=Value</pre> <p>If you deselect this check box, only one set of tests will be generated for the adjacent parameter.</p>
Repeat all adjacent parameter/cookie tests whenever the value of this parameter/cookie changes.	<p>During the Test Stage, treat URLs whose only difference is the value of this parameter/cookie, as different URLs, and test adjacent parameters again.</p> <p>For example, in the case of the following two URLs, <i>two full sets</i> of tests will be generated for the adjacent parameter, one set for each URL.</p> <pre>...page.jsp?adjacentParam=&lt;test_this&gt;&amp;thisParam=Value1 ...page.jsp?adjacentParam=&lt;test_this&gt;&amp;thisParam=Value2</pre> <p>If you deselect this check box, only one set of tests will be generated for the adjacent parameter.</p> <p><b>Note:</b> This option is irrelevant when the parameter or cookie is tracked.</p>

*See also:*

"Redundancy tuning" on page 69

#### Redundancy tuning defaults:

Redundancy tuning for parameters and cookies with special attributes, listed in Configuration > Parameters and Cookies.

If the Explore Optimization eXtension is not run, the default redundancy tuning setting is this:

Check box	Default value
Explore the URL again whenever this parameter/cookie is added or removed.	✓
Explore the URL again whenever the value of this parameter/cookie changes.	✓
Repeat all adjacent parameter/cookie tests whenever this parameter/cookie is added or removed.	✓
Repeat all adjacent parameter/cookie tests whenever the value of this parameter/cookie changes.	✗

If the Explore Optimization eXtension is activated and run, the redundancy tuning setting for navigational parameters is *increased*, while the default setting (applied to all non-navigational parameters) is *decreased*.

Check box	Navigational	Default
Explore the URL again whenever this parameter/cookie is added or removed.	✓	✓
Explore the URL again whenever the value of this parameter/cookie changes.	✓	✗

Check box	Navigational	Default
Repeat all adjacent parameter/cookie tests whenever this parameter/cookie is added or removed.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Repeat all adjacent parameter/cookie tests whenever the value of this parameter/cookie changes.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

See also:

"Redundancy tuning" on page 69

#### Changing redundancy tuning defaults:

The default redundancy tuning setting is applied to all new additions to the list in Configuration > Parameters and Cookies, unless manually changed.

#### Procedure

- At the bottom of the Parameters and Cookies tab, click **Redundancy Tuning Defaults**.

The Redundancy Tuning Defaults dialog box opens. Its options are described in "Redundancy tuning options" on page 70.

- Adjust the settings as necessary (see "Redundancy tuning options" on page 70 for details).

Changes you make here are applied to all new parameters as they are discovered by AppScan or defined by the user.

**Note:** Changes to the defaults are *not* applied retroactively to parameters that have already been defined. This must be done manually for each parameter.

#### What to do next

See: "Redundancy tuning options" on page 70

#### Changing redundancy tuning for a specific parameter:

##### Procedure

- In the Parameters and Cookies tab, select the parameter and click 

The **Parameter Definition** dialog box appears (see "Parameter definition" on page 66 for details).

- At the bottom of the dialog box, click **Additional Options**.

The Redundancy Tuning options open (see "Redundancy tuning options" on page 70 for details).

#### What to do next

See "Redundancy tuning options" on page 70

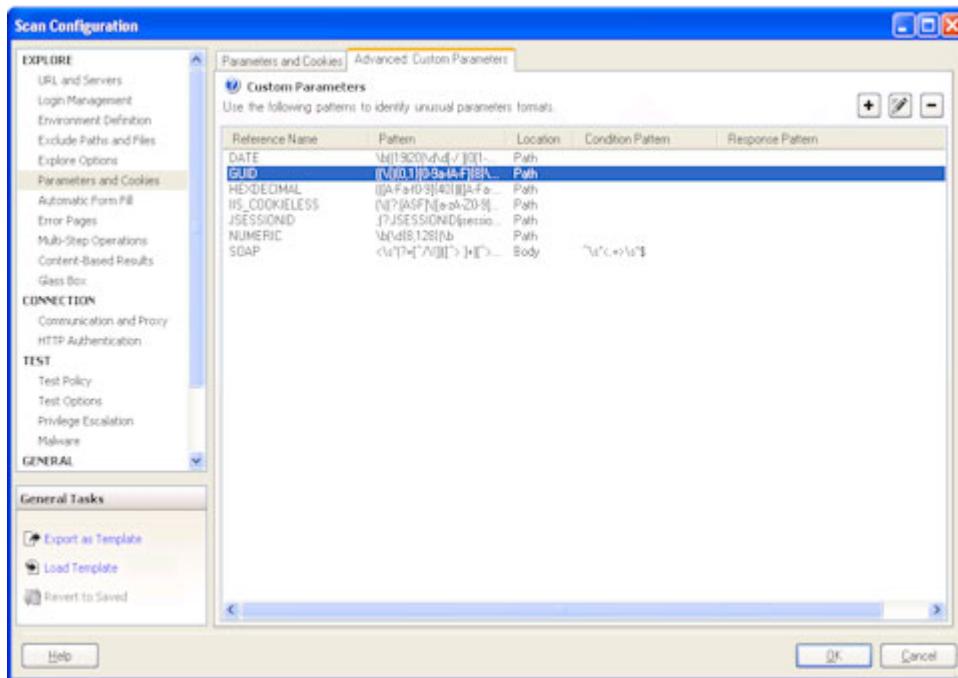
#### Advanced: Custom Parameters tab

The Advanced tab of Parameters and Cookies view in the Configuration dialog box.

#### About this task

The second tab of the Explore: Parameters and Cookies view lets you create and manage custom parameters with formats that AppScan would not be able to recognize automatically.

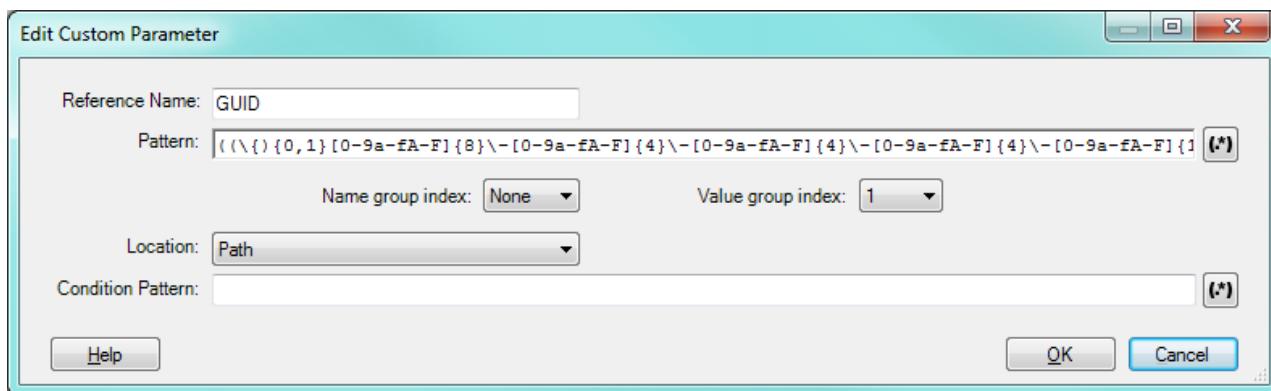
AppScan automatically recognizes parameters in standard HTML format, but if parameters are in other formats (for example within the path or within another parameter), you need to define them to AppScan so it can recognize, follow and manipulate them during scanning.



## Procedure

Click

The **Add Custom Parameter** dialog box opens. Its fields and options are described in the table following.



Setting	Description
Reference Name	Assign the parameter an easily recognizable name. Custom parameters appear on the Application Data tree with the prefix " <u>__patternParameter__</u> " followed by the parameter name.

Setting	Description
Pattern	<p>A regular expression containing one or more groups that defines the parameter.</p> <p>A "group" is a section of the regular expression delineated by parentheses. One of the groups contains the parameter value, and there may also be a group that contains the name.</p> <p>You can open the Expression Test PowerTool by clicking  to help you verify the syntax of the regular expression.</p>
Name group index	<p>(Optional) If the name is included in the regular expression, indicate which group (1,2,3...) contains it.</p> <p>AppScan uses this value to "count through the groups" and locate the parameter name (see example below).</p>
Value group index	<p>Indicate which group (1, 2, 3...) in the above regular expression contains the parameter's value.</p> <p>AppScan uses this value to "count through the groups" and locate the parameter value (see example below).</p>
Location	<p>Indicate which component of the request contains this parameter: Body / Path / Query.</p> <p><b>Note:</b> The selection you make here will apply to both the Pattern and the Condition Pattern (if any), but not to the Response Pattern.</p>
Condition Pattern	<p>(Optional) You can enter a regular expression that defines the whole component (Body, Path or Query) containing the parameter. AppScan will create the parameter only when the whole component matches this pattern; saving scan time.</p> <p>For example, if the parameter is located in the Body, and the Body must be XML, you could set as the Condition Pattern a regex that verifies that the Body starts and ends with XML tags. In cases where it does not, AppScan will not create the parameter.</p>

**Note:** Fields that require or accept a regular expression have this button , which opens the Expression Test PowerTool, to help you verify the syntax of the regular expression.

## Group indexes

To understand the indexing system for groups in the Pattern regular expression, consider the following example:

Pattern: (abc)((def)(ghi))

The groups in this expression would be indexed as follows:

Group 1: (abc)  
 Group 2: ((def)(ghi))  
 Group 3: (def)  
 Group 4: (ghi)

Use the **Name group index** and **Value group index** drop down lists to select the correct groups for the parameter. The selected group is highlighted in the Pattern field.

**Note:** If you change the pattern after indexes have been selected, and the selected index no longer exists in the pattern, a warning appears, but the value is not automatically changed, and you must change it manually.

## Automatic Form Fill view

Automatic Form Fill view of the Configuration dialog box contains the values used to fill forms in your application.

Automatic Form Fill values are the values AppScan uses to fill forms in your application. There are default values for many of these, and they are automatically updated to include any values you enter during a “Recording a login” on page 47.

You can view, add to and edit these values from this view of the Scan Configuration dialog box.

Setting	Details
Enable Automatic Form Fill	When this check box is selected (default), AppScan attempts to automatically fill-in the user input forms of your application during the Explore stage.
Automatic Form Fill Properties List	List of values used by AppScan to fill in forms automatically during the scan. You can add to and edit this list.  For details, see “Form properties”
Username and Password Parameters	Shows the strings, comma separated, that AppScan uses to recognize fields that must be filled with the Username or Password. You can edit this string as required. Add a comma without a space, between strings.
Username and Password Values	Shows the Name and Password used for login if Automatic Login is selected in Login view. You can edit it here. (For details see “Login / Logout tab” on page 46.)
Fill unknown fields with	You can type in a string that AppScan will use whenever it encounters a field that it does not recognize. (The default string is 1234.)

## Form properties

Form Properties view of the Configuration dialog box lists the values that AppScan will use to fill in forms automatically.

### About this task

You can add to or edit the **Form Properties** that AppScan uses to automatically fill forms during scanning.

### Procedure

To add a new definition, click  (or, to edit an existing form property, select it and click 

Column	Description
Description	A name that describes the parameter.
Parameters	The name, or part of the name, that is used to define this parameter in the HTML. When AppScan finds a field with this parameter name it fills in the value supplied.  To enter multiple parameter names in one row, use a comma without a space. <b>Example:</b> Description: Zip Code   Parameters: zip,postal
Value	The user input that AppScan will send for this parameter.
Override	Sometimes the site fills in its own default value for a particular field. In such cases, by default AppScan will use the site's suggestion rather than the one entered here.  Select this check box if you want AppScan to use the value defined here, <i>even when the site offers a different default value</i> .

Column	Description
Match Type	<p>Defines how AppScan will search for the parameter defined here, and how it relates to field-filling options supplied by the site. Options are "Partial" or "Complete."</p> <p><b>Complete:</b> The value will be input <i>only</i> for parameters that exactly match one of the text strings in the Parameters field. (For example, the parameter defined as <code>addr</code>, will not be used for a field named <code>address</code>.) Furthermore, if the field offers a drill-down list of options, this value will be used only if it <i>exactly</i> matches one of the options.</p> <p><b>Partial:</b> This value will be input for parameters that match <i>or partially match</i> one of the parameter strings listed here. (For example, if the parameter is defined as <code>addr</code>, it will also be used for a fields named <code>address</code>, or <code>ADDR</code>.) Furthermore, if the field offers a drill-down list of options, and none of the options exactly match the value entered here, a close match will be used instead.</p>
URL	<p>If this field is left empty, this value will be used for this parameter regardless of URL.</p> <p>If a specific URL is given, this value will be used for only for parameters that occur in this URL. (You can therefore define a parameter one way for one URL and differently for another.)</p> <p>However, if there is no other row with the same parameters and an empty URL, the value of this row is default for all other URLs too.</p>

## Example

"Exporting and Importing Form Properties"

"Importing a saved ASFF file"

### Exporting and Importing Form Properties: About this task

You can save form filler and authentication properties for future use, by exporting your **Form Properties** table as an XML file.

#### Procedure

1. In **Explore: Automatic Form Fill** view, click **Export**.

The **Save As** dialog box that appears lets you save the data as an ASFF file (AppScan Form Filler file). The content is in XML format.

2. Name your file and click **Save**.

Example of ASFF file content:

```
<FormFiller Version="1.0" Enabled="True" DefaultValue="1234"
UseDefaultValue="True">
<Group Name="InternalAppScanUserName" Value="" MatchType="Partial" Action="">
<MatchNames>
<MatchName>user</MatchName>
<MatchName>name</MatchName>
<MatchName>uid</MatchName>
<MatchName>login</MatchName>
<MatchName>usr</MatchName>
</MatchNames>
</Group>
```

### Importing a saved ASFF file:

#### Procedure

1. In **Explore: Automatic Form Fill** view, click **Import**.

A message appears, warning you that by importing an ASFF file, you will delete any current data in the table. If you want to continue, click **OK**.

2. Browse to the relevant ASFF file and click **Open**.

## Form properties in Flash files

Form properties in Adobe Flash files may require special attention.

### About this task

An important element in configuring scans is defining valid values for form fields, especially when they have unusual names. For example if an email field has an unexpected name, AppScan might send input that is not a valid email address format, resulting in the failure of the tests.

In the case of Flash files, the problem is greater, because of the inaccessibility of the instance names (form fields). Accessing these names and configuring valid values for them in "Automatic Form Fill view" on page 75 of the Scan Configuration dialog box.

## Error Pages view

Error Pages view of the Configuration dialog box.

When AppScan gets a 404 error page in response to a test, it generally records the test as failed, since this response indicates that the site successfully recognized the request as illegal. In some cases the opposite is true, and the error page indicates a successful result. In both cases it is important that error pages are correctly defined so that AppScan can recognize them as such.

Web applications and servers often use customized or dynamically generated 404 error pages that may be hard to recognize automatically. AppScan attempts to recognize customized 404 error pages, but in some cases may not succeed in doing so. If it receives a custom error page and does not recognize it as such, it may record the result as positive, where in fact it should be negative, or the other way around. By default the Error Pages list includes standard error page definitions. For each definition type and value are shown.

If your application's error pages are not covered by the definitions in this list, you should add the necessary strings, regexps and URLs that will enable AppScan to recognize your error pages. By doing this you can reduce the number of "false positives" in your scan results. There are two ways you can do this:

- You can define the error page manually before scanning. See "Define a new error page"
- If you have run the Explore stage, you can set a URL that was discovered as an error page. See "Set an error page" on page 78

**Important:** Incorrect error page definitions may cause both "false positive" and "false negative" results, therefore when you add or delete error pages *after* the Test stage of a scan, the scan results should be updated.

- For tests where the previous definition indicated the success of a test, results can be updated by clicking **Apply to Current Results**
- For tests where the previous definition indicated the failure of a test, you must re-test.

*See also:*

"Applying error page changes" on page 79

### Define a new error page

#### Procedure

1. To add a new Custom Error Page definition, click  (or, to edit an existing one, select it and click 

- The **Custom Error Page** dialog box appears.
2. In the **Type** list, select:
    - **String:** Match a string found in the HTML content of your error pages.
    - **Regular Expression:** Match a regular expression found in the HTML content of your error pages.
- You can open the Expression Test PowerTool by clicking  button, to help you verify the syntax of the regular expression.
- **URL:** Match the URL of the response page.
  - **Page:** Match the exact response page. (This method is used when you click **Set as Error Page** for a particular variant in the "Request/Response tab" on page 185.)
3. In the **Value** text box, enter the string, regular expression, or URL (relative path and filename).
- Important:** Incorrect error page definitions may cause both "false positive" and "false negative" results, therefore when you add or delete error pages *after* the Test stage of a scan, the scan results should be updated.
- For tests where the previous definition indicated the success of a test, results can be updated by clicking **Apply to Current Results**
  - For tests where the previous definition indicated the failure of a test, you must re-test.
4. Click **OK**.

Any response that matches the string, regular expression, or URL will be recognized by AppScan as an error page from now on.

## Example

"Custom page filtering examples"

### Custom page filtering examples:

The following table provides examples of how to add pages to the Custom Error Page list.

Type	Value	Adds
URL	/fileNotFound.aspx	Specific file
String	"page not found"	Any page with this string
Regular Expression	(?i)(URL page) (*) not found	Any page with: "URL name not found", "url name not found", "Page page.ext not found", etc.

**Note:** A regular expression (regexp) is a string that describes a set of strings, according to certain syntax rules. The Expression Test PowerTool (**Tools > Expression Test**) can be useful to verify the syntax of your regexps.

If you need additional help writing regexps you may find the following link useful: <http://www.regular-expressions.info/quickstart.html>

## Set an error page

You can set a page listed in the Explore results as a custom error page.

## About this task

You can view the list of URLs that were discovered in the Explore stage in Data view > Pages. If this wrongly lists one of your site's custom error pages, you can set the page as an error page.

## **Procedure**

1. Open the Explore results in Application Data view (F2).
2. In the Result pane, click **Pages**.
3. Right-click on the URL and click **Set as Error Page**.  
You are asked to confirm the update of the results to incorporate the new definition.
4. Click **Start**.  
The URL is added to the Error Page list in the scan configuration, and the results are updated as needed.

## **Applying error page changes**

### **About this task**

If you edit or add to the list of Error Pages after scanning, you must update the scan results to reflect the changes (remove "false positives").

After editing the *Apply to Current® Results* button above the list becomes activated.

## **Procedure**

Click **Apply to Current Results**.

The results are updated to take into account that the new/updated pages are error pages.

## **Multi-Step Operations view**

Multi-Step Operations view of the Configuration dialog box is for testing parts of the site that can only be reached by clicking links in a specific order.

This view is used when parts of the application can only be reached by sending requests in a specific order.

Consider, for example, an online shop where the user visits pages in the following order:

**Page 1:** User adds one or more items to a shopping cart

**Page 2:** User fills in payment and shipping details

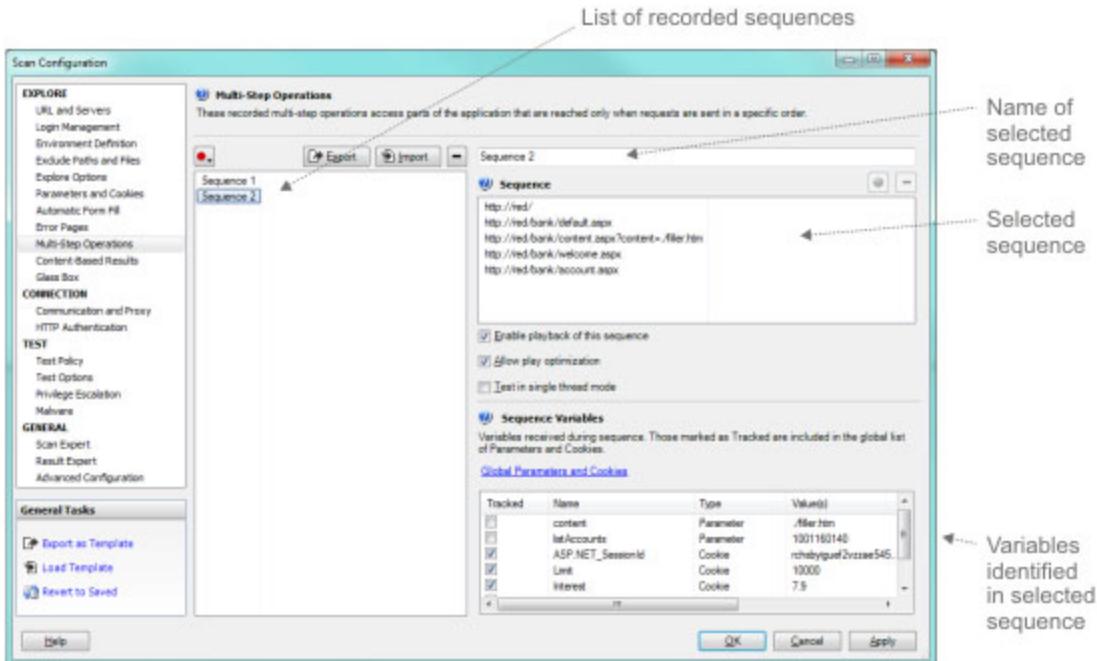
**Page 3:** User receives confirmation that the order is complete

Page 2 can be reached only via Page 1. Page 3 can be reached only via Page 1 followed by Page 2. This is a sequence. In order to be able to test Pages 2 and 3, AppScan must send the correct sequence of HTTP requests before each test.

In the case of the above example you would record a single sequence: Page 1 > Page 2 > Page 3. AppScan would extract the necessary sub-sequences from this sequence, as required. (When testing Page 2 it would send a Page 1 request first; when testing Page 3, it would send Page 1 followed by Page 2.)

**Note:** It is suggested that the number of multi-step operations be limited to five, with no more than 25 steps in any one operation, and no more than 70 steps altogether.

**Note:** Configuring multi-step operations should not be mistaken for manual exploring, and should only be used in cases like the one described above. For more details see "Using AppScan" on page 115



Setting	Details
Record	Click to record a new sequence. If login details have been configured, you can select <b>Login and then Record</b> , or <b>Record (without login)</b> by clicking the down arrow.  For details, see “Record a Sequence” on page 81
Export/Import/-	Export a sequence for use with a different scan; import a sequence from a different scan (if you exported it); delete individual sequences from the current scan.
Sequence Name	The name of the sequence that is selected in the List of Sequences.
Sequence	Shows the links in the selected sequence. <ul style="list-style-type: none"> <li>View any link in the sequence by selecting it and then clicking the browser button</li> <li>Delete any link in the sequence by selecting it and clicking <input type="button" value="Delete"/></li> </ul>
Enable playback	Deselecting this check box will disable playback of the selected sequence temporarily without deleting it from the scan configuration. (The sequence remains in the List of Sequences, grayed out, and you can activate it later by selecting it and then selecting this check box.)
Allow play optimization	When selected (default) AppScan attempts to optimize scan time by avoiding unnecessary playback. You should not disable this setting unless you find that AppScan is missing parts of the application due to play optimization. The “Scan Log” on page 273 can help in determining this.
Test in Single-Thread mode	AppScan may send two or more requests simultaneously, if they don't require the replaying of a sequence between them. If this results in parts of the application being missed, select this check box.
Sequence Variables	Lists variables that were received while recording the sequence(s), and indicates those that AppScan has determined should be tracked. These may be session IDs or other variables. You can change the status of variables in this list to improve how AppScan deals with them (for details see “Sequence variables” on page 82).

“Record a Sequence” on page 81

“Sequence variables” on page 82

"Using AppScan" on page 115

"Scan Multi-Step Operations Only" on page 160

## Record a Sequence

### About this task

If a login sequence has been configured (see "Login / Logout tab" on page 46), there are two options when recording a multi-step operation:

#### Log in and then record

AppScan will log in to the application automatically before starting the recording, but the login will not be recorded as part of the sequence.

#### Record without login

AppScan will begin recording the sequence without logging in. If you log in once the *Record* button is pressed the login steps will be included in the sequence recorded. Use this method if you want AppScan to login each time the sequence is played.

#### CAUTION:

During state induction playback, in-session detection is Off (see "Login / Logout tab" on page 46). This means that AppScan does not verify that it is logged in while playing the sequence. Therefore, if the failure of the sequence will cause the user to be logged out of the application, it is important that login be recorded as part of the sequence (so it will be replayed each time the sequence runs). If this is not done state induction may fail.

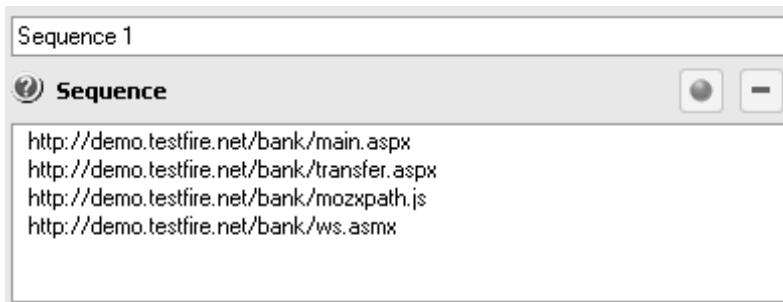
If no login sequence has been configured there is just one option: **Record**.

## Procedure

1. Click the **Record**  button and select one of the record options (see above).  
The browser opens and begins recording.

2. Click on links and fill in fields as necessary to induce the required state. (You can use the **Pause**  button if you want to perform steps that should not be included in the sequence.)  
3. Close the browser.

The sequence appears in the *Sequence* pane (upper right). Sequences are automatically named in order: "Sequence 1", "Sequence 2" etc., but you can rename by typing into the name field (top).



4. To edit the sequence:
  - Give the sequence an easily recognizable name
  - View any step by selecting it and clicking 
  - Remove any unnecessary step by selecting it and clicking 

## Sequence variables

The Sequence Variables pane lists all variables that were received during the recorded sequences. Those variables that AppScan recognizes as session IDs are automatically added to the global list of parameters and cookies that are tracked during the scan (see "Parameters and Cookies view" on page 65). These are marked with a checkmark in the "Tracked" column, to indicate that they are being tracked.

The screenshot shows the 'Sequence Variables' pane with the following details:

**Global Parameters and Cookies**

Tracked	Name	Type	Value(s)
Not tracked	content	Parameter	Random string
Not tracked	listAccounts	Parameter	1001160140
Tracked	ASP.NET_SessionId	Cookie	1chbyiguer2vzzze545...
Tracked	Unit	Cookie	10000
Tracked	Interest	Cookie	7.9
Tracked	CardType	Cookie	Gold
Tracked	UserInfo	Cookie	UserId=100116014&U...

Annotations:

- An arrow points to the first two rows (content and listAccounts) with the label "Not tracked".
- An arrow points to the third row (ASP.NET\_SessionId) with the label "Tracked".
- An arrow points to the value of the ASP.NET\_SessionId row with the label "Dynamic value set as 'Random string'".

## Tracking variables

You can change the status of any variable by selecting it and selecting or deselecting the **Tracked** check box. When a variable is "tracked" AppScan will always send the most recently received version, to help stay in-session.

Note that if you designate a variable as "not tracked" it is removed from the global list of parameters and cookies (see "Parameters and Cookies view" on page 65).

## Defining a variable's format

To help AppScan send (during the Test stage) valid alternative versions of a parameter that varies, you can define the parameter's format:

- Right-click on the parameter and select one of the "Dynamic value" options:
  - Random integer (between 1 - 1000)
  - Incrementing integer (starts from 1, and adds 1 each time)
  - Random string (5 random characters)
  - Date and time (MMddyyHHmmss)
  - Date, time and milliseconds (MMddyyHHmmssSSS)
  - Random email address

If the sequence registers a new user, and the process requires a new email address to be entered each time (so that the site will initiate the registration process rather than an "Existing User" page), define the variable as a "Random email address". Then, each time a test request is sent that includes the variable, a different email address will be used, enabling AppScan to test the registration pages.

**Note:** For more advanced ways of modifying a variable's value, refer to this IBM technote:  
<http://www.ibm.com/support/docview.wss?uid=swg21327313>

## Content-Based Results view

Content-Based view of the Configuration dialog box. You can use this view to define a logical structure for the application tree, if AppScan will not be able to do this based on URL structure.

- If your site content is structured in such a way that the URLs reflect a folder-like hierarchy, the scan results will automatically reflect this, making them easy to navigate.
- If your site uses "breadcrumbs", or other "content-based" navigational methods, so that the URLs do not indicate the user's "location" within the site, it is recommended that you "teach" AppScan how the

site is "logically" structured, so it can present the scan results in an easily understood format, rather than long lists of results under one or two URLs. This is not essential, but will make it easier for you to navigate the results.

For example, the code snippet below has a logical structure **Home** | **Buy** | **Books** and it would be useful to structure the results so that "Books" appears under "Buy", and "Buy" under "Home".

```
<td class="navigation">
  <a href="http://www.onlineshop.com/">Home</a> &gt;
  <a href="http://hub.onlineshop.com/buy?ssPageName=h:h:cat:US">Buy</a> &gt;
    <b>Books</b>
</td>
```

To do this, you define the rules that will enable AppScan to identify and extract the relevant content (in this case "Home", "Buy" and "Books") to construct a content-based tree.

Once you have defined the rules, you can select the Content-Based option in the Application Tree, to display the results using this information. (See "Security Issues: Application Tree" on page 173.)

**Note:** The total number of security issues (shown at the top of the "Result List" on page 22) is a measure of the vulnerable locations in the site, and depends in part on how site is structured. If you define a content-based structure, the total number of issues in the application tree may not be the same as it is for the URL-based application tree (for the same results). When site structure is content-based (rather than URL-based), and content-based view is configured correctly, the issue count in content-based view represents more accurately the number of "vulnerable locations" that exist in the site. The total number of variants (at the top of the Result List in parentheses) is independent of site structure, and does not change between content-based and URL-based views.

## Adding new Content-Based view rules

Instructions and examples of adding rules to define the site's logical structure.

### Before you begin

In Content-Based view of the Scan Configuration dialog box (Explore > Content-Based), you can define content-based structure using two types of definition:

- Logical Content Path (such as breadcrumbs)
- Custom (use regexps to define your own custom nodes)

Doing this enables AppScan to display a logical application tree, instead of listing large parts of the site under a single node. When more than one rule is defined, AppScan attempts to match each URL to the rules in the order they are listed. As soon as it finds a match it includes the URL in the Content-Based tree following that rule and proceeds to the next URL.

### Procedure

1. Add new definitions by clicking  (or edit existing definitions by selecting a definition and clicking 

Setting	Description
Name	Name for this rule.
Description	(Optional) Description for this rule.
Rule Type	Select Logical Content Path or Custom. The remaining fields vary depending on your selection.
<i>Logical Content Path</i>	
Content Path	Regexp that matches the content path in the HTML.

Setting	Description
Separator	Regexp that matches the hierarchy separators in the content path in the HTML.
Node Display Name	Regexp that matches the node names (that appear to the application user and that will be used in the Application Tree).
Condition	(Optional) Regexp that defines the responses that are to be included in this node location.
<i>Custom</i>	
Node Location	Use standard URL format to describe where nodes that fulfil the condition for this rule should be placed in the application tree (for example: /Home/Buy/Books). If the path does not exist it will be created.
Condition	Regexp that defines the responses that are to be included in this node location. If the page content matches this condition it will be included in the Content-Based tree.

**Note:** Fields that require or accept a regular expression have the Expression Test button:  , which opens the Expression Test PowerTool, to help you verify the syntax of the regular expression.

If you need additional help you may find the following link useful: <http://www.regular-expressions.info/quickstart.html>

2. Use the up/down arrows to arrange the definitions in the order they are to be applied.
3. Verify that the check box next to each rule that is to be applied, is selected.
4. Click **OK** to save changes.

## Example

The following tables shows sample content for the two rule types.

*Table 1. Content-Based View rule*

Setting	Sample	Meaning
Content Path	<title>(.*)</title>	All text between the title tags will be used as the basis for the node names.
Separator	[:\-\> ]	These five characters will be treated as separators. Therefore:  <b>&lt;title&gt;Home:Accounts&lt;/title&gt;</b>  and  <b>&lt;title&gt;Home:Plans&lt;/title&gt;</b>  will appear in the application tree as a parent node <b>Home</b> with two child nodes: <b>Accounts</b> and <b>Plans</b>
Node Display Name	^\s*(.*?)\s*\$	Spaces before and after the text will be removed when naming the node.
Condition	<b>Log out</b>	Only pages containing the words "Log out" will be included in the application tree under this rule.

*Table 2. Custom rule*

Setting	Sample	Meaning
Node Location	/root/child/grandchild	All pages that fulfil the Condition are added to the application tree under /root/child/grandchild  <b>Note:</b> If the parent nodes in the specified branch do not exist, they will be created.

Table 2. Custom rule (continued)

Setting	Sample	Meaning
Condition		As this field is empty, all content to which the rule is applied will be added at this node location. Therefore, if this rule appears after the rule above, all pages that do not contain the words "Log out" will be added under the node <b>/root/child/grandchild</b>

## Glass Box view

Glass Box view of the Configuration dialog box.

Glass box scanning uses an agent installed on your application server. This agent monitors server-side activity during the scan, collecting source-code information and other data. This results in a faster and more accurate scan. The relevant glass box agent for the configured starting URL is selected by default, and both of the glass box scanning functions are enabled.

Glass box scanning can discover hidden URLs in the Explore stage, and additional issues and information during the Test stage.

Setting	Details
Use this glass box agent	<p>If the glass box agent has been installed on your application server, and defined in AppScan, you can select it for use in the scan. If you have entered a Starting URL, AppScan attempts to select the appropriate agent automatically.</p> <p>When an agent is selected, AppScan attempts to connect to it, and indicates whether this was successful.</p> <p><b>Note:</b> If you select an agent and get the message "Credentials needed", check that the credentials supplied in Tools &gt; Glass Box Management are correct.</p> <p>If the required server does not appear in the drop-down list, you can define it by clicking the <b>Glass box agent management</b> link.</p> <p><b>Restriction:</b> Only one glass box agent can be selected for use in a scan. If the application being scanned has more than one server, you must scan using each server agent separately.</p>
Use glass box in the Explore stage	<p>(Selected by default.)</p> <p>This function can increase coverage of the site, by examining the server-side source code for the presence of parameters that affect the behavior of the server, but do not appear in the response.</p> <p>Example server-side code:</p> <pre>String debugOn = request.getParameter("debug"); if (debugOn == "true"){     response.getWriter().println(SECRET_SERVER_DATA); }</pre> <p>In this example the developer has left the parameter "debug" in the code. It does not appear in any link on the site, but if an attacker were to send a request containing it, SECRET_SERVER_DATA could be obtained.</p>
Use glass box in the Test stage	(Selected by default.) Select this check box to send glass box tests during the Test stage of the scan. This function can verify the success or failure of certain tests, such as Blind SQL Injection, with greater accuracy, and also reveal the existence of certain security issues that cannot be detected by black box techniques.
Skip equivalent black box tests	(Cleared by default.) This means that both glass box tests and black box tests for the same vulnerability (WASC Threat Classification) are sent. This is because although the glass box tests are generally both more accurate and give more detailed results, occasionally a glass box test may fail while the equivalent black box test succeeds. If the results for your application are unchanged when black box tests are skipped, you can reduce scan time by selecting this check box.

**Note:** By default the two main check boxes are selected. *Deselecting* both of them will disable glass box scanning.

*See also:*

"Glass box scanning" on page 135

"Installing the glass box agent" on page 136

"Defining the glass box agent in AppScan" on page 147

"Scanning with glass box" on page 149

## Communication and Proxy view

Communication and Proxy (if AppScan needs a proxy to access the tested application) view of the Configuration dialog box.

Setting	Details
Communication	
Timeout	Set the time limit for AppScan to wait for a response from the web server.
Number of Threads	Set the maximum number of threads that will be explored/tested simultaneously.  By default this is set to its maximum possible value of 10. If you find that AppScan overloads your network or server with high-speed requests beyond its capacity, you can reduce this number. <b>Note:</b> You cannot change this setting while a scan is in progress.
Proxy	
Proxy settings	If AppScan needs a proxy to access your application, configure it here. By default AppScan is configured to use the Internet Explorer proxy settings.  <b>Use Internet Explorer proxy settings</b> (Default) Select this to use the Address and Port of the Internet Explorer connection. If authentication is required, add username, password and domain below.  <b>Don't use Proxy</b> Select if you do not want to allow AppScan to use a proxy.  <b>Use custom proxy settings</b> Select to define your own proxy settings. Type in the address and port of for the proxy. If authentication is required by the proxy, add username, password and domain below.

## HTTP Authentication view

HTTP Authentication view of the Configuration dialog box.

Use this view to configure platform authentication information and client-side certificate, if required. AppScan supports multiple certificates from the user's personal store, or a single certificate (per scan) in PKCS#12 (PFX) or PEM format.

Setting	Details
HTTP Authentication	If your site requires Basic, Digest, NTLM, Negotiate, or Kerberos HTTP authentication, enter the Username, Password and Domain for AppScan to use during scanning here.
Client-Side Certificate	<p>If your site server uses client-side certificates to verify user identity, AppScan will need them for scanning.</p> <p><b>Don't use certificate</b> (Default)</p> <p><b>PFX</b> Use a single PKCS#12 (PFX) certificate. Click <input type="button" value="+"/> to select <b>Certificate File</b>, and type in the <b>Password</b>.</p> <p><b>PEM</b> Use a single PEM certificate. Click <input type="button" value="+"/> to select <b>Certificate File</b>, <b>Key File</b>, and type in the <b>Password</b>.</p> <p><b>Note:</b> Support for PEM format was deprecated as of AppScan 8.7. It can still be used, but in future versions will not be supported. See also "Convert a PEM Certificate to PFX format."</p> <p><b>Installed certificates (including smart card)</b> Use certificates installed on this machine.</p> <ul style="list-style-type: none"> <li>• If you are not sure which certificates are needed:           <ol style="list-style-type: none"> <li>1. Leave the <b>Identify required certificates automatically</b> check box selected.</li> <li>2. If a PIN is required and you know it, type it into the PIN field, otherwise you will be prompted to enter it when you start the scan.</li> </ol> </li> <li>• If you know which certificate is needed, it is preferable to add it manually:           <ol style="list-style-type: none"> <li>1. Clear the <b>Identify required certificates automatically</b> check box.</li> <li>2. Click <b>Add</b>.</li> <li>3. Select certificate (or certificates) from the Windows Personal Certificates Store (for the current user).</li> <li>4. Click <b>Add</b>. The certificates are added to the table.</li> <li>5. If the certificate requires a PIN, double-click in the column and type it in.</li> </ol> </li> </ul> <p><b>Note:</b> For most smart card certificates the card must be available in the reader during scanning.</p>

## Convert a PEM Certificate to PFX format

As support for PEM is deprecated, it is preferable to convert PEM certificates to PKCS#12 (PFX) format.

### About this task

This conversion can be done using an external tool such as OpenSSL, as described below.

#### Procedure

1. Go to <http://www.openssl.org/related/binaries.html>
2. Download and install version 1.0.1c.
3. Run the following command format from the OpenSSL installation bin folder.  

```
openssl pkcs12 -export -out Cert.p12 -in cert.pem -inkey key.pem -passin pass:root -passout pass:root
```

## Test Policy view

Test Policy view of the Configuration dialog box shows details of the current test policy.

The number of possible AppScan tests for a site can reach the thousands. Rather than manually filter the large number of tests and test variants, you can set a general policy for the type of test you do, or do not, want to be run on your application.

Use Test Policy view to view, edit and manage test policies, that define which tests are included in scans, and to define the policy for the current scan.

Tests are grouped and listed in the upper of the two panes. The Advisory and Fix Recommendation for the selected test appear in the lower pane.

In Test Policy view you can:

- View details of the current policy
- Edit the current policy to create a User-Defined Test Policy of your own
- Import a predefined policy, or a previously saved user-defined policy

Field/Pane/Option	Details
Test Policy	Shows the name of the current Test Policy. Tests are grouped and listed in the upper of the two panes. The Advisory and Fix Recommendation for the selected test appear in the lower pane.
Grouping method	Use the drop-down list to select a grouping method for the tests in the upper pane.
Filter	Use the drop-down list to filter the tests in the upper pane. You can choose: All, DAST Only (Dynamic analysis), IAST Only (Glass box runtime analysis), or SAST Only (Static analysis).
Search	Typing text into the Search field will display only tests that contain the search string. The Magnifying glass drop-down list lets you define whether to look for the string in <i>all</i> test fields, or only specific ones (such as Test Name or CVE ID).
Export	Click to save the current Test Policy so you can load it on another occasion.
Import	Click to load a predefined or user-defined Test Policy (see “Importing a Test Policy” on page 90).
Policy description	The upper-right pane shows the description of the current policy. For user-defined policies this field can be edited.
Test pane	<p>The upper main pane lists all AppScan tests that meet the filter/search criteria. For each test the following information is listed: Name, Variant ID, CVE ID, CWE ID, Severity assigned to the issue (and whether the severity is CVSS or user-assigned), XFIID (X-Force ID), Type, Invasiveness, and WASC threat classification. You can <b>Sort</b> tests by some of these fields, by clicking on the column header.</p> <p>Tests whose check box is selected are included in the current policy. You can edit the policy by selecting/deselecting tests (see “Editing a Test Policy” on page 89).</p>
Update Settings link	<p>This link opens a dialog box that lets you define which types of test can be added to this policy when new tests are added to the database.</p> <p>For details see “Test Policy update settings” on page 91</p>
Advisory and Fix Recommendation tabs	<p>The lower main pane shows the Advisory and Fix Recommendation for the selected test.</p> <p>You can <b>Edit</b> any Advisory to your own specifications, or <b>Reset to Default</b> an Advisory that has been edited (see “Editing advisories and fix recommendations” on page 91).</p>
Policy files	Load an existing Test Policy by clicking one of the <b>Recent Policies</b> , or <b>Predefined Policies</b> , or by clicking <b>Browse...</b> and browsing to the required policy.

## **Editing a Test Policy**

Test Policy view can be used to fine-tune your selected test policy.

### **About this task**

You can fine-tune the current test policy by adding or deleting tests, and also export the changed configuration as a user-defined test policy for future use.

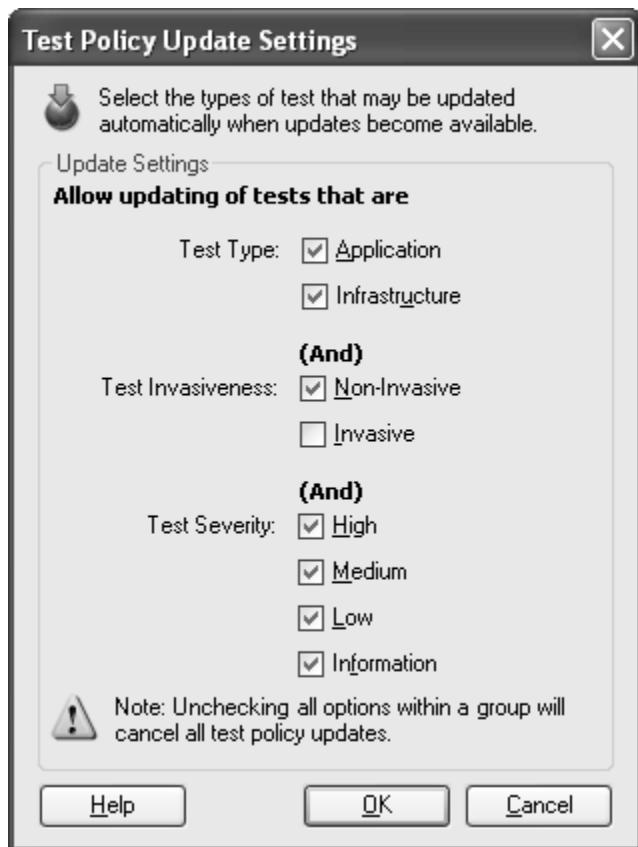
### **Procedure**

1. In the Scan Configuration dialog box, click **Test Policy** (or in Scan Configuration Wizard > **Test Policy**).  
The upper area lists all AppScan tests and indicates which are included in the current scan (check box selected).
2. Include/exclude tests or variants by selecting/deselecting the check box(es). (To view individual variants, click the + icon next to a Test Name.)

**Note:** For each test the following information is listed: Name, Variant ID, CVE ID, CWE ID, Severity assigned to the issue (and whether the severity is CVSS or user-assigned), Type, Invasiveness, WASC threat classification, and XFID (X-Force ID). You can **Sort** tests by any of these fields, by clicking on the column header.

**Note:** The **Search** facility lets you search for tests using free text search.

3. In the Information field at the top right of the dialog, you can edit the description.
4. New tests are continually being added to AppScan's database of tests. By default, all new tests except Invasive tests are added to all user-defined test policies. However, you can define which groups in your policy will be updated: Click **Update Settings**, select/deselect check boxes in the **Test Policy Update Settings** dialog box as required, then click **OK**.



The dialog box contains three groups: Test Type, Test Invasiveness, and Test Severity. Only the tests that belong to a selected category *in all three groups* will be added to the current policy, when new tests are added to your AppScan database of tests. For example: If you select **High Severity**, but deselect **Invasive**, high severity, invasive tests will *not* be added to this policy when updates become available.

5. You can optionally give the scan a name and save it for future use (click **Export**, and save in **.policy** format).
6. Click **OK** to save the changes to the current Test Policy.

## Importing a Test Policy

### About this task

You can load recent, predefined and user-defined test policies.

### Procedure

Do one of the following:

- In the Policy Files area select one of the listed **Recent Policies** or **Predefined Test Policies** (see “**Predefined Test Policies**” on page 91)
- To open a user-defined policy that is not listed in the Policy Files area, click the **Import** button in the upper part of the screen (or click **Browse** in the Policy Files area).

### Results

The selected policy loads, and its name and description appear in upper part of the dialog box.

## Predefined Test Policies

The Policy Files pane at the lower left of Test Policy view lets you select one of the recently used policies, or one of the predefined ones. The predefined policies provide a range of useful policies for common requirements.

Policy Name	Description
Default	Includes all tests except invasive and port listener tests.
Application-Only	Includes all application level tests except invasive and port listener tests.
Infrastructure-Only	Includes all infrastructure level tests except invasive and port listener tests.
Invasive	Includes all invasive tests (tests which might affect the server's stability).
Complete	Includes all AppScan tests.
web Services	Includes all SOAP related tests except invasive and port listener tests.
The Vital Few	Includes a selection of tests that have a high probability of success. This can be useful for evaluating a site when time is limited.
Developer Essentials	Includes a selection of <i>application</i> tests that have a high probability of success. This can be useful for evaluating a site when time is limited.

## Test Policy update settings

Select which types of tests (if any) will be added to the current Test Policy when new tests are added to your AppScan database of tests.

This dialog box contains three groups: Test Type, Test Invasiveness, and Test Severity. Note that only tests that belong to a selected category *in all three groups* will be added to the current policy when new tests become available.

For example: If you select **High Severity**, but deselect **Invasive**, high severity, invasive tests will not be added to this policy when updates become available.

## Editing advisories and fix recommendations

### About this task

You can change the text that appears in the Advisory and Fix Recommendation tabs of the scan results for a test (see "Advisory tab" on page 183). (When you do this the edited file is saved in .xml format on your computer. The exact location is configured in Tools | Options | Preferences Tab, see "Preferences tab" on page 218.) The Advisory and the Fix Recommendation text are stored in the same file. (You can restore the original text at any time using the *Reset to Default* button.)

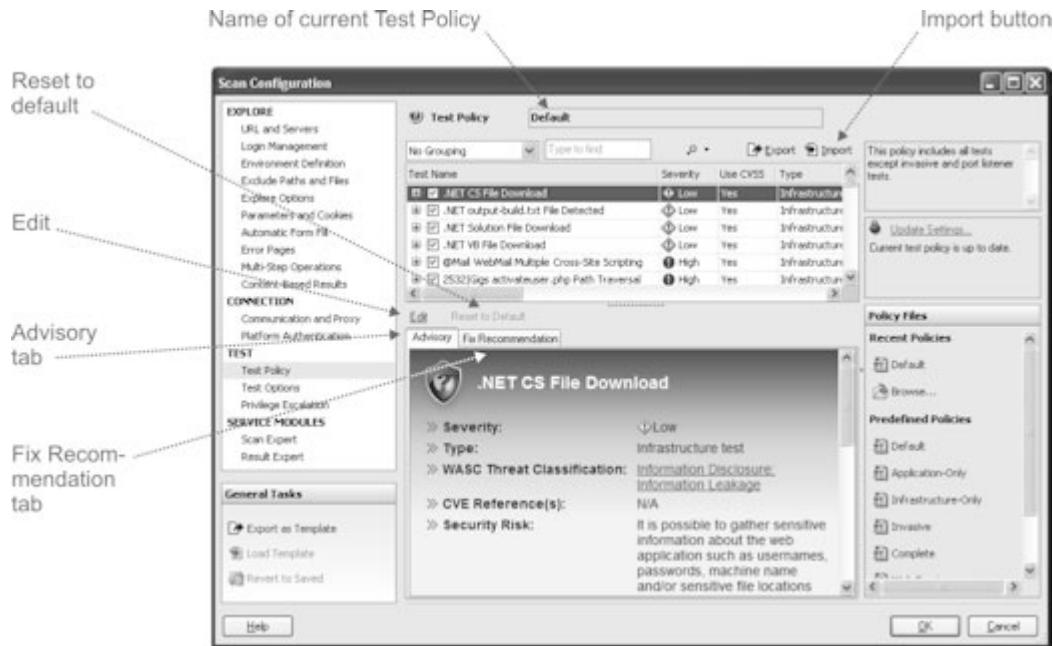
**Note:** If your scan is sent to another computer, the original, *default* advisory for the issue, and not your edited advisory, will be displayed (unless you send the .xml file to the other computer, and have the user save it to the Custom Advisories folder of *that computer*; see "Preferences tab" on page 218).

**Note:** If IBM releases an updated version of an advisory/fix recommendation that you have edited, the updated version will not appear in your AppScan interface (unless you discard the changes you made by clicking the *Reset to Default* button).

### Procedure

1. Select the required test.

The advisory and fix recommendation for the selected test appear in the lower pane.



- Click the **Edit** icon above the Advisory tab. (The advisory and fix recommendations for any particular test are stored in the same file. Therefore it does not matter which of the two tabs is on top when you click **Edit**.)

The advisory file is opened in the embedded XML editor.

- Edit the text as required, then click **OK**.

**Note:** If you use invalid XML syntax, or tags that are not part of the advisory file schema, a warning appears and the changes are not saved.

The XML Editor closes, and the changes appear in the current advisory.

#### Restore default advisories:

#### About this task

When the advisory for a particular test has been changed by the user, the **Restore to Default** button is active when that test is selected.

#### Procedure

Select the relevant test(s) in Test Policy Manager, and click **Restore to Default**.

## Test Options view

Test Options view of the Configuration dialog box.

This view lets you configure various settings that affect the length and thoroughness of the scan. However, the default settings are sufficient in most cases.

Setting	Details
Test Options:	

Setting	Details
Use Adaptive Testing	<p>AppScan can send many thousands of tests to a site. However, in order to reduce scan time, it can send preliminary tests that intelligently determine which are the appropriate tests to send and which can be dispensed with. This is "Adaptive Testing" and it can greatly reduce scan time, without sacrificing efficiency.</p> <p>Clear this check box if you want AppScan to send <i>all</i> its tests to the site.</p>
Allow Multiphase Scanning	<p>AppScan analyzes responses to the tests that it sends your application. From this analysis, AppScan frequently discovers additional content, such as links that were invisible on the first "phase" of the scan. Multiphase scanning enables AppScan to repeat the Explore and Test stages on this newly detected content. (The additional phase is usually shorter, as it involves the new links only.)</p> <p>Multiphase Scanning is configured by default to allow a maximum of 4 scan phases.</p> <p>Note that multiphase scanning applies only when you run a Full Scan. If you use the Explore Only and Test Only functions, the result will be a single-phase scan.</p>
Send Tests on Login and Logout pages	<p>It is recommended to allow AppScan to test login and logout pages, unless your application locks out users who provide illegal input, or the application flow would be altered by AppScan testing these pages.</p>
Do not send session identifiers when testing login pages	<p>(Active only if previous check box is selected.) It is recommended to leave this check box selected, since session identifiers could limit test success when testing login pages. Clear it only if you are sure that valid session tokens are necessary to test your login pages.</p> <p>Note that even when this check box is cleared, <i>some</i> tests are still sent with session identifiers, to prevent false positive results.</p>
Analyze results for inadvertently-triggered issues	<p>When selected, AppScan analyzes each test response for additional security issues over-and-above the specific issue tested for. Deselect this option if the application is very large, or if scans produce a large number of false-positive results.</p>
Include all variants of each issue	<p>(Active only if previous check box is selected.) When selected, AppScan analyzes all variants of each inadvertently-triggered issue; when deselected, only one variant per issue is analyzed. Selecting this check box is not usually necessary, and can significantly increase scan time.</p>
Test for cookie security issues in form submission requests only	<p>When selected (default), AppScan will submit cookie related tests only on cookies used in form submission requests. For higher accuracy (but increased scan time), deselect this check box, and AppScan will submit cookie tests on all relevant HTTP requests.</p>
Save Non-Vulnerable Test Variant Information	<p>During a scan, AppScan sends many thousands of test variants to the site it is testing. The responses to many of these indicate that they do not pose a security threat of any kind, and by default AppScan discards all these "non-vulnerable" results, considerably reducing the volume of the result data.</p> <p>If you select this check box AppScan will save all non-vulnerable variants. A warning will appear this option may reduce AppScan performance and significantly increase the disk space required.</p> <p>For more details see "Non-vulnerable variants" on page 191</p>
Enable JavaScript Security Analysis	<p>Enables static JavaScript analysis that can detect a range of client-side issues, particularly DOM-Based Cross Site Scripting, in client-side JavaScript code.</p> <p>For more details see "JavaScript analysis" on page 94</p>
<b>Issue Management:</b>	

Setting	Details
Apply previous noise classifications to this scan	If in a previous scan you classified one or more issues as "Noise" (not relevant to your application), the same settings are automatically applied to future scans, unless you deselect this check box.  For more details see "Issue state: Open or Noise" on page 175

**Note:** If you make changes to Test Options *after* a scan, you may be prompted to re-scan, as not all changes can be applied to existing results.

## JavaScript analysis

This section describes the basics of how static analysis of JavaScript works, and how the results are presented.

### JavaScript analysis results

AppScan performs static analysis of JavaScript source code, therefore the issues found include source-level trace information showing the vulnerable source code. You can view this trace information in the Issue Information tab of the Detail pane.

Highlighted and numbered lines in the code show, step-by-step, from source to sink, how untrusted data that enters the application gets propagated until it is used in an insecure way.

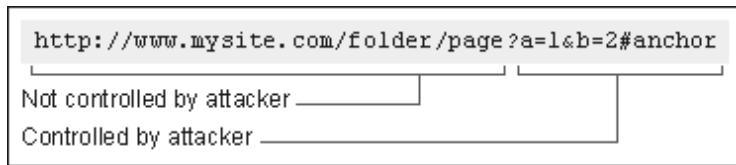
```
http://demo.testfire.net/disclaimer.htm?url=http://www.netscape.com
19
20
21
22  1    var iPos = document.URL.indexOf("url=")+4;
23  1    var sDst = document.URL.substring(iPos,document.URL.length);
24
25  </script>
26  </head>
27  ...
28
29  1    <td>
30  1      <p>This hyperlink allows you to access a third party website:
31  1      <br /><br />
32  1      <b><script>document.write(unescape(sDst));</script></b>
33  1      <br /><br />
34  2      Please read the privacy policy of the linked website, which
35  2      may differ from the privacy policy of the Altoro Mutual website.
36
37
```

AppScan utilizes IBM's advanced "string analysis" technology to eliminate false positives and classify issues more accurately.

### The principals of string analysis

AppScan applies two stages for analyzing each page: taint (or data flow) analysis and string analysis. In the first, AppScan looks for a trace from source to sink that does not go through a sanitizer. If this is found, AppScan validates it in a second step, using string prefix analysis (SPA), a variant of string analysis.

In certain cases, where the source is a URL, or part of it, a trace is only exploitable when the host and path parts of the URL can be controlled by an attacker at the point of the sink. SPA validates these cases by tracking the string operations that manipulate the source URL. SPA assumes that the host and path of the URL are not controlled by the attacker, while the query and/or anchor part are:



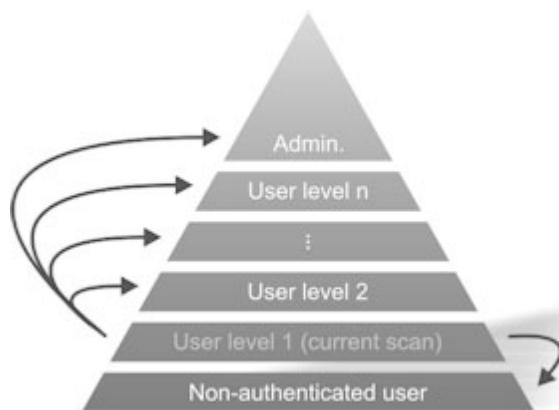
At the point of the sink, SPA checks whether the host and path parts of the URL can be controlled by an attacker. If it is determined that the host and path parts are both fixed and not controlled by an attacker, the issue is eliminated.

If the redirect target begins with the prefix `javascript:` or `mailto:` the issue is classified more accurately as “DOM Based Cross-Site Scripting” or “DOM Based Email Spoofing” respectively.

## Privilege Escalation view

Privilege Escalation view of the Configuration dialog box lets you compare results for different user levels.

### About this task



AppScan can refer to scans that were run using different user privileges, in order to investigate the extent to which privileged resources are accessible to users with insufficient access permissions. This can be done in two ways:

- **By comparison with a higher privileged user:** You point AppScan to scan results that were produced using a higher level of access permissions than the current scan. During the scan AppScan attempts to access the additional links that were available to the higher level user, using the current (lower level) access permissions. The scan results indicate where these attempts were successful.
- **By comparison with a non-authenticated user:** You point AppScan to scan results that were produced *without* user authentication. AppScan then runs a scan using the current authentication and notes the new links it accesses. It then logs out and attempts to access these new links *without* authentication. The scan results indicate where these attempts were successful.

**Important:** Scans being compared must have the same scan configuration, and equivalent Explore data. For example, if the site was explored manually before testing in one of the scans, the *same* Manual Explore must be performed before the Test stage in the scans being compared with it.

### Procedure

1. **(To compare with a higher privileged user:)** In the upper area ("Higher Privilege User Tests"), click  , and browse to a scan that was run with higher access permissions than the current scan.
2. Click **Open**.

3. Type in a name that represents the authentication level used in the scan (for example Guest or Administrator), then click **OK**.  
The selected scan is added to the list, and its role (e.g. Admin, Operator, Visitor) appears in the left column.
4. Repeat these steps to add scans with different authentication levels, as required.

**Note:** You may add more than one scan for Higher Privilege User tests, one for each role. For example, if the current scan is configured with the Username and password of a normal user, you could add two scans to this list: one that was run with Administrator permissions, and one that was run with Supervisor permissions. The results will indicate which user's resources were found to be accessible to the Normal User.

5. **(To compare with a non-authenticated user:)** You can also optionally load the results of a scan run *without* authentication. To do this, in the lower area click  , and browse to the scan results.

## Malware view

Malware view of the Configuration dialog box is for configuring malware testing.

A test for malware consists of two modules with various configuration options. They are configured using this view. Malware test configuration does not affect regular scans.

Setting	Details
<i>Malware Tests</i>	When this checkbox is selected, AppScan checks the application for links to malicious external websites, during the scan. An Internet connection required.
<i>Scope</i>	You can create a list of regular expressions that define URLs you want to <i>exclude</i> from malware testing. Use  to add a regular expression, and in the dialog box that opens  , to open the Expression Test PowerTool which can help you verify the syntax of your regular expressions. If you need additional help writing regular expressions you may find the following link useful: <a href="http://www.regular-expressions.info/quickstart.html">http://www.regular-expressions.info/quickstart.html</a>
<i>Additional Results</i>	The check boxes in this section let you widen the scope of the malware analysis. When AppScan analyzes results for links classified as "Malicious", it can also report on links classified as <b>Unwanted</b> and <b>Harmless</b> , as well as those <b>Unclassified</b> by ISS, that may nonetheless be problematic. Select any issue types you want included in the results. <b>Note:</b> Some links of issue type "Unclassified" may in fact be malicious. <b>Note:</b> Including Harmless links may increase the size of the results significantly.

## Scan Expert view

Scan Expert view of the Scan Configuration dialog box.

Scan Expert runs a short Explore stage before the main scan starts. It analyzes the results of this Explore to evaluate the efficiency of the settings you have configured and suggest changes as necessary. The changes it suggests can then be implemented before the main scan starts, to optimize its efficiency. (See "Scan Expert" on page 134 for more details.)

The settings in this view let you decide how thoroughly Scan Expert explores, whether configuration changes are implemented automatically or manually, and which "modules" of the configuration are included in the evaluation.

**Note:** By default Scan Expert runs automatically before every full scan. You can check or change this setting in **Tools > Options... > Preferences**.

Setting	Details
Scan Expert Behavior	<p><b>Evaluation Limit:</b> Define how quickly (or deeply) Scan Expert explores your application. Deeper exploration may produce a more complete evaluation.</p> <p>You can define a maximum number of URLs to be explored during the evaluation, a time limit, or both. If you limit the evaluation by <i>both</i>, the evaluation will stop when the <i>first</i> of the two limits is reached. (Note: The scan may not always stop exactly at the limit, but shortly after it.)</p> <p><b>After Evaluation:</b> Define whether Scan Expert's recommendations are implemented - and the scan started - automatically, or whether the system waits for you to review the recommendations and implement manually those you approve. (Note that some modules cannot be implemented automatically, so if you select "auto-enabled" the settings for these modules will not be evaluated.)</p>
Modules	<p>The Modules pane lists all modules that Scan Expert is able to evaluate.</p> <p>Select/clear check boxes so that the selected modules are those you want included in the evaluation.</p> <p>Use the up/down arrows to rearrange the order of the modules (this is for convenience only, and does not affect the way they are run).</p> <p>For details, see "Scan Expert modules."</p>

See also: "Scan Expert" on page 134

## Scan Expert modules

A table of Scan Expert modules and their descriptions.

The list is arranged in the order that the modules run. If there is a conflict between the recommendations of two or more modules, the one *lower* in the list prevails.

If you add a module of your own, it is added to the bottom of the list, so that if there are conflicts with existing modules, your new module will prevail.

If necessary you can use the up and down arrows to promote and demote modules in the list.

Module	Description
Parameter-Based Navigation	Checks if the application uses parameter-based navigation, and helps configure the scan correctly for this.
Site Detection	
Server Authentication	Checks for missing or failed NTLM and HTTP Authentication details.
Proxy server	Checks that no Proxy Connection or Authentication errors occurred.
Client-Side Certificate	Detects whether a client certificate is required.
Missing Recorded Login	If the scan is configured to detect login pages automatically, alerts user when no such page is detected.
Detect AJAX Frameworks	Looks for markers of common AJAX Frameworks, which require JavaScript execution.
Detect Session IDs	Checks whether Session IDs were found for the automatic or recorded Login Sequence.
Unfilled Forms	Detects unfilled forms.
Detect Untested Servers	Detects links to servers other than the Starting URL server, in the same domain as the Starting URL. If such links are found, the module will recommend adding to the list of Additional Servers and Domains (Configuration > URL and Servers).
Communication Timeout	Evaluates the accuracy of the Communication Timeout configuration.

Module	Description
Number of Threads	Assesses the ability of the application to withstand multiple scanning threads
Detect Error Pages	Detects custom error pages in the application.
Case Sensitive Path	Checks web application server for case sensitivity.
Suspicious URL	Detects suspicious URLs that should probably be excluded from the scan.
Logout Link Missing	Detects Logout pages.
Depth Limit	Checks if links are unreachable, due to depth limit. If so, it may suggest that depth limit be increased or disabled.
Detect Flash Objects	Detects the use of Flash within the web application.
web Services	Detects the existence of web services in the application.
Check Environmental Settings	If any global questions exist in the Environmental Settings tab, alerts the user if none of these questions have been answered.
WebSphere Portal Detection	Checks if the tested application is based on WebSphere Portal.
Scan Expert Evaluation	Checks whether Scan Expert evaluation was performed successfully.
Hacme Bank Detection	Checks if tested application is McAfee Foundstone Hacme Bank
WebGoat Detection	Checks if tested application is OWASP's WebGoat

"Scan Expert view" on page 96

## Result Expert view

Result Expert can run one or more modules to process scan results and display the additional information in the Issue Information tab of the Detail pane.

- Select/clear check boxes so that the selected modules are those you want included in the process.
- Use the up/down arrows to rearrange the order of the modules (this is for convenience only, and does not affect they way they are run).

Whether or not Result Expert runs at the completion of a scan is set in the Options dialog box (Tools > Options). Options are:

- **Automatic** (starts Result Expert as soon as scan is complete)
- **Prompt** (when scan is complete asks whether to start Result Expert)
- **Don't run**

*See also:*

"Result Expert" on page 161

"Issue Information tab" on page 178

## Advanced Configuration view

The Advanced tab of the **Scan Configuration** dialog box is used to change advanced registry settings that affect specific scans (Scan > Scan Configuration > Advanced tab), and it should only be used by experienced AppScan users, or when instructed to do so by the support team to troubleshoot a problem.

**Tip:** Advanced registry settings that affect AppScan *generally* (rather than a specific scan) are located in the **Advanced** tab of the **Options** dialog box (Tools > Options > Advanced tab).

**Note:** Each setting has an ID that you can use when discussing settings with the Support team. The items in the grid can be sorted by Name or ID, by clicking the relevant column heading.

**Note:** Where the default setting is a regular expression, deleting it altogether will result in the setting being treated as undefined (and not as a regular expression that includes everything).

Name	Description	Possible use cases
<i>Communication:</i>		
Accept-Language request-header value	<p>The string sent for the Accept-Language header in all HTTP requests.</p> <p>If not defined by the user, AppScan will use the value that was sent by the browser the first time in this scan that the user opened it to record the login procedure, a multi-step operation, or to view a page.</p> <p>Note: If you change the default browser, refer to the conditions listed in “Changing the default browser” on page 297</p> <p>Default: en-US</p>	During the Explore stage AppScan might receive an unexpected response, due to the Internet Explorer header value. In such cases you should check which value should be used in the Accept-Language header when interacting with the site, and define it in this setting (or in Internet Explorer).
Custom headers	<p>Allows you to define custom headers to add to all requests that AppScan sends the site.</p> <p>Default: empty</p>	<p>If your site expects specific header content (for example, because access to the site is via a specific client or browser plug-in), define the header(s) here. Each header must be preceded by a delimiter. Between the header and the value must be a colon and a single space.</p> <p>Format: delimiter   header   colon-and-single-space   value</p> <p>Example1: ;Header: Value</p> <p>(In this example the delimiter is ;)</p> <p>Example2: ,Header1: Value1,Header2: Value2</p> <p>(In this example the delimiter is ,)</p>
Force an HTTP request without parameters to every form action	<p>In some cases, server-side logic may behave differently when a form submission <i>without</i> parameters is received.</p> <p>When set to True, AppScan will send an additional request, without parameters, to every form. This may result in the return of custom error pages with links to additional web pages and functionality.</p> <p>Default: True</p>	If you notice, when viewing traffic during the scan, that form submissions without parameters cause timeouts or crash the application, you may decide to set this option to False.
GSC SSL port	<p>This setting defines the port number used in GSC for SSL communication.</p> <p>Default: 443</p>	In the case of links supplied through GSC Explore, AppScan identifies HTTPS based on this port number. If your application uses a different port for SSL communication, define it here. If the correct SSL port is not defined, AppScan will send all tests as HTTP.

Name	Description	Possible use cases
Include AppScan debug headers in all requests	<p>When set to True, an HTTP header is added to all requests sent by AppScan to the site. The header name is "X-AppScan-Debug", and its value includes information about AppScan's reason for sending this particular request (Explore, Test, Login Playback, Server Down Check, and so on).</p> <p>Default: False</p>	<p>Configuring the scan to send "X-AppScan-Debug" headers can be useful in tracking AppScan traffic in external tools such as web debuggers, proxies, analyzers and sniffers.</p> <p>Note: Some sites may reject any requests that include special headers such as this.</p>
Maximum response length	<p>AppScan truncates long responses to avoid memory consumption issues. This setting defines the maximum allowed response length in Megabytes. Longer responses are treated as errors.</p> <p>Default: 8</p>	If AppScan seems to miss links or get out of session, and the application is known to send long responses, increasing the maximum response length may solve the problem.
Remove 'Accept-Encoding' header	<p>AppScan does not support all encodings, and removes the encodings it does not support. If this setting is enabled AppScan will remove the entire header and not just the encodings it does not support.</p> <p>Default: True</p>	If the server rejects AppScan's requests, returns unexpected responses, or AppScan is unable to maintain session, you should check the traffic log and compare the requests AppScan sends with those of your usual browser. If the Accept-Encoding header is different or missing in your browser, you should enable this setting.
Reuse server connections	<p>By default, AppScan closes TCP connections after use, since open connections, and saved data, may affect scan results.</p> <p>When set to True, AppScan leaves connections open after use, and attempts to reuse open connections whenever possible.</p> <p>Default: False</p>	If there are network resource exhaustion errors on the web server, changing this setting to True may solve the problem.
Security package order	<p>AppScan supports Basic, Digest, NTML, Negotiate, and Kerberos HTTP authentication. If you want to force AppScan to use or not use a specific method, or apply an order of preference for method selection when the site/proxy allows more than one, you can edit this value.</p> <p>For example, if you want to allow only NTLM and Basic, and prefer to use NTLM if available, edit this string to: ntlm, basic</p> <p>Default: basic, digest, ntlm, negotiate, kerberos</p>	<p>If your site uses a specific authentication method and AppScan is denied access, defining the required method as the only method can solve the problem.</p> <p>If you want to test your site with specific methods - say Basic and NTLM - you could configure one scan with Basic only, and another with NTLM only.</p>
Slash Normalization	<p>Normalize URLs by replacing two or more consecutive slashes with a single slash.</p> <p>Default: True</p>	If your site URLs utilize consecutive slashes, deactivate this setting.

Name	Description	Possible use cases
Treat error response as valid	<p>AppScan treats error pages differently to regular pages (for example, by not parsing for links). This setting lets you tell AppScan to treat error pages as regular pages for the starting URL only, or always.</p> <p>When set to 0, AppScan treats all error responses as invalid.</p> <p>When set to 1, AppScan treats all error responses to the starting URL (4xx and 5xx) as valid</p> <p>When set to 2, AppScan treats all error responses as valid for both regular pages and the starting URL.</p> <p>Default: 0</p>	<p>If your starting URL response is an error page, change the setting to 1.</p> <p>If you want the scan to extract data from error pages, and test them, change this setting to 2.</p> <p>Note that changing the default setting is likely to affect performance.</p>
<i>Flash:</i>		
Coverage	Defines the "coverage level" for the scan. 1 = faster scan; 2 = more complete coverage  Default: 1	For more complete (but slower) coverage of Flash content, this setting can be changed to 2.
Exclude Instances	Define problematic GUI instances to be excluded from Flash scan. Use this option only when instructed to by Support.	If scan fails or is endless due to repetitive operations. Instances should be comma separated.
File download string	<p>In order to be able to scan Flash movies efficiently it is important that AppScan can identify controls that result in the downloading of a file. Define these here.</p> <p>Note that AppScan will pause after clicking a control containing the strings defined here, for the "Long Operation wait time", even if these strings are not included in the "Long operation string" setting.</p> <p>Default: Download</p>	<p>If the movie includes file downloads, they should be defined here.</p> <p>If there are more than one, separate them with semicolons.</p> <p>Example: Download;Save;Copy</p>
File upload path	Path to a file that AppScan can upload to the application when the movie includes an option to upload a file.  Default: empty	If the Flash movie includes an option to upload a file, define the path to the file, including filename, and make sure the file exists there. Note that you must also define the "File upload string".
File upload string	<p>In order to be able to scan Flash movies efficiently it is important that AppScan can identify controls that allow the user to upload files. Define these here.</p> <p>Note that AppScan will pause after clicking a control containing the strings defined here, for the "Long Operation wait time", even if these strings are not included in the "Long operation string" setting.</p> <p>Default: Upload;Browse</p>	<p>If the movie allows the user to upload files, the text of the links or controls that do this should be defined here.</p> <p>If there are more than one, separate them with semicolons.</p> <p>Example: Upload;Browse;Add</p> <p>Note that the "File upload path" must also be defined.</p>
Flash movie dependency exists	Determines that the site has a Flash movie that contains another Flash movie on which the first movie depends.  Default: False	If the site contains a Flash movie that loads another Flash movie, and the first depends on the second, set this to True.

Name	Description	Possible use cases
Frame rate factor	When playing Flash, increase the frame-rate by this factor (1,2,3,4).  Default: 4	By default AppScan speeds up the frame rate by a factor of four when playing movies during scans. If this is too fast to imitate real use, you can decrease this.
Load movie wait time	The length of time (ms) that AppScan waits for a movie to load before starting to explore it.  Default: 1600	If AppScan starts to explore a movie before the movie has loaded, the results will not be accurate. Therefore, if your movie takes longer than the default time to load you should increase this setting.
Long operation string	After most operations the Flash browser waits the amount of time defined in "Regular wait time" before proceeding, to allow the movie to reach its new state. In the case of operations that are likely to take longer, and that could therefore result in the scan continuing before the new state has been reached, you should define "Long operations" (such as uploading files or logging in) for which the browser will pause for a longer time.  Default: empty	Operations you might want to define as "Long operations" could be "Upload" or "Login" links.  Note that in the case of a file upload, the link you need to define here is the one that actually results in the uploading of the file, not the link that lets the user <i>select</i> a file for uploading.  If you enter two or more strings, separate them with single spaces.
Long operation wait time	Time (ms) for the Flash browser to wait for operations defined in "Long operation string" to complete and reach the new state.  Default: 5000	If a "Long operation" has been defined, you should define its "wait time" here.
Time between samples	Minimum time (ms) to wait between "user operations" (such as mouse clicks), to allow the movie to reach its new state.  Default: 160	If the movie has a lot of animation, the wait required between user-clicks or filling forms may be longer than the default setting. You can increase the amount here.
<i>General:</i>		
Merge redundant tests	When set to True, AppScan sends only a single set of tests on two (or more) requests that are identical except for additional cookies. If set to False, all such requests will be tested separately.  Default: True	Changing this setting to False can impair performance; do so only if advised to by Support.

Name	Description	Possible use cases
Proxy file extension filter	A regular expression that defines file extensions which will be removed from the list of URLs that is saved when you record a login, Manual Explore or Multi-Step operation. If you remove an extension from the regexp., URLs ending with that extension will not be filtered out of recordings.  Default: "\.(zip Z tar t?gz sit cab pdf ps doc ppt xls rtf dot mp(p t d e a 3 4 ga) m4p mdb csv pp(s a) xl(w a) dbf slk prn dif avi mpe?g mov(ie)? qt moov rmi? as(f x) m1v wm(v f a) wav ra au aiff midi? m3u gif jpe?g bmp png tif?f ico pcx css xml)\$"	In rare cases where you need a particular kind of file, such as a CAPTCHA image file, included in your Login recording for reference, you could remove its file extension (in this case jp?g) from the regular expression.
Sanitize logs	Removes sensitive information from logs.  Default: False	If you need to remove sensitive information from logs, activate this option and define the pattern to be removed in the "Sensitive information pattern" option.  Note that changing this setting does not affect logs already generated.
Sanitize reports	Removes sensitive information from reports.  Default: False	If you need to remove sensitive information from reports, activate this option and define the pattern to be removed in the "Sensitive information pattern" option.  Note that changing this setting does not affect reports already generated.
Send all tests through GSC	AppScan can use GSC to send tests on some or all links that were found by GSC.  0 = Send only SOAP messages using GSC 1 = Use GSC to send all tests on links found by GSC 2 = Do not send any tests using GSC  Default: 0	If you didn't define any special security settings when exploring your site in GSC, letting AppScan (rather than GSC) send tests during the Test stage will significantly reduce scan time. However, if during the Test stage many tests get no response, or unexpected error responses, the problem may be due to the difference between how GSC and AppScan send requests. Sending such tests using GSC may solve the problem.
Sensitive Information pattern	A regular expression that defines one or more groups that will be filtered out of logs and reports, if the Sanitize Logs or Sanitize Reports options are activated.  Default: empty	If you need to remove sensitive information from reports or logs, activate the relevant option ("Sanitize logs" or "Sanitize reports"), and define one or more groups in a regular expression here.  The sensitive text is replaced with: **CONFIDENTIAL 1**, **CONFIDENTIAL 2**, and so on.
JavaScript:		

Name	Description	Possible use cases
Automatic JavaScript Execution in Manual Explore	You can configure AppScan to extract <i>all</i> JavaScript-generated links on pages that were manually explored, even those that were not triggered during the Manual Explore.  Default: False	This can be a way of increasing coverage when manually exploring.
Clear cache	If set to True, JavaScript execution (if enabled), will not cache requests sent during login. This may increase scan time and file size.  Default: False	
Fetch external links	When JavaScript Execution is enabled, allow AppScan to fetch external links even if their server is not configured in AppScan as an additional server.  Default: False	HTML pages frequently link to external JavaScript source files, such as Dojo and jQuery source files. If you want JavaScript Execution to access all relevant links discovered during the Explore stage, without adding all the servers to the list of Additional Servers and Domains that AppScan tests, activate this setting.  Note that AppScan will fetch the link, but will not test it, or parse it for new links.
JavaScript and Flash filter	This regular expression defines pages that on which JavaScript will not be executed and Flash not parsed, during the Explore stage of the scan. (It does not limit JavaScript <i>parsing</i> .)  Default: empty	If AppScan repeatedly crashes or freezes on a specific page containing JavaScript or Flash (this can be confirmed by reference to the Traffic Log), defining the page here will solve the problem.
JavaScript link pattern	AppScan uses a variety of patterns to identify links in JavaScript code. If your site uses unusual patterns they should be defined in this regular expression.  Default: empty	If AppScan seems to miss links from your JavaScript code, and your site uses unusual JavaScript link patterns, define one or more patterns here to tell AppScan what to search for.
<i>Localization:</i>		
HTML encoding	Overrides the encoding defined in your site's HTML responses.  Default: empty	If the content of responses in the scan results looks distorted, this may mean that:  1) The encoding method was not correctly identified by AppScan, or  2) The encoding method is incorrectly defined in your site's HTML  To solve 1: Select the correct method in the Explore Options drop-down list.  To solve 2: Enter the correct encoding method here.
<i>Parameters &amp; Cookies:</i>		
Exclude redundant JSON parameters from testing	JSON content type body can contain multiple values of a single parameter that need not be tested individually. When set to True, AppScan attempts to identify redundant values and limit testing to a subset, reducing scan time.  Default: True	If you find that a particular, significant parameter was not tested, change the setting to False.

Name	Description	Possible use cases
Exclude redundant XML parameters from testing	XML content type body can contain multiple values of a single parameter that need not be tested individually. When set to True, AppScan attempts to identify redundant values and limit testing to a subset, reducing scan time.  Default: True	If you find that a particular, significant parameter was not tested, change the setting to False.
Track custom parameters in headers	By default AppScan searches for custom parameters only in the body of responses, not in their headers. If you change this setting to True, AppScan will search in headers too.  Default: False	If AppScan gets out of session due to changes to a parameter in the response header, changing this setting may solve the problem. Note that this may increase scan time.
Track dynamic parameters in Test stage only when inline content exists	Tracking dynamic parameters during the Test stage may result in performance problems. Therefore, by default, dynamic parameters are tracked during the Test stage only in responses with inline content.  Default: True	Change this setting to False only if this kind of tracking is essential.
<i>Server Down Detection:</i>		
Check for "server down" in Explore	Enables the sending of heartbeat requests to check for "Server Down" during Explore stage.  Default: True	If AppScan gets server-down errors during the Explore stage, and the server is not down, this may be due to the server blocking the frequent heartbeat requests.  If AppScan frequently gets out-of-session during scanning, this may be due to the Starting URL being sent to the server as a heartbeat, without cookies.  Deactivating this setting may solve the problem, but note that AppScan will not be able to verify server status.
Check for "server down" in Test	Enables the sending of heartbeat requests to check for "Server Down" during Test stage.  Default: True	If AppScan gets server-down errors during the Test stage, and the server is not down, this may be due to the server blocking the frequent heartbeat requests.  If AppScan frequently gets out-of-session during scanning, this may be due to the Starting URL being sent to the server as a heartbeat, without cookies.  Deactivating this setting may solve the problem, but note that AppScan will not be able to verify server status.
Explore stage reconnect attempts	When AppScan is about to finish the Explore stage but several tests failed due to "server down", and the server is still down, AppScan will try to connect to the server several times.  Default: 5	If you know that your server is sensitive, or you see that the scan stopped due to a communication error while several tests failed due to communication errors, you should increase this number.

Name	Description	Possible use cases
Request retry interval	Interval in seconds before resending failed requests (including failed heartbeat requests).  Default: 1	If you know that you have a poor connection or an unstable server (which would result in false negative results, or reduced coverage), you can increase this interval to reduce the impact.
Request retry limit	Number of times to retry sending failed requests.  Default: 3	Increasing this setting may result in a more efficient scan if your server is unstable or communication is poor.
Server down timeout	When AppScan is unable to connect to the server or gets out-of-session, this setting defines (in seconds) for how long AppScan will try to reconnect or get back into session before stopping the scan.  Default: 95	If you have a slow connection, or your server takes a long time to reload after down time, you might want to increase this setting.
Server-down heartbeat interval	Interval in seconds between "server down" heartbeats.  Default: 3 s  Max: 60 s	If AppScan gets server-down errors during the scan, it may be due to a poor connection or unstable server. Increasing this interval may solve the problem.
Test stage reconnect attempts	When AppScan is about to finish the Test stage but several tests failed due to "server down", and the server is still down, AppScan will try to connect to the server several times.  Default: 5	If you know that your server is sensitive, or you see that the scan stopped due to a communication error while several tests failed due to communication errors, you should increase this number.
<i>Session Management:</i>		
Ad domains	Regular expression describing common web advert domains. Requests sent to these domains when the login sequence is recorded, will be discarded.  Default: ad\d.googlesyndication doubleclick\.net coremetrics\. webtrends\. 112\.2o7\.net view.atdmt.com ad.yieldmanager.com ads.adbrite.com oasn04.247realmedia.com segment-pixel.invitemedia.com"	Since the login sequence is replayed continually during the scan, you can improve scan efficiency by filtering out these unnecessary requests.  Note that if you delete the regexp altogether, no domains will be filtered out.
Clear cookies before playing login	Determines whether cookies are deleted before replaying the login sequence.  Default: True	
Common static parameter values	Common static parameter values. Used for the detection of non-random parameter values, which should not be tracked during login.  Default: true false \bon\b \boff\b \bout\b \checked \enabled \log\s?in \log\s?out \exit \submit \sign ever \disabled \agree	
Disable Explore stage in-session buffering	During the Explore stage: If the response to a request indicates that the user was out-of-session when it was sent, AppScan queues the request to send again. This insures that as much of the site as possible is scanned.  Default: False	If your site throws the user out of session frequently, in-session buffering may result in the Explore stage continuing indefinitely. Setting this option to True will make the Explore stage faster, but may reduce site coverage.

Name	Description	Possible use cases
In-session before multi-step operations	<p>By default AppScan verifies in-session status before replaying multi-step operations.</p> <p>Default: True</p>	<p>If you want to test multi-step operations with a non-authenticated user, or if your multi-step sequence includes login steps, change this setting to False.</p> <p><b>Important:</b> If Configuration &gt; Login Management &gt; Details &gt; Activate In-session detection is <i>deselected</i>, and this advanced setting is set to True (default), the entire login sequence will be replayed before each multi-step operation.</p>
In-session heartbeat interval	<p>Interval in seconds between in-session heartbeats.</p> <p>Default: 5</p>	<p>If AppScan gets out-of-session during the scan, it may be due to a poor connection or an unstable server. Increasing this interval may solve the problem.</p>
Login content type filter	<p>Regular expression defining content types that should be filtered out of the login and multi-step operation sequences. When a login or multi-step operation sequence is recorded, requests whose responses include headers with these content types will be removed from the sequence. Therefore, when AppScan replays the sequence during scanning, requests whose responses contain headers with these content types will not be sent as part of the sequence.</p> <p>Default: text/javascript application/javascript application/x-javascript image text/css</p>	<p>If your site's login procedure, or one of the multi-step operations you have recorded, requires clicking on a link that contains a header with a content type listed here, you should remove it from the regular expression.</p>
Login retry interval	<p>Interval in seconds before re-sending failed login requests.</p> <p>Default: 3</p>	<p>If AppScan gets out of session, and repeated login retry attempts fail, this may be because the server is sensitive to frequent login attempts. Increasing this interval may solve the problem.</p>
Navigational parameter hosts	<p>Regular expression describing hosts. Used for the detection of navigational parameters (by value), which should not be tracked during the login sequence.</p> <p>Default: https?://</p>	<p>If your site uses unusual hosts in navigational parameters, that are not filtered out by the default regexp, you can add them to improve scan efficiency.</p> <p>If you delete the item navigational parameters might not be identified properly.</p>
Navigational parameter scripts	<p>Regular expression describing server-side scripts used in the detection of navigational parameters (by parameter value) which should not be tracked during the login sequence.</p> <p>Default: /[^/\.\.]+\.(htm jsp jsf ws dll asp php do)</p>	<p>If your site uses unusual server-side scripts in navigational parameters, that are not filtered out by the default regexp, you can add them to improve scan efficiency.</p> <p>If you delete the item navigational parameters might not be identified properly.</p>
Navigational parameters	<p>Regular expression describing navigational parameters, which should not be tracked during the login sequence.</p> <p>Default: \bnav url page step redirect request location target argument item article goto node action ctrl control source menu frame command</p>	<p>If your site uses unusual navigational parameters that are not filtered out by the default regexp, you can add them to improve scan efficiency.</p> <p>Modifying this regular expression might result in insufficient scan coverage or improper session tracking.</p>

Name	Description	Possible use cases
Parse in-session page	If set to False AppScan will not parse the in-session page, and will not update tracked parameters or cookies whose values were changed in the in-session page.  Default: True	If your in-session page does not contain tracked cookies or parameters, you can improve performance by changing this setting to False. Note, that if set to False, AppScan will not update cookie/parameter values on the in-session page, which could result in getting out-of-session.
Requests between heartbeats	Following an in-session detection request, AppScan will send at least the number of requests defined here before sending another in-session detection request.  Default: 1	In cases where a slow response from the server results in the scan consisting mostly of in-session detection requests (see Traffic Log), increasing this value can reduce scan time.
<i>Special Patterns:</i>		
Exclude from Automatic Form Fill	Parameter names listed here are excluded from the Automatic Form Filler.  Default: ^CFID __EVENTVALIDATION __VIEWSTATE ^CFTOKEN __EVENTARGUMENT __EVENTTARGET ^BV_	Parameters with very long values may slow down the scan and increase file size. If your application uses parameters with long values, and they are not needed for filling forms, add them to this list.
<i>Tests:</i>		
Account Lockout Threshold	If your server locks users out after n login attempts, AppScan could get locked out during testing, after sending as few as n test requests. If you define the lockout threshold here, AppScan will replay the login sequence after every n-1 tests, to ensure it does not get locked out. If your site does <i>not</i> use a lockout mechanism, or if it is disabled for the scan, leave this setting disabled.  Default: 0= disabled 1=invalid value	If your site has a lockout mechanism and you cannot disable it for the scan, set the lockout threshold here. The lower the threshold, the longer the scan will take.
CSRF: Pattern of meaningful request	By default AppScan tests POST requests, and requests whose response was "Transaction Successful", for Cross-Site Request Forgery.  This setting lets you define additional requests as "meaningful" for Cross-Site Request Forgery vulnerability, in addition to POST requests.  This definition is used in conjunction with "CSRF: Pattern of meaningful response".  Default: ^POST	If you want to test for Cross-Site Request Forgery on GET requests too, change this regular expression.
CSRF: Pattern of meaningful response	By default AppScan tests POST requests, and requests whose response was "Transaction Successful", for Cross-Site Request Forgery.  This setting lets you define additional responses as "meaningful" for Cross-Site Request Forgery vulnerability, in addition to "Transaction Successful".  This definition is used in conjunction with "CSRF: Pattern of meaningful request".  Default: Transaction Successful	If you want to test for Cross-Site Request Forgery on requests that receive other kinds of responses, define them in this regular expression.

Name	Description	Possible use cases
Disable cookie testing	This setting is used to turn off cookie testing altogether.  Default: False	If cookie testing for your application results in a very long scan, you might want to disable it. However, doing so might result in security issues being missed ("false negatives").
Disable cookie testing for static content	Don't test cookies in requests for pages with this extension.  Default: ;htm;html;ahtm;ahtml;  cthm;cthtml;fhtm;fhtml;mht;  mhtm;mhtml;css;css1;js;	In order to reduce scan time and memory consumption, you may want to exclude additional types of page extension. If so, add them to the list of extensions to exclude, separated by a semicolon.
Don't test directory or page	This option lets you define a regular expression to exclude specific directories or pages from attacks during the Test stage. Note that this will only exclude the directories or pages defined, and not any subdirectories or files.  Default: /wps/[^\/*]!/ut/	If you know that certain directories or pages are not vulnerable, or are concerned that testing them might harm site stability, you can exclude them from the scan by defining them in this regular expression.  For excluding a folder <i>and all its sub-folders</i> , see "Exclude Paths and Files view" on page 56
Extract links from all responses	By default during the test stage AppScan will only search for new links in vulnerable responses.  Default: False	If you think AppScan might miss links, or that its coverage isn't sufficient, you can enable this setting, though doing so will increase scan time and file size.
Follow all automatic links	By default AppScan only follows certain types of automatic link* that are likely to contain vulnerabilities. These are: iFrame, Frame and Redirect. You can configure it to follow all types of automatic link.  Note that requests that match the regular expression defined in "Automatic links to ignore" will <i>never</i> be sent, regardless of this setting.  Default: False	If you think your site may contain a vulnerability in other types of automatic link, such as scripts, enable this setting. This will increase scan time and file size.
Login after test	Send tests in a single thread, and send login requests after each test.  Default: False	
Multi-step Operation: Validation limit	The maximum number of consecutive requests from a Multi-Step Operation sequence that will be validated in Cross-Site Scripting tests.  Default: 0	

Name	Description	Possible use cases
Pattern to ignore in response	<p>This regular expression defines sections of the response that AppScan will ignore when analyzing test responses.</p> <p>When comparing responses to decide if a test succeeded, AppScan measures the percentage of change in the entire response. If the response is very long, and the change very small, AppScan might ignore the difference and miss the vulnerability.</p> <p>Default: &lt;input[^&gt;]+(_VIEWSTATE __EVENTTARGET __EVENTARGUMENT __EVENTVALIDATION)[^&gt;]+&gt;</p>	If your site sends responses that include long sections that are not important, defining them here can improve scan accuracy and performance.
Refresh original response interval	<p>Interval in seconds before refreshing the original response (by sending the request again) during the Test stage.</p> <p>One of the ways AppScan decides whether a Test response reveals a vulnerability is by comparing it with the Explore response. When an Explore response is older than the value set here, the Explore request will be sent again, before sending tests, so that an updated Explore response can be used for the comparison. This is important for cases where the Explore response is likely to vary with time, and comparing the Test response to the outdated Explore response might result in a false positive.</p> <p>Default: 30 (seconds)</p>	If you are sure that your application's responses will never become outdated in this way, you can change this setting to zero to reduce scan time. Explore stage requests will then never be resent.
Send port listener tests	<p>By default AppScan doesn't send port listener tests because of the likelihood of failure and the time it takes to validate.</p> <p>Default: False</p>	If the external site is part of your network, so that it is aware of local IP addresses, you might want to activate this type of blind SQL injection test.
Similarity threshold	<p>Certain tests compare the response they receive with the original response, to decide whether the response is similar, and therefore whether they succeeded. For most tests the similarity threshold is 95%. In some test policies it is higher.</p> <p>If you enter a value between 1 and 100 (percent), it will override the thresholds for individual tests.</p> <p>Default: 0 (Ignored)</p>	If your site has no "dynamic" text that causes the similar responses to be slightly different, increasing this percentage may reduce false positive results.
XSS: Test all reflected probes	<p>Usually multiple occurrences of the payload text in a response from the site have the same level of vulnerability, therefore AppScan tests only one of them.</p> <p>Default: False</p>	Set this to True if you want to test <i>all</i> occurrences of the payload text in a single response.

## Automatic links to ignore

Some types of automatic link can be ignored during scanning because they almost certainly do not contain vulnerabilities.

If your application contains automatic links that redirect illegal requests to a main page, rather than an error page, you may be able to reduce scan time significantly by configuring the scan to ignore those links. Note that doing this will not stop AppScan from testing such links *on their own*. It means only that they will not be followed when found during attacks on *other* pages. The Automatic Links to Ignore setting is defined by a regular expression in **Configuration > Advanced Configuration > Tests > Automatic links to ignore**.

During the Test stage, if AppScan finds that it is redirected to a particular link a large number of times, it assumes that the link is one that it should ignore, and adds it to the regular expression in the above setting. The default number of times for a link to be visited before it is ignored is 100,000, but if AppScan finds that memory consumption is nearing its limit, it will automatically lower the link limit to 100.

The link limit is configured in **Configuration > Advanced Configuration > Tests > Automatic link limit**.

The memory consumption limit is configured in **Tools > Options > Advanced > PerformanceMonitor: MaxMemUsage**.

To disable this feature altogether, set the limit to 0.

---

## Scan templates

A scan template is simply a scan *configuration* that has been saved so that you can use it again.

- The Regular Scan template can be used to run a scan without changing any of its configurations (however, you must at least set the Starting URL for the scan).
- You can use one of the supplied “Predefined templates” designed for testing AppScan on a specific test site, or a specific type of site.
- You can create custom “User-defined templates” on page 113, configured for your own specific requirements.

When you save a *scan template*, you save the configuration definitions of the scan (for future use); when you save a *scan*, you save both the configuration *and* the scan results.

## Predefined templates

### About this task

AppScan comes with some predefined scan templates that are configured to produce the best results when testing AppScan on a test site. These templates save you the need to adjust the many configuration options to optimize your scan. (Updates to these templates may be included in AppScan updates from time to time.)

- Regular Scan
- Quick and Light (configured to produce useful results in a short time)
- Parameter-Based navigation
- WebSphere Commerce
- WebSphere Portal
- demo.testfire.net (for scanning IBM's △Altoro Mutual Bank△ website, which has been created for demonstration purposes)
- Hacme Bank
- WebGoat v5

- Worklight® (for scanning in an IBM Worklight server environment)

The table below shows some basic configuration details of some of the predefined templates.

Test Application	Path Exclusion	Path Limit	Explore Method*	Case Sensitive	Login
WebGoat	.*attack\?Num=.*	Off	Depth First	Yes	Username: guest Password: guest
demo.testfire.net	none	5	Breadth First	No	Username: jsmith Password: demo1234

\* For details of *Explore Method* refer to “Explore Options view” on page 62

#### To scan with a predefined template:

#### Procedure

1. Select a template:
  - On the *Welcome Screen*, select one of the *Predefined Templates* area, or
  - On the main screen, click  (or **File > New**), and then in the *Predefined Templates* area select the required template.
2. Click **Save As**, type in a name for the scan, and save the scan.
3. Define the Starting URL for the scan (see “URL and Servers view” on page 43).
4. If applicable, record the login procedure, or supply username and password (see “Login / Logout tab” on page 46).
5. Click 

#### The Parameter-Based Navigation template

Explains the configuration of this predefined template.

The Parameter-Based Navigation template includes the following settings that answer the challenges to scanning of this type of site.

Item	Location	Setting	Comments
Redundant Path Limit	Explore Options	500	Can be further increased if necessary
Depth Limit	Explore Options	10	Enables the handling of ASP.NET 2.0 postback links
Redundancy Tuning Defaults	Parameters and Cookies	1st and 3rd check boxes selected	
Additional parameter	Parameters and Cookies	All Redundancy Tuning check boxes deselected (“Don’t Track”)	<p>The parameter is in the form of a regular expression:  <code>.*(?:i)(page redirect content target EVENTTARGET EVENTARGUMENT goto node action ctrl source .* </code></p> <p>You may need to edit this regexp. if your site uses a navigational parameter that is not listed</p>

See also:

“Sites that use parameter-based navigation” on page 288

"The challenge of parameter-based navigation sites" on page 289

## User-defined templates

When you configure a scan, you can use it for the subsequent scan, and you can save the configuration as a scan template for use in future scans. You can save the template before running the first scan based on it; or you can save a finished scan as a template for future scans.

Also, if you have a scan with interesting results, and you want to rerun the same scan throughout the development and QA process, you can save the scan as a scan template.

"Creating and Saving Scan Templates"

## Creating and Saving Scan Templates

### Procedure

1. Do one of the following:
  - Configure a scan using the **Scan Configuration Wizard**.
  - Configure a scan using the **Scan Configuration** dialog box.
  - Load a saved scan whose configuration you want to save as a template.
2. On the **File** menu, click **Save As**.
3. Choose the **.scant** file type in the **Save As** dialog box.
4. Type in an easily recognizable name for the template.
5. Click **Save**.

## Loading scan templates

### About this task

After you have saved a scan template, you can load it, to run a scan based on the predefined scan configuration of the template, or to change the template.

### Procedure

1. Click **File > New**.
2. In the **Select Scan Template** list, select the template that you want to use.
3. Deselect the **Launch Scan Configuration Wizard** check box.
4. Click **OK**.  
The predefined configurations of the selected scan template are loaded.
5. To start the scan:
  - **Scan menu > Start Scan > [Full Scan / Explore / Test]**.
  - If Start Scan is not available, click **Re-scan > [Full Scan / Explore / Test]**.

## Editing Scan Templates

### About this task

You can edit a scan template, as a persistent modification for all scans that will be based on this template.

You cannot permanently edit the Default scan template. If you change the configuration of Default, the changes are used in the next scan, but they are not kept in the Default scan template.

### Procedure

1. Open a scan template (see "Loading scan templates").
2. Make changes to the configuration settings.

3. Do one of the following:

- **Scan Configuration Wizard**: on the final page of the wizard, select **I will start the scan later**. Then, on the **File** menu, click **Save** or **Save As** and choose the **scant** file type.
- **Scan Configuration** dialog box: click **Save as Template**.

---

## Changing the configuration during a scan

If you change the configuration after you have started a scan, you will need to rerun the scan, or at least the Test stage, to see the effect of the changes. In general:

- If you changed Explore configurations, you should fully re-scan the application, by running Explore and Test (**Scan > Re-Scan > Re-Scan (Full)**).
- If you changed the Test configurations, you need not run the Explore stage again (if it is complete), and can simply re-test the application (**Scan > Re-Scan > Re-Test**).

---

## Chapter 5. Manual Exploring

This section deals with ways in which you can manually explore your application or services before starting the Test stage of the scan, or before starting an automatic scan (that includes both automatic Explore and Test stages).

Manual exploring is when you explore your site to gather data that can be used by AppScan to ensure that when it tests the site it covers parts of the application or services that it might have missed with its automatic Explore stage. This may be because specific user input is required, or because the site responds only to a different type of tool or device. You can manually explore using AppScan, or using it as a recording proxy, or using Generic Service Client (GSC).

In general:

### **AppScan**

is used for manual exploring when the site does not include services. Your actions and input are recorded and used by AppScan in creating tests for the Test stage of the scan.

### **AppScan as recording proxy**

is used when you want to explore manually using your own remote device (such as a mobile phone) or external application (such as a mobile simulator or emulator), with AppScan configured as recording proxy. Use cases may include SOAP web services, or non-SOAP services that do not include a security envelope, or applications that require a different browser. Requests are sent to the site from an external device or application, with AppScan configured as recording proxy. This enables AppScan to record the responses and use them to create tests for the Test stage of the scan.

### **Generic Service Client (GSC)**

is used for exploring web services that have a WSDL file. GSC creates a simple interface for sending requests to the service, and the responses are imported into AppScan and used in creating tests for the Test stage of the scan.

---

## Using AppScan

Manual exploring enables you to explore specific parts of your application, filling in fields and forms as you go. This can be a way of ensuring that particular areas of the site are covered, and that AppScan has the information needed to complete forms correctly.

The AppScan browser opens and you record your actions, links and input data, while you browse through the application. When you stop the recording, AppScan shows you list of the links you crawled, with relevant form filler data that it can now use when exploring or testing the site automatically.

Manual exploring can be done before, instead of, or after the Automatic Explore stage of the scan.

**Note:** URLs discovered in a Manual Explore are tested individually, just like those found in the Automatic Explore. If you need AppScan to test URLs that it will only be able to access by clicking two or more links *in a specific order*, you must record a multi-step operation (see "Multi-Step Operations view" on page 79).

### **Before Automatic Explore**

There are several reasons why you might choose to perform a Manual Explore before an Automatic Explore:

- You want to use the Manual Explore as a way of providing AppScan with data for filling forms, by manually exploring and completing the data as you go.
- You want to ensure that AppScan tests a certain important part of the site.
- You want to scan a specific user process (the URLs, files, and parameters that a user will access given a certain scenario), you can create a manual explore for this process only. This manual explore could be done before you begin a scan.
- Your application uses JavaScripts or Java applets that reveal certain parts of the application only when states (such as Hover and Mouse Over) follow each other in a specific order. Note that this is not the same as a multi-step operation. For a multi-step operation, AppScan must visit links in a specific order; in this case, once AppScan has the link, it can test it in a single step, like any other link.

After performing the Manual Explore, you can continue with an automatic Explore stage (**Explore Only**, or **Full Scan**), so that the scan covers your entire application.

## Instead of Automatic Explore

Sometimes you may choose to perform a Manual Explore *instead of* an Automatic Explore:

- You want to scan only a small part of the site, and prefer to define the parts to test by manually exploring.

After manually exploring you can complete your scan by clicking **Test Only**.

## After Automatic Explore

There are two main reasons why you may choose to perform a Manual Explore *after* an Automatic Explore:

- Your scan resulted in some URLs being categorized as Interactive (see “User Interaction Needed” on page 171), meaning that AppScan was unable to automatically fill in the required data. You can manually explore these URLs to do this.

**Note:** After you do this the URLs are removed from the list of Interactive URLs.

- Your site includes SWF (Adobe Flash) files. AppScan tests these, if configured (see “Explore Options view” on page 62), but if you find it missed certain files, you can identify them for AppScan using Manual Explore. Note that you do not need to explore the movie itself, just click on the SWF file, close Manual Explore, and then rerun Automatic Explore.

## Recording a Manual Explore

### Procedure

1. Click **Scan > Manual Explore > Using browser**

The AppScan embedded browser opens, with the **Record** button  selected (grayed out).

2. Browse the site manually, completing data and clicking on links as you go.

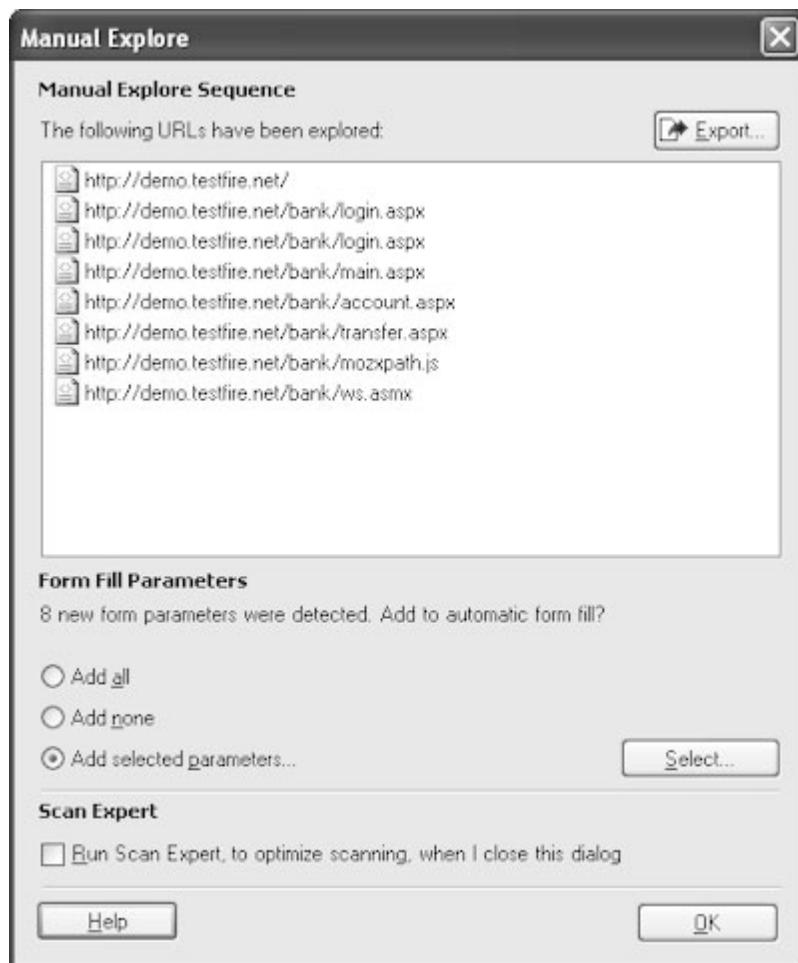
3. When you have finished exploring, click **Pause**  or simply close the browser.

**Note:** You can create a Manual Explore recording that includes two or more unrelated parts of the application: Click **Pause**, browse to a different location, and then click **Record** to resume recording.

The **Explored URLs** dialog box appears, displaying the URLs that you visited during your login.

## Reviewing the manually explored URL list

### About this task



### Procedure

#### 1. Review the list.

If the Manual Explore performed is a standard procedure for the site, you may want to save it for use in future scans. For details see "Exporting Manual Explore data" on page 118.

If you filled HTML forms during the Manual Explore, AppScan determines which information can be added to the Automatic Form Filler settings. If applicable input is found a message appears in the lower half of the dialog-box, and the radio buttons are activated:

[n] new form parameters were detected. Add to Automatic Form Fill?

- If you want *all* your recorded input to be added to the Automatic Form Filler, click **Add All**. Your input is added automatically to the Form Filler information.
- If you do not want your recorded input to be saved, click **Add None**. Even if it is not permanently saved, your manual input has already been applied.
- If you want to add only selected parameters and input values, click **Add selected parameters**, and then click **Select**. The **Add Auto Detected Form Parameters** dialog box appears (see "Add automatically detected form parameters" on page 118).

#### 2. Using the links that you have added, a Scan Expert evaluation may be able to reveal weaknesses in the configuration. To run Scan Expert when this dialog box closes, select the **Scan Expert** check box. (For more details see "Scan Expert" on page 134.)

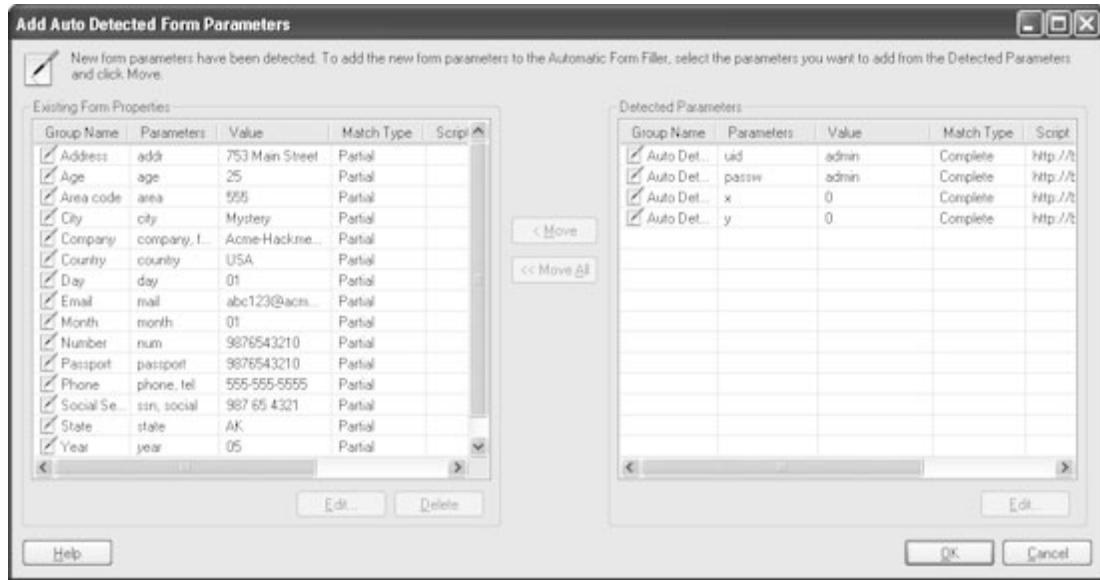
#### 3. Click **OK**.

AppScan creates the test requests of your Manual Explore. This may take a few moments.

## Add automatically detected form parameters

### About this task

The **Add Auto Detected Form Parameters** dialog box lets you select which of the newly detected form parameters you want to add to the Form Properties, to be used automatically in future scans.



### Procedure

1. In the **Detected Parameters** list (right pane), select the rows that you want to add to the Automatic Form Filler (left pane) and then click **Move**; or select any parameter in the list and click **Move All**. Parameters that you move to **Existing Form Parameters** will be available to future scans of these URLs.
2. Click **OK**.

AppScan analyzes the URLs that you crawled and creates tests based on this analysis.

## Exporting Manual Explore data

Standard procedures can be "recorded" in the form of a Manual Explore, and then imported to future scans.

### About this task

If the Manual Explore you performed is a standard procedure for the site, you can save the data (requests sent and responses received) for use in future scans. The data is saved as an Explore Data file (.EXD) in XML format, and can be loaded when required, saving you the trouble of recording the same procedure each time.

### Procedure

1. Record the Manual Explore as in the first steps of the previous procedure.
2. When the **Explored URLs** dialog box appears, click **Export**. A **Save As** dialog box appears.
3. Enter a name for the file and then click **OK**.

**Note:** To view or change the default location to which the file is saved, see **Options > Preferences** (see “Preferences tab” on page 218).

## What to do next

*Related tasks:*

“Recording a Manual Explore” on page 116

“Importing Manual Explore data”

*Back to:*

“Using AppScan” on page 115

## Importing Manual Explore data

If you have saved a standard procedures in the form of a Manual Explore, you can import it to another scan to test it again, even on a different server.

### About this task

You can import a manual explore procedure that you previously saved, to use as part of a new scan. This is useful in the case of standard procedures that need to be rescanned. This can be done even if scanning the same application on a different server (see note in Step 3 below).

### Procedure

1. Click **File > Import > Explore Data**.
2. Browse to the saved EXD file and click **Open**.  
The Import dialog box opens.
3. The **Import with responses** check box is selected by default, and AppScan will analyze these responses when preparing to test the site. If your site has changed in a way that affects its responses, you can clear the checkbox, so that AppScan sends the requests again and gathers up-to-date responses. However, in this case it may be more effective to perform the Manual Explore again.

**Note:** If the file was saved in a version of AppScan that did not save the response data, the check box is cleared and grayed out. You can select **Continue anyway** to import the requests, send them to the site, and gather *new* responses to analyze for testing. However, in this case it may be more effective to perform the Manual Explore again.

4. To start scanning, click **Continue** or **Continue anyway**.
  - a. AppScan analyzes the Explore stage data, to check for host conflicts.  
If the file covers hosts that are not included in the current configuration, the Conflicts in Explore Data File dialog box appears. Resolve any conflicts by selecting an appropriate **Action** for each host:
    - **Ignore:** Do not explore this host.
    - **Add to tested hosts:** Add the listed host to the servers of the scan.
    - **Replace with <hostname>:** For each host defined in the scan, there is an option to replace the conflicting host with host already in the scan configuration.

**Note:** This feature effectively lets you record a manual explore on one host and replay the process on a different server that hosts the same application. This can be used, for example, to record a process on a staging server and replay it on a production server.

- b. The scan begins:

- If the imported data includes responses, the cached data is analyzed in preparation for the Test stage.
  - If the imported data does *not* include responses, a full Explore stage is run first, followed by analysis of the responses to formulate tests for the Test stage.
- c. When the Explore stage and analysis are complete, the scan pauses.
5. At this point you can optionally continue Manual or Automatic Exploring.
  6. To continue with the Test stage of the scan, click **Scan > Continue > Test**.

## What to do next

*Related tasks:*

"Recording a Manual Explore" on page 116

"Exporting Manual Explore data" on page 118

*Back to:*

"Using AppScan" on page 115

## Using AppScan as a proxy server

You can set AppScan to act as a proxy server and then manually explore your application using a third part browsing utility (browser, web services client, Automatic Explore script, mobile phone and so on) through AppScan. In this mode of operation, AppScan records the HTTP/HTTPS traffic going through it, analyzes it, and creates the appropriate tests.

## About this task

**Note:** Exploring with AppScan as a proxy server can be used to explore SOAP and non-SOAP web services, or in cases where the application client is not compatible with Internet Explorer. By configuring your web services client to use AppScan as a proxy, AppScan will collect the requests to the web service as sent by the client. GSC (Generic Service Client), supplied with AppScan, is used for this purpose for SOAP web services. See "Exploring with the External Traffic Recorder" on page 121.

**Note:** In the unlikely event of the AppScan browser producing error messages while your external browser does not, you may use this facility to browse manually using an external browser, with AppScan as proxy. (If this happens, contact the AppScan support team so the issue can be resolved.)

## Procedure

1. Make sure that AppScan is not configured to use Internet Explorer proxy settings. In **Scan Configuration > Connection View**, do one of the following:
  - Select **Don't use proxy**
  - Select **Use custom proxy settings**; type in proxy address, port, and authentication information.

**Note:** If you do not do this the setting is automatically changed to **Don't use proxy** when you start exploring.
2. Find the AppScan listening port. Open **Tools > Options > Recording Proxy tab** (see "Recording Proxy tab" on page 219).  
The section labeled **Communications** shows the port that AppScan is using to listen to traffic going to the web application. It may be a port that AppScan assigned to itself (the port number you are looking for will be gray), or a port that you selected manually.
3. Configure your web browser to use AppScan as its proxy:

In your browser, find the section for configuring a proxy server. Change the hostname or address to the IP address (localhost is usually an acceptable entry) in use by the machine running AppScan, and change the port to the AppScan listening port.

4. Perform a Manual Explore of your application (see “Using AppScan” on page 115 for full details):
  - a. Click **Scan > Manual Explore**, to open the AppScan internal browser.
  - b. Without closing the internal browser, open your external browser.
  - c. Manually explore the application as necessary.
  - d. Close the external browser.
  - e. Close the AppScan internal browser.

## Configuring your browser to use AppScan as a proxy

An example showing how to configure your browser to use AppScan as a proxy.

### Procedure

1. In AppScan, click **Tools > Options > Scan Options tab**.  
Let's say the port is set to **Automatic** and is **4744**.
2. Open **Microsoft Internet Explorer > Tools > Internet Options > Connections tab > LAN Settings**, and select the check box in the **Proxy Server** section.  
The **Address** and **Port** text boxes are enabled.
3. In Internet Explorer, enter the configuration you found in AppScan:  
**Address:** localhost and **Port:** 4744

---

## Using AppScan as recording proxy

You can use AppScan's External Traffic Recorder as a recording proxy, and manually explore RESTful or other non-SOAP web services - or SOAP services that do not require security envelopes - using a mobile phone, simulator, or emulator. AppScan displays the domains and requests in its External Traffic Recorder, and create appropriate tests from the input.

**Note:** For web services that have a WSDL file, and utilize SOAP security envelopes, you should use GSC to explore the service instead (for details see “Using GSC” on page 124).

**Note:** If your application uses man-in-the-middle protection, it is not possible to scan it with AppScan as proxy.

## Exploring with the External Traffic Recorder

This is an example of a simple workflow for a non-SOAP web services scan.

### About this task

This sample workflow shows each conceptual step separately.

**Note:** You can configure and send requests from more than one mobile device through the same instance of the External Traffic Recorder. All domains and requests will be listed together.

### Procedure

1. **Select a template**

Click **File > New** and select a template:

- For IBM Worklight Developer:
  - a. Select the Worklight template

- b. If your Worklight application code uses app authentication (authenticity): In the Worklight server, open the Worklight console, and make sure that application authentication (authenticity) is disabled, or disable it in the application code
- For other environments: Use the Regular Scan template

**Note:** If Internet Explorer is configured on your machine to use AppScan as a proxy, you must verify that AppScan is not configured to use your Internet Explorer proxy settings, as this would result in a loop. To resolve this conflict, in Configuration > Communication and Proxy tab, select one of the other two options:

- **Don't use proxy**
- **Use custom proxy settings**

If you record a Manual Explore using the External Traffic Recorder without doing this, the setting will automatically be changed to **Don't use proxy**. The Test stage Redundancy Tuning is used whether or not the check box is selected.

2. In the wizard welcome dialog box, select External device/client (with AppScan as recording proxy), and then click **Next**.
3. Follow the wizard steps:
  - a. "Recording Proxy" on page 35
  - b. (Optional;) "Communication and Proxy" on page 36
  - c. "SSL Certificate" on page 36
  - d. "Login Management" on page 37
  - e. (Optional;) "Login Management Details" on page 38
  - f. "Test Policy" on page 38
  - g. "Complete" on page 39
4. When the External Traffic Recorder opens with status "Waiting for incoming connections, manually Explore the web service from your device/application:
  - a. Using your device or application, explore the web service.  
As you explore, domains detected are listed in the left pane of the recorder, and URLs are listed in the right pane.
  - b. When finished, in AppScan click **Stop Recording**.
5. Review and edit the Manual Explore data:

#### Domains detected

All domains to which requests were sent are listed, and by default selected for adding to the list of Additional Servers and Domains (Configuration > URLs and Servers > Additional Servers and Domains) so they can be included in the scan. You can deselect any you do not want included in the scan.

**Tip:** You should deselect any domains that belong to other companies.

#### Requests sent

All requests sent by the device to domains *that are selected* (in the left pane), are listed. If you select/clear domains in the left pane, the requests list is updated. You can delete specific requests if they are not needed.

**Tip:** If the total number of filtered requests is more than 200, deleting some of them may produce a more efficient scan.

**Note:** At this stage you can click **Export** to save the Explore data for use on another machine.

6. Click **OK** to close the recorder.  
AppScan takes a few moments to process and display the data.

7. To start the Test stage, click **Scan > Test Only**

The Test stage starts and when complete the scan results are displayed.

Related topics:

- “Recording Proxy tab” on page 219
- Chapter 8, “Results: Security Issues,” on page 173

## External Login Recorder

When you manually explore using AppScan as recording proxy, this recorder displays traffic received during login, and enables you to edit and approve it for use logging in to the application during the scan.

The External Login Recorder opens when you click **Configuration > Login Management > Record > Use external device**, and is used for recording the login sequence. It shows the requests sent.

Item	Description
Proxy Connection Status	Shows whether incoming connections are being recorded, and other status messages.
Listening on port	Shows the current port assigned to the recorder.
Recording Proxy Configuration	Opens Tools > Options > Recording Proxy tab to change the port, or any other recording proxy configurations (for details, see “Recording Proxy tab” on page 219).
Login Requests sent	Shows all requests recorded during the Manual Explore. Requests from domains that are selected in the left pane are shown in black; others are gray.
Stop recording	Stops recording incoming requests, but leaves the dialog box open for review and editing the login sequence.  To delete individual requests from the list that are not relevant to the scan, select the request and then click <input type="button" value="Delete"/>
OK	Closes the recorder.

For the full workflow, see “Exploring with the External Traffic Recorder” on page 121.

See also:

- “Recording Proxy tab” on page 219
- “Redundancy tuning” on page 69

## External Traffic Recorder

When you manually explore using AppScan as recording proxy, this recorder displays domains detected and traffic received, and enables you to control which of these will be tested. A limited version of the recorder is used to record the login sequence.

The External Traffic Recorder opens when you click **Manual Explore > Using external device**.

Item	Description
Proxy Connection Status	Shows whether incoming connections are being recorded, and other status messages.
Listening on port	Shows the current port assigned to the recorder.  To change the port, or any other recording proxy configurations, click <b>Recording Proxy Configuration</b> (for details, see “Recording Proxy tab” on page 219).
Recorded Traffic	

Item	Description
Domains Detected (left pane)	<p>List of all domains detected in the recorded traffic.</p> <p>Select the domains that should be included in the scan. When you close the recorder, all selected domains are added to the Additional Servers and Domains list (Configuration &gt; URL and Servers &gt; Additional Servers and Domains) and will be included in the scan.</p>
Requests Sent (right pane)	<p>Shows all requests recorded during the Manual Explore. Requests from domains that are selected in the left pane are shown in black; others are gray.</p> <ul style="list-style-type: none"> <li>To see <i>only</i> the requests from the selected domains, click the <b>Hide requests from filtered domains</b> check box</li> <li>To delete individual requests from the list that are not relevant to the scan, select the request and then click </li> </ul>
Export	Click to export the recording for use on another machine. This button is active only after the recording has been stopped.
Apply Explore stage Redundancy Tuning	<p>(Selected by default) When selected, Explore stage Redundancy Tuning (Configuration &gt; Parameters and Cookies tab &gt; Redundancy Tuning Defaults &gt; Explore) will be applied to the current recording when you close the dialog box, to help avoid duplicate requests.</p> <p>Clear the checkbox only if selecting it results in cookies from the Manual Explore being missed.</p>
Stop Recording	<p>Stops the recording, leaving the dialog box open for viewing and editing the lists.</p> <p><b>Note:</b> Once you stop the recording you cannot start recording again without discarding the current data.</p>
OK	Closes the dialog box and adds all currently selected domains to the list of additional servers and domains included in the scan (Configuration > URL and Servers > Additional Servers and Domains).

For the workflow, see “Exploring with the External Traffic Recorder” on page 121.

See also:

- “Recording Proxy tab” on page 219
- “Redundancy tuning” on page 69

---

## Using GSC

Generic Service Client (GSC) uses your web service's WSDL file to display a simple interface showing the services available, and lets you input parameters and view the results. Use the GSC interface to explore your web service manually, so that AppScan can use your input to create appropriate tests.

**Note:** GSC is used for exploring web services that are defined with a WSDL file. However, if your service does not have a WSDL file, or the service does not utilize SOAP security envelopes, or if you have an application of your own that you want to use for the Explore stage, you can instead use a mobile device, emulator or simulator, with AppScan's in-built External Traffic Recorder configured as a recording proxy (for details see “Using AppScan as recording proxy” on page 121).

After you have sent some parameters to the service and received responses, and you close GSC, click **Run Tests Only** to start an automatic scan based on the input. (This scan is actually a Test phase only, since the Explore stage is completed during your manual exploration using GSC.)

If SSL is used without a client-side certificate, no special configuration of GSC is needed (if you are using the version of GSC supplied with AppScan 8.0). For configuration in the case of CCS, refer to the GSC Information Center, available from the main GSC toolbar.

**Note:** AppScan remains open while GSC is open, but does not function until GSC is closed.

**Note:** Some of the functionality described in the GSC Help file may not apply when using it through AppScan.

## Exploring with GSC

This is an example of a simple workflow for a SOAP web services scan.

### Before you begin

In order to send tests to web services, GSC must be installed on your system. During AppScan installation you were asked if you want to install it. If you did not install GSC then, you can do so at any time by clicking on the GSC\_Setup.exe file in your main AppScan folder.

### Procedure

1. Open URL and Servers view of the Scan Configuration dialog box, and in the Starting URL field add one of the following:
  - The URL of the WSDL file
  - The path to the WSDL file on your local network, in the form:  
file:///c:/mywsdlfile.wsdl
2. If relevant, select the Case Sensitive check box.
3. Click **OK** to close the Scan Configuration dialog box.
4. Click **Scan > Explore web Services.**

GSC opens and a tree of the web services appears in the left pane. (Click the  icons to expand the tree and see the individual web services.)

5. Explore the services:
  - a. Click on a service in the tree to select it. An interface appears in the right pane for sending requests to the service.
  - b. In the Message tab in the right pane, type in the value to send.
  - c. Click **Invoke** to send the request.  
The result is displayed in the main pane, and the request is added to the Call History pane at the bottom left of the screen.
  - d. Repeat for additional services as required.
6. When you have sent enough requests, close GSC.  
GSC closes and tests are created based on the data.
7. To start the scan, click **Scan > Test Only.**

### Results

When the scan is complete the results are displayed.

### Example

See Chapter 8, "Results: Security Issues," on page 173

## **Scanning a site that includes a SOAP web service as part of the site**

Use this procedure if your site includes both a web Service and other pages that require scanning.

### **About this task**

If your site contains both web services and other pages that need to be scanned, you must explore the web services manually using GSC, but can then let AppScan explore the rest of the site automatically, and then test the *whole* site. In this case you need to provide AppScan both with the URL of the web service WSDL file, and a starting URL for exploring the site.

### **Procedure**

1. Click **Scan > Scan Configuration > URL and Servers**, and in the **Starting URL** field, type in the URL of the WSDL file for the web service.
2. In the **Additional Servers and Domains** area, click  and type in the starting URL for scanning the application.
3. Click **Explore web Services**.

GSC opens.

4. Send requests to the service and then close GSC.
5. Click **Scan > Full Scan**.

AppScan explores the application, and then tests the whole site and presents the results.

*See also:*

Chapter 8, “Results: Security Issues,” on page 173

---

## Chapter 6. Scanning

Learn how to start a scan, and what happens during the scan; how to manually manipulate the Explore stage, and how to export the results of a scan.

### Starting scans

**Note:** If configured, Scan Expert will run before the main scan, when you start the scan. It will evaluate your configuration and suggest changes to optimize the main scan. For details see “Scan Expert” on page 134.

#### Starting scans from the Scan Configuration Wizard

##### About this task

If you use the **Scan Configuration Wizard** (see “Scan configuration wizard” on page 29) to create your scan, the final step of the wizard offers options for starting a scan.

##### Procedure

Select an option (see table below), and click **Finish**.

Option	Click to:
Start a full automatic scan	Start a scan with the configuration you just created in the wizard; start with automatic Explore and proceed automatically to the Test stage.
Start with automatic Explore only	Start the automatic Explore stage of a scan, but do not proceed automatically with the Test stage.
Start with Manual Explore	Open the browser to let you manually explore the application (see “Using AppScan” on page 115).
I will start the scan later	Close the wizard without scanning (for example, if you want to further edit the scan configuration before starting the scan, or if you want to start the scan later).
Start Scan Expert when the Scan Configuration wizard is complete	Scan Expert analyzes your configuration and suggests configuration changes to increase scan efficiency. (For more details see “Scan Expert” on page 134) Select this option to run Scan Expert as soon as you close the wizard.

#### Starting scans from the Scan menu or the toolbar

##### About this task

When AppScan is open, you can start a scan with the current configuration from the Scan menu or the toolbar.

##### Procedure

On the **Scan** menu, or from the **Scan** button  on the toolbar, select one of the following:

- **Full Scan:**  Run a full scan. Explore the application until there are no more unvisited URLs, and then continue automatically with the Test stage. (If multiphase scanning is configured, complete multiple phases as necessary.)

- **Explore Only:**  Explore the application but do not continue with the Test stage. This lets you examine the Explore results, and perform “Using AppScan” on page 115 if necessary (see “Using AppScan” on page 115), before proceeding to the Test Stage.
- **Test Only:**  (Active only if the site has already been explored) Test the site based on the existing Explore results.

## What to do next

**Note:** If you have configured one or more Multi-Step Operations (see “Multi-Step Operations view” on page 79), and they constitute a significant subset of your site that you want to scan, you can run a scan on those sequences only. See “Scan Multi-Step Operations Only” on page 160.

## Starting scans from the Welcome dialog box

When you start up AppScan, the **Welcome** dialog box appears.

**Note:** If the **Welcome** dialog box does not appear on startup, see “General tab” on page 221 to learn how to make the **Welcome** dialog box appear again.

You can:

- “Create a new scan” or
- “Load an existing scan”

### Create a new scan

#### Procedure

1. Select **Create a New Scan**.
2. Do one of the following:
  - If you want to run the scan using one of the predefined configurations, select the **Launch Scan Configuration Wizard** check box. (See “Automatic scan” on page 133.)
  - If you want to make changes to the configuration before starting the scan, *clear* the **Launch Scan Configuration Wizard** check box. (See “Scan configuration wizard” on page 29.)
3. Click on a scan template. If you do not need a specific template, select **Regular Scan**.

### Load an existing scan

#### Procedure

1. Select **Open an Existing Scan**.
2. Select a saved scan from the list; or click **Browse** to open an unlisted saved scan.
3. Click **OK**.

## Starting scans from the New Scan dialog box

You can open the **New Scan** dialog box at any time (**File > New**). It allows you to choose to open the Scan Configuration Wizard or to choose a scan template, upon which you can run a full scan immediately.

Launching the Scan Configuration Wizard:

1. Make sure the **Launch Scan Configuration Wizard** check box is selected or cleared as needed.
2. In the Predefined Templates list, click on the required template. If you do not need a *specific* template, click **Regular Scan**.

**Note:** To use a saved template, click **Browse**, browse to the required SCANT file, and then click **Open**.

## Scan progress

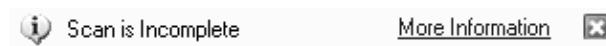
When a scan begins, the **Progress panel** appears below the Result list.



This shows:

- Current URL being explored or tested.
- Percentage completed of the Test stage.
- If this is a subsequent Test phase, the current phase number (see “Automatic multiphase scanning” on page 133).
- Amount of time (mm:ss or hh:mm:ss or dd:hh:mm:ss) since the scan began.

If the scan is stopped, either due to communication issues or by your request, the Scan Notice Panel appears in place of the Progress Panel



The **Scan Notice Panel** shows:

- Communication issues and timeout before automatic resume.
- Notification that scan is incomplete.
- "More Information" link, which opens a dialog box explaining what has been done so far and how to continue.

## Pausing and continuing scans

You can pause a scan at any time and resume it later. If a scan was paused due to connection problems, you can continue the scan after the problem has been fixed.

**Note:** When you pause a scan it may continue for some time before actually stopping.

1. To pause a scan, click the **Pause** button (P) on the toolbar (or click **Scan > Pause**).  
The **Notice Panel** appears, displaying: Scan is incomplete.
2. To continue a scan that was paused, click the **Scan** button (S) on the toolbar, and select one of the options:
  - **Full Scan:** (F) Resume full scan. (Continue with the Explore stage until there are no more unvisited URLs, and then continue automatically with the Test stage. If multiphase scanning is configured, complete multiple phases as necessary.)
  - **Explore Only:** (E) Resume and complete the Explore stage but do not continue with the Test stage.
  - **Test Only:** (T) Resume the Test stage, ignoring any unvisited URLs that had not been explored when the scan was paused.

## Scans stopped due to connectivity issues

A scan may be stopped due to connection problems between AppScan and the server it is scanning, or between AppScan and the local web proxy server.

When a connection problem is detected, a 90 second countdown begins while AppScan waits for the problem to be fixed. If the problem is fixed during the countdown, the Scan Notice Panel disappears and the scan resumes.

If there is a communication problem, the **Notice panel** shows the message:

Trying to connect to: <IP> Explore/Test stops in <n>

- The IP address shown is the IP of the scanned server, the IP of the proxy (if the connection problem is with the web proxy), or a list of IPs, if the scan is connecting to multiple servers and more than one are down.
- The number <n> is the 90 second countdown.
- If the problem is not fixed by the time the countdown reaches zero, the scan stops.
- After the problem is fixed, you can continue the scan where it left off (**Scan > Continue**) or redo the scan (**Scan > Re-scan**).

## Scans stopped due to application issues

### About this task

Sometimes a scan stops and the **Notice panel** displays a link for **More Information**.

### Procedure

#### 1. Click **More Information**.

A message box appears, explaining what to do next. The instructions include how to find the list of Interactive URLs or the list of Broken Links due to missing NTLM authentication. If a scan finds either of these, and is unable to resolve them automatically, the **More Information** message box will summarize the issue.

#### 2. Use the following procedures as appropriate:

- “Dealing with interactive URLs”
- “Dealing with failed requests”

### Dealing with interactive URLs

How to include in the scan URLs that require user interaction.

### About this task

If the scan is stopped because it could not automatically fill in all the input forms, you can complete the Explore manually.

### Procedure

1. Select **Application Data** view.
2. Select the **User Interaction Needed** display.
3. Create a Manual Explore of these URLs (see “Manually exploring interactive URLs” on page 171) and insert the form-filler values.
4. Continue the scan, preferably including another automatic Explore phase.

### Dealing with failed requests

How to deal with requests that failed due to missing NTLM authentication.

### About this task

If the scan encounters broken links due to missing NTLM authentication, the scan will stop.

## **Procedure**

1. Select **Application Data** view.
2. Select the **Failed Requests** display.
3. In the **Scan Configuration** dialog box > **Login/Logout**, enter your platform authentication details.
4. Continue the scan with a re-run of the **Explore** stage.

---

## **Saving and loading scans**

When you save a scan, the configuration and all data that AppScan has collected so far (in both **Explore** and **Test** stages) are saved as a .SCAN file.

After you save a scan, you can load the .SCAN file to resume or rerun the scan.

## **Saving scans**

### **About this task**

You can save a scan any scan that has finished, was paused, or that stopped due to external causes.

## **Procedure**

Click **File** > **Save** (or press **[Ctrl] + S**).

**Note:** To view or change the default folder where scans are saved, click **Tools** > **Options** > **Preferences**. See “**Preferences tab**” on page 218 for details.

## **Automatic scan save**

### **About this task**

You can set AppScan to save data to a .SCAN file automatically during scanning. When this option is selected, AppScan saves both periodically (at an interval you define) and at certain milestones during the scan.

## **Procedure**

Click **Tools** > **Options** > **Scan Options tab** and select the **Automatically Save during Scan** check box. You can also adjust the interval between saves.

The running scan will be saved at the interval you selected. It will also be saved:

- When you start or continue any part of the scan
- When the **Explore** stage ends
- When the **Test** stage ends

When a new scan starts and auto-save is enabled, a message appears, explaining that the scan needs to be saved because auto-save is turned on.

- Click **Yes** to open a **Save As** dialog box to save the scans
- Click **No** to disable the “Automatically save during scan” function for the current scan only
- Click **Disable** to disable the “Automatically save during scan” function for this and future scans

## **Loading saved scans**

### **Procedure**

Click **File > Open** and select a scan.

The **Loading** progress bar opens for a few moments, and then the **Application Tree** of the scan, as explored or tested when you last saved it, appears in the main window.

## **Legacy scan files**

This section describes working with scans saved with a version of AppScan earlier than 8.6 ("legacy scan files").

In AppScan version 8.6 the scan file format was redesigned. If you have a version of AppScan earlier than 8.6 installed on your machine, it need not be removed when you install the current version. The two versions can exist on the same machine, and even be open together.

Due to the new design, scans saved in the earlier version (legacy files) cannot be opened in the newer version. It is possible to convert the Explore stage data from a legacy scan and import it into a new scan with the same configuration, or to use the configuration only to create a new scan, but the Test stage data cannot be converted.

Therefore, if you try to open a legacy scan in the newer version, you are given the following options.

#### **If an earlier version of AppScan is installed on the same machine:**

- **Convert:** Convert Explore stage data and import into a new scan. This may take some time. Note that data from the Test stage is not included, and if you need Test data you will need to run the Test stage on the new scan.
- **New Scan:** Import the configuration only into a new (empty) scan. You can then run a full scan in the new version of AppScan.
- **Open in legacy version of AppScan:** Open the original scan in the older version of AppScan.

#### **If an earlier version of AppScan is *not* installed on the same machine:**

- Import the configuration only into a new (empty) scan. You can then run a full scan in the new version of AppScan.
- Download AppScan Standard 8.5 to view results or convert Explore stage data.

## **Limitations**

The Convert and Import functionality for legacy files has the following limitations:

- For the conversion process to start, the starting URL of the scan being imported must be covered by your license.
- Because of the difference in logic and architecture between the older and newer versions of AppScan, Explore stage results converted to the new format will not look identical to the original results, nor to the results of an Explore stage run in the new version.
- Unexplored links do not appear in the converted scan. If you need these in your results you should run a new scan using the original configuration.
- Scans saved in a version of AppScan earlier than 8.0 cannot be converted or imported to the new format directly. They must first be opened and saved in version 8.0/8.5 and can then be converted or imported into the current version.

## **Program file locations**

The current default path for AppScan is:

... (Program Files Folder) \IBM\AppScan Standard

The default path for versions of AppScan earlier than 8.6 is:  
...\\Program Files Folder\\IBM\\Rational AppScan

---

## Automatic scan

A full automatic scan consists of one or more cycles of an Explore stage followed by a Test stage.

“Automatic multiphase scanning”

### Explore stage

When you begin a full automatic scan, the Explore stage starts first. During this stage AppScan does the following:

- Crawls the application as a user would, from the starting URL you gave in the scan configuration (see “URL and Servers view” on page 43), to every URL in the application that has not been excluded.

**Note:** Scan Configuration lets you filter the Explore stage (see “Exclude Paths and Files view” on page 56). If you exclude specific paths from it, or set specific Explore limits, these filters are applied to the Explore stage.

- Builds the **Application Tree**, a hierarchical model of the URLs in the application.
- Analyzes the explored URLs and generates tests.

### Test stage

During the Test stage, AppScan does the following:

- Logs in to the application.
- Performs preliminary tests on the URLs, which help interpret results.
- Tests URLs by sending requests designed to reveal vulnerabilities.
- Records the response to each request.
- Provides test results.

During the Test stage the Security Issues view of the **Result List** shows the results of the scan.

## Automatic multiphase scanning

### About this task

Some tests may reveal new areas of the web application which are not accessible through normal browsing of the website (for example, directory listings, or the contents of the robots.txt file). If Multiphase scanning is enabled, AppScan will add URLs that were discovered during the Test stage, to the list of URLs to be explored. When the Test stage is complete, AppScan then automatically explores the newly discovered URLs, creates new tests for them, and tests them.

By default, scans may contain up to four phases, if necessary. You can configure AppScan to run between 1 and 10 phases.

**Note:** The multiphase scanning setting only applies if you run a Full Scan. If you activate the **Explore Only** and **Test Only** functions, the result will be a single-phase scan.

### Procedure

1. Open **Scan Configuration** | **Test Options** view.
2. Select the **Enable Multiphase Scanning** check box.

3. In the **Max. Phases** text box, enter the maximum number of phases you want to allow (a number between **1** and **10**; default **4**).

As AppScan tests the site it analyzes test responses for additional URLs that it could not access during the initial Explore stage. It can then run additional Explore and Test stages on these new links. The number you enter here determines the number of times AppScan will do this. (By default, multiphase scanning is enabled for four phases.)

**Note:** The Progress Panel (see “Scan progress” on page 129) shows which phase the scan is currently working on.

**Note:** If you re-scan your application (**Scan > Re-scan**), the phase number restarts at 1.

**Note:** If you save a scan, the current phase number is saved. If you later load that scan to run again, it begins from the saved phase number.

---

## Scan Expert

Scan Expert is a feature that explores your application and network behavior to evaluate the efficiency of your scan configuration. Based on its findings it may recommend changes to the configuration to ensure a more rigorous scan.

Scan Expert connects with your application and runs a short Explore stage on it. During this time the Scan Panel indicates progress, and the Application Tree shows the parts of the application that have been explored, just as during a regular scan. The Scan Expert pane also opens (showing that this is not a regular scan).

Ways of using Scan Expert:

- Scan Expert can be configured to run automatically before a full scan. Depending on how you configure it, it can then:
  - prompt you with recommendations which you can then accept or reject manually, or
  - apply auto-enabled recommendations automatically (not all recommendations are auto-enabled) and proceed with the scan(See “Scan Expert view” on page 96 for details. To enable/disable Scan Expert from running automatically at the start of scans go to **Tools > Options... > Preferences**.)
- Scan Expert can be run on its own at any time, to evaluate your configuration:
  - To have Scan Expert run a short Explore stage on the application and then analyze the responses to evaluate the configuration, click **Scan > Run Scan Expert Evaluation**.
  - If you have already explored the application, you can save time by letting Scan Expert analyse the *existing* data, without running its own Explore stage. Click **Scan > Run Scan Expert Analysis Only**.
- If you try to start a scan and AppScan detects a serious problem with the configuration, it may run Scan Expert even if it is not configured to run automatically, in order to attempt to resolve the issue.

## Scan Expert recommendations

When its Explore stage is complete, Scan Expert analyzes the responses received and recommends changes to the configuration. Depending on how you have configured Scan Expert (see “Scan Expert view” on page 96), the suggested changes may be applied automatically or displayed as an interactive list.

The table below summarizes the options available from the Scan Expert Recommendations list.

Option	Description
Recommendation	Recommendations which AppScan is able to implement without additional user input are automatically selected. Select/deselect the check boxes as required.  Recommendations which require user input to implement, appear with a blue link. Click the link to open the Configuration dialog box at the relevant tab, and provide the necessary input (such as recording a login procedure).
More Info	Click to open the More Info window for the selected recommendation.  This window includes the reasoning behind the recommendation and instructions on how to apply the recommendation manually.
Manual Edit	Lets you apply the currently selected recommendation manually by opening the Scan Configuration dialog box at the relevant view.
Apply Recommendations	Updates the scan configuration to comply with all recommendations whose check box is selected, and closes the Scan Expert pane.
Ignore All	Discards all recommendations and closes the Scan Expert pane.

## Glass box scanning

### Overview

Outlines the principles of glass box scanning and its setup.

While regular scanning looks on the application as a "black box", analyzing its *output* without "looking inside" it; glass box scanning uses an agent installed on the application server to inspect the code itself, during the scan. Hence the term "glass" box. To do this the AppScan glass box agent must be installed on the same server as the application you want to test, not on the local machine where AppScan itself is installed.

Glass box scanning has the following advantages:

- During the Explore stage, glass box scanning can reveal HTTP parameters that affect the server-side but which are not found in responses, and which would therefore not be discovered by black box scanning alone.
- During the Test stage, glass box scanning can verify the success or failure of certain tests, such as Blind SQL Injection, with greater accuracy, resulting in fewer "false positive" results. It can also reveal the existence of certain security issues that cannot be detected by black box techniques.
- Glass box scanning enables AppScan to show you the vulnerability in the actual source code, simplifying both reporting and remediation.

Including glass box scanning adds an extra dimension to the scan in terms of the kind and number of issues that can be found, and the issue information offered.

### To set up and work with glass box scanning:

Task	Description
1. Install agent	Install the AppScan glass box agent on your application server.  <i>Do this once only for a single server.</i> <b>Note:</b> Agents can be installed on more than one server, but only one server can be included in a glass box scan.

Task	Description
2. Define agent	<p>Define the installed agents in AppScan, so it can communicate with them.</p> <p><i>Do this once only for each AppScan machine.</i></p> <p><b>Note:</b> Multiple instances of AppScan (on different machines) can use the same glass box web server agent, but they cannot do so simultaneously.</p>
3. Configure scan	<p>Configure the <i>scan</i> to use the glass box agent you require. By default this is configured automatically, but it can be adjusted in <b>Scan Configuration &gt; Glass Box</b>.</p> <p><i>Do this for each scan.</i></p>
4. Run scan	Scan your application with glass box scanning enabled.
5. Update agent rules	<p><b>Update</b> the server agent rules when prompted to do so by the automatic update process, so that the version of the rules on the web server remains synchronized with the rules on your local AppScan version.</p> <p><b>Note:</b> After running the update process you must restart the web application server.</p>

## For Java platforms

Installing and using the glass box agent on a Java server.

### Installing the glass box agent

This section describes how to set up the server-side glass box agent, to enable glass box scanning.

The AppScan glass box agent must be installed on your application server (or servers), by copying certain files from the AppScan installation to the server.

**Note:** The glass box agent installs and uses a dedicated Java agent (`gbAgent.jar`) to function. If there are other Java agents defined on the web server, you can add the glass box agent to the application server command line (the exact path will depend on your installation):

```
java ... -javaagent:c:\otherAgent\otherAgent.jar  
-javaagent:c:\glassbox\gbAgent.jar ...
```

### System requirements

The following platforms and technologies are supported.

Software	Details
Operating System	<p>Supported Microsoft Windows systems (both 32-bit and 64-bit editions):</p> <ul style="list-style-type: none"> <li>Microsoft Windows Server 2012</li> <li>Microsoft Windows Server 2012 R2</li> <li>Microsoft Windows Server 2008 SP2</li> <li>Microsoft Windows Server 2008 R2</li> </ul> <p>Supported Linux systems:</p> <ul style="list-style-type: none"> <li>Linux RHEL 5, 6, 6.1, 6.2, 6.3, 6.4</li> <li>Linux SLES 10 SP4, 11 SP2</li> </ul> <p>Supported UNIX systems:</p> <ul style="list-style-type: none"> <li>UNIX AIX 6.1, 7.1</li> <li>UNIX Solaris (SPARC) 10, 11</li> </ul>
Java EE container	JBoss AS 6, 7; JBoss EAP 6.1; Tomcat 6.0, 7.0; WebLogic 10, 11, 12; WebSphere 7.0, 8.0, 8.5, 8.5.5

## Before you start

It will save you time if you have the following information ready before starting to install. You may need to consult your web application server administrator to get the information.

- Your server's operating system (Windows, Linux, Unix)
- Your Java EE application server (WebSphere, WebLogic, Tomcat, JBoss), and the mode in which it is installed (standard or as an operating system service)
- The web applications deployment location of your Java EE application server (for example: D:\apache-tomcat-6.0.32\webapps)
- The location of the Java runtime used by your Java EE application server (for example: C:\Program Files (x86)\Java\jre6)
- The credentials for your Java EE application server administration (for the purpose of deploying a new web application)

**Note:** If the server will be running in secure mode, you must add special permissions. See “Permissions needed to work in secure mode” on page 145

### Install automatically using the glass box agent installer:

This section describes how to install the glass box agent automatically through the user interface.

#### About this task

When AppScan is installed, the files needed for installing the server agent are saved in a dedicated folder on your machine. To perform this task you will need access to this folder, and to your application server.

#### Procedure

1. Open ...\\Program Files\\IBM\\AppScan Standard\\Glass box  
The exact path will depend on where you installed AppScan.
2. Copy the relevant setup file to the web server:
  - **Linux server:** Copy the file GB\_Java\_Setup.bin
  - **Windows server:** Copy the file GB\_Java\_Setup.exe
3. Launch the GB\_Java\_Setup application, and follow the online instructions. During this process you will be asked to:
  - Select your web application server. If the server is not listed (for example JBoss Service, Tomcat Service, or WebLogic Service), select **Other**.
  - Define a username and password for the agent. Later, when the agent is defined in AppScan, these credentials will be needed to enable communication between it and the agent. Only English ASCII characters can be used.

**Note:** If you selected **Other**, continue as for relevant *manual* installation, starting from Step 5.

**Note:** You may also be asked to supply the path to the relevant JAVA\_HOME (JDK) or JRE\_HOME (JDK or JRE) folder for your framework.

**Tip:** The installer interface language options include only languages supported by your operating system. If you want to run the installer in a different language, you can start the installer through the command line, and add a flag for the required language. For example, to run the installer in Japanese on an English OS, run the command GB\_Java\_Setup.bin -l ja

- A starting script is created in the glass box agent installation folder
- If you have a desktop, a shortcut is created for starting the server with the glass box agent activated.

**Important:** In order to enable glass box scanning, the application server must be started using one of these, as this starts the server with the glass box agent activated.

### Install using the command line:

This section describes how to install the glass box agent through the command line, automatically or interactively.

#### About this task

When AppScan is installed, the files needed for installing the server agent are saved in a dedicated folder on your machine. To perform this task you will need access to this folder, and to your application server.

#### Procedure

1. Open ...\\Program Files\\IBM\\AppScan Standard\\Glass box  
The exact path will depend on where you installed AppScan.
2. Copy the relevant setup file to the web server:
  - For a Linux server, copy the file GB\_Java\_Setup.bin
  - For a Windows server, copy the file GB\_Java\_Setup.exe
3. Do one of the following:
  - **To install interactively:** Run the following command line:  
`GB_Java_Setup.bin/GB_Java_Setup.exe -i console -DMODE=interactive`  
You will then be able to fill in the information through the command line. (At any step of the process you can return to the previous step by typing back).
  - **To install automatically:** Run a command line containing *all* of the following arguments *that apply to your server*, separated by spaces.

Command	Description
GB_Java_Setup.bin (Linux) or GB_Java_Setup.exe (Windows)	The setup file.
-i console	
-l en	Sets the language for the installation. Use the code for your language ("en" is for English).
-DCHOSEN_INSTALL_SET=JBoss / WebSphe / Tomcat / WebLogic / Other	The type of server on which you are installing the agent. <b>Note:</b> If your server is not one of those listed (for example, if it's JBoss Service, Tomcat Service, or WebLogic Service), set <b>Other</b> .
-DUSER_INSTALL_DIR=value	Sets the path where the agent will be installed.
-DGLASS_USERNAME=value	Sets a username for accessing the agent. Only English characters and numbers can be used.
-DGLASS_PASSWORD=value	Sets a password for accessing the agent. Only English characters and numbers can be used.
-DWEBLOGIC_PATH / -DJBOSS_PATH / -DTOMCAT_PATH / -DWEBSPHERE_PATH=value	Sets the path of the web server installation directory. <b>Examples</b> (actual location depends on your system):  WebLogic: C:\\weblogic\\user_projects\\domains\\base_domain\\ JBoss: C:\\jboss-6.0.0\\ Tomcat: C:\\apache-tomcat-6.0.32\\  WebSphere: C:\\Program Files\\IBM\\WebSphere\\AppServer\\profiles\\AppSrv01\\

Command	Description
-DJBOSS_SERVER_NAME=value / -DWEBSPHERE_SERVER_NAME=value	The server name.
-DWEBLOGIC_USERNAME / -DWEBSPHERE_USERNAME=value  -DWEBLOGIC_PASSWORD / -DWEBSPHERE_PASSWORD=value	Username and password to access the application server
-DWEBLOGIC_LIB=value	The path to the LIB folder, for example:  C:\weblogic\wlserver_10.3\server\lib\
-DWEBLOGIC_TARGET=value	The server/target name.
-DWEBLOGIC_PORT=value	The administration port.
-DSELECTED_JBOSS_TYPE=value	Use one of the following values:  JBoss EAP Managed Domain  JBoss EAP Standalone  JBoss EAP Application Server

#### Example for WebSphere:

```
GB_Java_Setup.bin -i console -l en -DCHOSEN_INSTALL_SET=WebSphe
-DUSER_INSTALL_DIR=/opt/glass_box -DGLOSS_USERNAME=jsmith
-DGLOSS_PASSWORD=1234
-DWEBSPHERE_PATH=/opt/IBM/WebSphere/AppServer/profiles/AppSrv01
-DWEBSPHERE_SERVER_NAME=server1
-DWEBSPHERE_USERNAME=admin -DWEBSPHERE_PASSWORD=admin_pw
```

**Important:** Whenever the server is started, it must be started using the desktop shortcut or the starting script in the glass box agent installation folder, so that the glass box agent is active.

#### Install manually on a JBoss server or a JBoss Service server:

This section describes how to manually install the glass box agent on a JBoss or JBoss Service server.

##### About this task

When AppScan is installed, the files needed for installing the server agent are saved in a dedicated folder on your machine. To perform this task you will need access to this folder, and to your application server.

##### Procedure

1. Open ...\\Program Files\\IBM\\AppScan Standard\\Glass box  
The absolute path will depend on where you installed AppScan.
2. Locate GB\_Java\_Manual\_Setup.zip and copy it to your web server.
3. Extract the contents of this folder to your preferred location on the web server.
4. Define a username and password for the agent (use only English ASCII characters):
  - **Linux server:** Run AgentCredentials.sh <username> <password>

**Note:** AgentCredentials.sh must have execution permissions.

- **Windows server:** Run AgentCredentials.bat <username> <password>

5. Deploy GBootStrap web application:
  - a. Log in to the JBoss Administration Console. The default location is http://<server\_name>:<port\_number>/admin-console/

- b. Click **Applications > web Application WARs**, then click **Add a new resource**.
- c. Type in the path to **GBootStrap.war** (found in the glass box folder you extracted), then click **Continue**.

A message confirms that the resource was created successfully.

6. Shut down the JBoss server/service.
7. Supply the Java agent **gbAgent.jar** (found in the folder you extracted) as a JVM argument by doing *one* of the following:
  - **Through the environmental variable:** Configure JBoss's JVM through the **JAVA\_OPTS** environmental variable (if it does not exist, create it), with the value:  
`-javaagent:<path_to_gbAgent.jar>/gbAgent.jar`
  - **Through batch/sh scripts:** Pass arguments to the JVM running JBoss by editing the configuration files that are shipped with the server:

#### **Linux server:**

- a. In the JBoss folder (usually located in `<path_to_JBoss_folder>/bin`), open `run.conf`
- b. Locate a line that begins with `JAVA_OPTS` and add the following line below it:  
`export JAVA_OPTS = $JAVA_OPTS -javaagent:<path_to_gbAgent.jar>/gbAgent.jar`
- c. Save and close the file.

#### **Windows server:**

- a. In the JBoss folder (usually located in `<path_to_JBoss_folder>\bin`), open `run.conf.bat`
- b. Locate a line that begins with `set JAVA_OPTS=` and add the following line below it:  
`set JAVA_OPTS = %JAVA_OPTS% -javaagent:<path_to_gbAgent.jar>/gbAgent.jar`
- c. Save and close the file.

**Important:** Ensure that `JAVA_OPTS` is initialized only once, and prior to the lines above, so that other `JAVA_OPTS` assignments do not overwrite `-javaagent`.

**Note:** To add arguments to `JAVA_OPTS`, use the `%JAVA_OPTS%` / `$ JAVA_OPTS` convention.

8. Restart the JBoss server/service.

### **Install manually on a Tomcat server:**

This section describes how to manually install the glass box agent on a Tomcat server.

#### **About this task**

When AppScan is installed, the files needed for installing the server agent are saved in a dedicated folder on your machine. To perform this task you will need access to this folder, and to your application server.

#### **Procedure**

1. Open ...\\Program Files\\IBM\\AppScan Standard\\Glass box  
 The absolute path will depend on where you installed AppScan.
2. Locate **GB\_Java\_Manual\_Setup.zip** and copy it to your web server.
3. Extract the contents of this folder to your preferred location on the web server.
4. Define a username and password for the agent (use only English ASCII characters):
  - **Linux server:** Run `AgentCredentials.sh <username> <password>`

**Note:** `AgentCredentials.sh` must have execution permissions.

- **Windows server:** Run `AgentCredentials.bat <username> <password>`

5. Deploy **GBootStrap** web application.

- Preferred method:
    - Log in to Tomcat Manager. The default location is `http://<server_name>:<port_number>/manager/html`
    - In **Deploy table > War file to deploy**, click **Choose file**
    - Locate GBootStrap.war (in the glass box folder you extracted), and click **Open**
    - Click **Deploy** and verify that GBootStrap has been added the applications list
  - Alternative method:
    - Copy the file GBootStrap.war from the glass box folder you extracted
    - Paste it into the \webapps folder, default location:  
`C:\apache-tomcat-[version]\webapps`
6. Shut down Tomcat.
7. Configure Tomcat to always use the glass box agent by doing *one* of the following:
- **Through the environmental variable:** Configure Tomcat's JVM through the JAVA\_OPTS environmental variables (if it does not exist, create it), with the value  
`-javaagent:<path_to_gbAgent.jar>/gbAgent.jar`
  - **Through batch/sh scripts:** Pass arguments to the JVM running Tomcat by editing the configuration scripts that are shipped with it:
 

**Linux server:**

    - In the Tomcat folder (usually located in `<path_to_Tomcat_folder>/bin`), open `startup.sh`
    - Locate a line that begins with CATALINA\_OPTS and add the following line below it:  
`export CATALINA_OPTS = $CATALINA_OPTS -javaagent:<path_to_gbAgent.jar>/gbAgent.jar`
    - Save and close the file.

**Windows server:**

    - In the Tomcat folder (usually located in `<path_to_Tomcat_folder>\bin`), open `startup.bat`
    - Locate a line that begins with `set CATALINA_OPTS=` and add the following line below it:  
`set CATALINA_OPTS = %CATALINA_OPTS% -javaagent:<path_to_gbAgent.jar>/gbAgent.jar`
    - Save and close the file.

**Important:** Ensure that CATALINA\_OPTS is initialized only once, and prior to the lines above, so that other CATALINA\_OPTS assignments do not overwrite -javaagent.

**Note:** To add arguments to CATALINA\_OPTS, use the `%CATALINA_OPTS%` / `$ CATALINA_OPTS` convention.
8. Restart Tomcat.

#### Install manually on a Tomcat Service server:

This section describes how to manually install the glass box agent on a Tomcat Service/Daemon server.

#### About this task

When AppScan is installed, the files needed for installing the server agent are saved in a dedicated folder on your machine. To perform this task you will need access to this folder, and to your application server.

#### Procedure

1. Open ...\\Program Files\\IBM\\AppScan Standard\\Glass box  
 The absolute path will depend on where you installed AppScan.
2. Locate GB\_Java\_Manual\_Setup.zip and copy it to your web server.
3. Extract the contents of this folder to your preferred location on the web server.

4. Define a username and password for the agent (use only English ASCII characters):
  - **Linux server:** Run `AgentCredentials.sh <username> <password>`

**Note:** `AgentCredentials.sh` must have execution permissions.
  - **Windows server:** Run `AgentCredentials.bat <username> <password>`
5. Deploy GBootStrap web application:
  - a. Log in to Tomcat Manager. The default location is `http://<server_name>:<port_number>/manager/html`
  - b. In **Deploy table > War file to deploy**, click **Choose file**.
  - c. Locate `GBootStrap.war` (in the glass box folder you extracted), and click **Open**.
  - d. Click **Deploy** and verify that `GBootStrap` has been added the applications list.
6. Shut down Tomcat.
7. Configure Tomcat to use the glass box agent:
  - a. Locate Tomcat JVM at `...\\Tomcat 7.0\\bin\\tomcat7w.exe`
  - b. Double-click on it and select **Properties > Java tab**.
  - c. In the Java Options area, add the following as a new line: `-javaagent:<path_to_gbAgent.jar>/gbAgent.jar`

**Note:** If there are other Java agents defined on the web server, you can *add* the glass box agent to the Java Options area (the exact path will depend on your installation):  
`... -javaagent:c:\\...\\otherAgent.jar -javaagent:c:\\glassbox\\gbAgent.jar ...`
8. Click **OK**.

## 8. Restart Tomcat.

### Install manually on a WebLogic server:

This section describes how to manually install the glass box agent on a WebLogic server.

#### About this task

When AppScan is installed, the files needed for installing the server agent are saved in a dedicated folder on your machine. To perform this task you will need access to this folder, and to your application server.

#### Procedure

1. Open `...\\Program Files\\IBM\\AppScan Standard\\Glass box`  
 The absolute path will depend on where you installed AppScan.
2. Locate `GB_Java_Manual_Setup.zip` and copy it to your web server.
3. Extract the contents of this folder to your preferred location on the web server.
4. Define a username and password for the agent (use only English ASCII characters):
  - **Linux server:** Run `AgentCredentials.sh <username> <password>`

**Note:** `AgentCredentials.sh` must have execution permissions.
  - **Windows server:** Run `AgentCredentials.bat <username> <password>`
5. Deploy GBootStrap web application:
  - a. Log in to the WebLogic Administration Console. The default location is `http://<server_name>:<port_number>/console`
  - b. In the **Domain Structure** pane, click **Deployments**, then click **Install**
  - c. In the **Path** field, type in the path to `GBootStrap.war`, then click **Next**
  - d. Select the **Install this deployment as an application** radio button, then click **Next**

- e. Verify that the text in the **Name** field is **GBootStrap**
  - f. Click **Finish**, then click **Save**
  - g. In the **Domain Structure** pane, click **Deployments**, and verify that **GBootStrap** has been added, and that it has a green checkmark icon in the **Health** column.
6. Shut down the WebLogic server.
7. Supply the Java agent gbAgent.jar (found in the folder you extracted) as a JVM argument by doing *one* of the following:
- **Through the environmental variable:** Configure WebLogic's JVM through the JAVA\_OPTS environmental variable (if it does not exist, create it), with the value:  
-javaagent:<path\_to\_gbAgent.jar>/gbAgent.jar.
  - **Through batch/sh scripts:** Pass arguments to the JVM running WebLogic by editing the configuration scripts that are shipped with the server:
- Linux server:**
- a. In the WebLogic folder (usually located in <path\_to\_Weblogic\_folder>\bin), open startWebLogic.sh
  - b. Locate a line that begins with JAVA\_OPTIONS and add the following line below it:  
export JAVA\_OPTIONS = \$JAVA\_OPTIONS -javaagent:<path\_to\_gbAgent.jar>/gbAgent.jar
  - c. Save and close the file.
- Windows server:**
- a. In the WebLogic folder (usually located in <path\_to\_WebLogic\_folder>\bin), open startWebLogic.bat
  - b. Locate a line that begins with set JAVA\_OPTIONS and add the following line below it:  
set JAVA\_OPTIONS = %JAVA\_OPTIONS% -javaagent:<path\_to\_gbAgent.jar>/gbAgent.jar
  - c. Save and close the file.

**Important:** Ensure that JAVA\_OPTIONS is initialized only once, and prior to the lines above, so that other JAVA\_OPTIONS assignments do not overwrite -javaagent.

**Note:** To add arguments to JAVA\_OPTIONS, use the %JAVA\_OPTIONS% / \$ JAVA\_OPTIONS convention.

8. Restart the WebLogic server.

#### Install manually on a WebLogic Service server:

This section describes how to manually install the glass box agent on a WebLogic Service/Daemon server.

#### About this task

When AppScan is installed, the files needed for installing the server agent are saved in a dedicated folder on your machine. To perform this task you will need access to this folder, and to your application server.

#### Procedure

1. Open ...\\Program Files\\IBM\\AppScan Standard\\Glass box  
The absolute path will depend on where you installed AppScan.
2. Locate GB\_Java\_Manual\_Setup.zip and copy it to your web server.
3. Extract the contents of this folder to your preferred location on the web server.
4. Define a username and password for the agent (only English characters and numbers can be used):
  - **Linux server:** Run AgentCredentials.sh <username> <password>

**Note:** AgentCredentials.sh must have execution permissions.

- **Windows server:** Run AgentCredentials.bat <username> <password>
5. Deploy GBootStrap web application:
    - a. Log in to the WebLogic Administration Console. The default location is `http://<server_name>:<port_number>/console`
    - b. In the **Domain Structure** pane, click **Deployments**, then click **Install**
    - c. In the **Path** field, type in the path to GBootStrap.war, then click **Next**
    - d. Select the **Install this deployment as an application** radio button, then click **Next**

**Note:** If the Install option is disabled (grayed out), you may need to change the Lock & Edit mode setting to enable it.

    - e. Verify that the text in the **Name** field is **GBootStrap**
    - f. Click **Finish**, then click **Save**
    - g. In the **Domain Structure** pane, click **Deployments**, and verify that **GBootStrap** has been added, and that it has a green checkmark icon in the **Health** column.
  6. Shut down the WebLogic service.
  7. Configure WebLogic to use the glass box agent by locating the registry key `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\<WEBLOGIC_SERVICE>` and editing the value of `CmdLine` to include `-javaagent:<path_to_gbAgent.jar>/gbAgent.jar`
  8. Restart the service.

#### Install manually on a WebSphere server:

This section describes how to manually install the glass box agent on a WebSphere server.

#### About this task

When AppScan is installed, the files needed for installing the server agent are saved in a dedicated folder on your machine. To perform this task you will need access to this folder, and to your application server.

#### Procedure

1. Open ...\\Program Files\\IBM\\AppScan Standard\\Glass box  
The absolute path will depend on where you installed AppScan.
2. Locate `GB_Java_Manual_Setup.zip` and copy it to your web server.
3. Extract the contents of this folder to your preferred location on the web server.
4. Define a username and password for the agent (use only English ASCII characters):
  - **Linux server:** Run `AgentCredentials.sh <username> <password>`  
**Note:** `AgentCredentials.sh` must have execution permissions.
  - **Windows server:** Run `AgentCredentials.bat <username> <password>`
5. Deploy GBootStrap web application:
  - a. Log in to the WebSphere's Integrated Solutions console. The default location is `http://<server_name>:<port_number>/ibm/console`  
**Note:** If more than one profile has been defined, log into the appropriate one.
  - b. Select **New Applications > New Enterprise Application**
  - c. In the **Path to the new application** area, add the full path to the GBootStrap.war file.
  - d. In the **Context root** field, type in `GBootStrap`, then click **Next**.  
The Install New Application wizard opens.
  - e. Unless you want to change any default options, simply click **Next** until the wizard completes. If you are asked to save the configuration, do so. When you have completed the wizard,

GBootstrap.war will appear in the list of applications. A red X or a green arrow icon in the Application Status column indicate whether or not it is active. If it is *not* active (red X), select it and click **Start**. A green arrow indicates that GBootstrap.war is active.

6. Now select **Servers > Application Servers**
7. In the right pane, click on your server's name.
8. Click on the **Configuration** tab to bring it to the front.
9. Under Server Infrastructure, click **Java and Process Management > Process Definition**.
10. Under Additional Properties click **Java Virtual Machine**.
11. Add the following argument to the Generic JVM Arguments:  
-javaagent:c:/path/to/gbAgent.jar (the example is for Windows operating systems; for other systems the path should be adjusted appropriately).

**Note:** The path should not contain any spaces.

**Note:** If there are other Java agents defined on the web server, you can *add* the glass box agent to the Generic JVM Arguments area (the exact path will depend on your installation):

... -javaagent:c:\otherAgent\otherAgent.jar -javaagent:c:\glassbox\gbAgent.jar ...

12. Click **Apply**, and save the configuration if asked.
13. Restart WebSphere server.

## Permissions needed to work in secure mode

This section describes the special permissions you need to add if the web application server will be running in secure mode during the scan.

If the server will be running in secure mode (that is, with Java Security Manager enabled), you must add the following special permissions to the GBootstrap web application:

- Permission to access "getClassLoader" (java.lang.RuntimePermission)
- Permission to use "accessClassInPackage.sun.net.www.protocol.\*" (java.lang.RuntimePermission)
- Read permission to the "java.io.tmpdir" property (java.util.PropertyPermission)
- Read/Write/Delete permission to "<<ALL FILES>>"

If necessary, consult your web server manual for instructions on how and where to add these Java Security permissions. The sections below provide some sources and examples that may help you in doing this. It should be stressed that the examples are examples only, and would almost certainly need adaptation to work in your particular case.

## IBM WebSphere

Guide:

[http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/topic/com.ibm.websphere.express.doc/info/exp/ae/tsec\\_waspolicyfile.html](http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/topic/com.ibm.websphere.express.doc/info/exp/ae/tsec_waspolicyfile.html)

Outline of lines to be added:

```
grant codeBase "file:${application}" {  
    permission java.lang.RuntimePermission "getClassLoader";  
    permission java.lang.RuntimePermission  
"accessClassInPackage.sun.net.www.protocol.*";  
    permission java.io.FilePermission "<>", "read, write, delete";  
    permission java.util.PropertyPermission "java.io.tmpdir", "read";  
};
```

Location:

```
<profile_root>/config/cells/<cell_name>/applications/<ear_file_name>
/deployments/<application_name>/META-INF/was.policy
```

Example of lines to be added:

```
"C:\Program Files (x86)\IBM\WebSphere\AppServer\profiles\AppSrv01
\config\cells\ROIS-PSNode01Cell\applications\GBootStrap_war.ear
\deployments\GBootStrap_war\META-INF\was.policy")
```

## Apache Tomcat

Guide:

[http://tomcat.apache.org/tomcat-6.0-doc/security-manager-howto.html#Configuring\\_Tomcat\\_With\\_A\\_SecurityManager](http://tomcat.apache.org/tomcat-6.0-doc/security-manager-howto.html#Configuring_Tomcat_With_A_SecurityManager)

Outline of lines to be added:

```
grant codeBase "file:${catalina.base}/webapps/GBootStrap/-"{
    permission java.lang.RuntimePermission "getClassLoader";
    permission java.lang.RuntimePermission
    "accessClassInPackage.sun.net.www.protocol.*";
    permission java.io.FilePermission "<>FILE<>>"", "read, write, delete";
    permission java.util.PropertyPermission "java.io.tmpdir", "read";
};
```

Location:

```
<CATALINA_HOME>/conf/catalina.policy
```

Example of lines to be added:

```
"C:\Software\Tomcat\apache-tomcat-6.0.33\conf\catalina.policy"
```

## Weblogic

Guide:

[http://download.oracle.com/docs/cd/E13222\\_01/wls/docs81/security/server\\_prot.html](http://download.oracle.com/docs/cd/E13222_01/wls/docs81/security/server_prot.html)

Specifically refer to the section entitled "Setting Application-Specific Security Policies"

Outline of lines to be added:

```
<security-permission>
  <description>
    Allow getting the J2EEJ2SETest4 property
  </description>
  <security-permission-spec>
    grant {
      permission java.util.PropertyPermission
      "welcome.J2EEJ2SETest4", "read";
    };
  </security-permission-spec>
</security-permission>
```

Line to be added:

```
weblogic.xml
```

## JBoss

Guide:

<http://docs.jboss.org/jbossweb/latest/security-manager-howto.html>

Outline of lines to be added:

```
grant codeBase "file:${catalina.base}/webapps/GBootstrap/-"{
    permission java.lang.RuntimePermission "getClassLoader";
    permission java.lang.RuntimePermission
"accessClassInPackage.sun.net.www.protocol.*";
    permission java.io.FilePermission "ALL FILES", "read, write, delete";
    permission java.util.PropertyPermission "java.io.tmpdir", "read";
};
```

Line to be added:

CATALINA\_HOME/conf/catalina.policy

## Defining the glass box agent in AppScan

This section describes how to configure AppScan to access a server-side glass box agent.

### About this task

Once you have installed the glass box agent on your application server, you must define it to AppScan so it can access the agent. After this is done AppScan will automatically use the agent for glass box scanning in any relevant scan (unless you deselect it in the scan configuration).

**Restriction:** You can define the same server agent on several AppScan machines, however an agent can be used for only one scan at a time.

**Restriction:** You can define multiple agents, but only one agent can be selected in a scan configuration.

### Procedure

1. In Configuration > URL and Servers tab, set the Starting URL. Note that in this example, the URL includes port 8080.

<https://altoro.testfire.net:8080/>

2. Click OK to close the Configuration dialog box.

3. Click Tools > Glass box agent management.

The Glass Box Agents dialog box appears, listing any agents that have already been defined.

4. To add a new agent to the list, click .

The Glass Box Agents Definition dialog box appears. The Glass Box Agent URL is automatically filled, based on the Starting URL you entered previously.

**Important:** The Glass Box Agent URL automatically contains a port value extracted from the Starting URL, (in this case 8080). If you installed your agent on a different port, replace the port value with the correct one. In this example the port value has been changed to 8888.

<http://altoro.testfire.net:8888/GBootstrap/>

5. Fill in the remaining fields and options as needed.

Option	Description
Username and password	Type in the username and password that were defined when the agent was installed on the server.
Agent log settings:	(Optional:) Click to open these settings. <b>Note:</b> Agent log settings are saved in the agent on the server-side.
Max. log rows	(Optional:) Use the slider to limit the size of the log.

Option	Description
<b>Log content</b>	(Optional:) Select the level of information to be include in the log: <b>Errors:</b> Include only error messages; <b>Warnings:</b> Include error and warning messages; <b>Info:</b> Include errors, warnings and informational messages; <b>Debug:</b> Include all messages.

AppScan tries to connect to the agent server. If successful, a green "Connected" icon appears. If there is any problem, a red icon will appear with a message such as "Unable to connect to agent" or "Credentials needed".



**Note:** If your site offers both HTTP and HTTPS, verify that the starting URL is HTTPS (and change it to HTTPS if it is not).

**Note:** If you get an error message, refer to

<http://www.ibm.com/support/docview.wss?uid=swg21567723>

6. Click **OK** to close the definition dialog box.  
The agent is added to the list.
7. Click **OK** to close the list.  
The agent is now defined on this machine.

#### Change user credentials:

This section describes how to change the username and password used to access the glass box agent.

#### About this task

When the glass box agent is installed, you are asked to define a username and password for accessing the agent. These credentials help insure that the agent is accessible only to AppScan. If you need to change the credentials after installation, follow these instructions.

#### Procedure

1. On your application server, open a command line to your installation directory.
2. Define the new credentials:
  - For a Linux server: Run `AgentCredentials.sh <newusername> <newpassword>`
  - For a Windows server: Run `AgentCredentials.bat <newusername> <newpassword>`
3. Undeploy GBootstrap from the server.
4. Deploy the new modified GBootstrap.war from the glass box installation folder to the server.

#### Configuring a glass box scan

Although the scan is configured automatically, this section describes options you may want to change.

#### Procedure

1. Configure your scan as usual.

2. In Configuration > Glass Box tab, select one of the defined glass box agents from the drop-down list, and optionally adjust the settings:

Setting	Details
Use this glass box agent	If the glass box agent has been installed on your application server, and defined in AppScan, you can select it for use in the scan. If you have entered a Starting URL, AppScan attempts to select the appropriate agent automatically.  When an agent is selected, AppScan attempts to connect to it, and indicates whether this was successful. <b>Note:</b> If you select an agent and get the message "Credentials needed", check that the credentials supplied in Tools > Glass Box Management are correct.  If the required server does not appear in the drop-down list, you can define it by clicking the <b>Glass box agent management</b> link. <b>Restriction:</b> Only one glass box agent can be selected for use in a scan. If the application being scanned has more than one server, you must scan using each server agent separately.
Use glass box in the Explore stage	(Selected by default.)  This function can increase coverage of the site, by examining the server-side source code for the presence of parameters that affect the behavior of the server, but do not appear in the response.  Example server-side code: <pre>String debugOn = request.getParameter("debug"); if (debugOn == "true"){     response.getWriter().println(SECRET_SERVER_DATA); }</pre> In this example the developer has left the parameter "debug" in the code. It does not appear in any link on the site, but if an attacker were to send a request containing it, SECRET_SERVER_DATA could be obtained.
Use glass box in the Test stage	(Selected by default.) Select this check box to send glass box tests during the Test stage of the scan. This function can verify the success or failure of certain tests, such as Blind SQL Injection, with greater accuracy, and also reveal the existence of certain security issues that cannot be detected by black box techniques.
Skip equivalent black box tests	(Cleared by default.) This means that both glass box tests and black box tests for the same vulnerability (WASC Threat Classification) are sent. This is because although the glass box tests are generally both more accurate and give more detailed results, occasionally a glass box test may fail while the equivalent black box test succeeds. If the results for your application are unchanged when black box tests are skipped, you can reduce scan time by selecting this check box.

The status bar indicates that glass box scanning is enabled, and you are ready to start the scan.



## Scanning with glass box

This section describes glass box scanning.

### About this task

Once you have defined the glass box agent to AppScan, glass box scanning is enabled by default. You can use the Scan Configuration dialog box to verify that the correct server agent is selected, and glass box scanning is configured to run as part of the scan.

Glass box scanning can discover hidden URLs in the Explore stage, and additional issues and information during the Test stage.

## Procedure

1. Click Configuration > Glass Box view.
2. Select the agent you want to use from the drop-down list.

**Note:** If your agent does not appear in the list, click the **Glass box agent management** link, and define it.

3. Verify that one or both of the two main glass box scan options are selected:
  - Use glass box in the Explore stage
  - Use glass box in the Test stage

**Note:** The **Skip equivalent black box tests** check box is cleared by default. This means that both glass box tests and black box tests for the same vulnerability (WASC Threat Classification) are sent. This is because although the glass box tests are generally both more accurate and give more detailed results, occasionally a glass box test may fail while the equivalent black box test succeeds. If the results for your application are unchanged when black box tests are skipped, you can reduce scan time by selecting this check box.

4. Click **Scan > Full Scan** to start the scan.

The scan starts and the status bar messages indicate that glass box scanning is active.



The scan results will include glass box data in the Issue Information tab where available.

## Uninstalling the glass box agent

This section describes how to uninstall the server-side glass box agent.

- “Uninstall automatically”
- “Uninstall manually on a JBoss server” on page 151
- “Uninstall manually on a JBoss Service server” on page 151
- “Uninstall manually on a Tomcat server” on page 152
- “Uninstall manually on a Tomcat Service server” on page 152
- “Uninstall manually on a WebLogic server” on page 153
- “Uninstall manually on a WebLogic Service server” on page 153
- “Uninstall manually on a WebSphere server” on page 153

### Uninstall automatically:

This section describes how to uninstall the glass box agent automatically through the user interface.

#### About this task

The glass box agent can be uninstalled automatically *only* if it was installed automatically using the glass box agent installer. If it was installed manually, it must be uninstalled manually too, following the instructions in the next sections of this guide.

To perform this task you will need access to your application server.

## Procedure

1. Open the glass box directory on your server.
2. Launch the uninstaller:
  - **Linux server:** Double-click on `Uninstall.bin`
  - **Windows server:** Double-click on `Uninstall.exe`

3. Follow the online instructions for completely uninstalling glass box.

#### **Uninstall manually on a JBoss server:**

This section describes how to manually uninstall the glass box agent on a JBoss server.

#### **About this task**

When glass box is installed on the application server, the installation includes a Java agent and a GBootStrap web application. To uninstall glass box both of these must be removed. This task requires access to your application server.

#### **Procedure**

1. Undeploy GBootStrap web application:
  - a. Log in to the JBoss Administration Console. The default location is `http://<server_name>:<port_number>/admin-console/`
  - b. Click **Applications > Web Application WARs**, then, in the GBootStrap.war entry, click **Delete**.
  - c. When asked to confirm, click **OK**.
2. Shut down the JBoss server.
3. Remove the Java agent from JAVA\_OPTS. During manual installation, JAVA\_OPTS is edited to include `gbAgent.jar`. Refer to the installation instructions to remove `gbAgent.jar` from JAVA\_OPTS, and restore JAVA\_OPTS to its original state.
4. Delete the glass box directory from the JBoss server, unless it is still to be used by other web application servers.
5. Restart the JBoss server.

#### **Uninstall manually on a JBoss Service server:**

This section describes how to manually uninstall the glass box agent on a JBoss Service server.

#### **About this task**

When glass box is installed on the application server, the installation includes a Java agent and a GBootStrap web application. To uninstall glass box both of these must be removed. This task requires access to your application server.

#### **Procedure**

1. Undeploy GBootStrap web application:
  - a. Log in to the JBoss Administration Console. The default location is `http://<server_name>:<port_number>/admin-console/`
  - b. Click **Applications > Web Application WARs**, then, in the GBootStrap.war entry, click **Delete**.
  - c. When asked to confirm, click **OK**.
2. Shut down the JBoss service.
3. Remove the Java agent from JAVA\_OPTS. During manual installation, JAVA\_OPTS is edited to include `gbAgent.jar`. Refer to the installation instructions to remove `gbAgent.jar` from JAVA\_OPTS, and restore JAVA\_OPTS to its original state.
4. Delete the glass box directory from the JBoss server, unless it is still to be used by other web application servers.
5. Restart the service.

## **Uninstall manually on a Tomcat server:**

This section describes how to manually uninstall the glass box agent on a Tomcat server.

### **About this task**

When glass box is installed on the application server, the installation includes a Java agent and a GBootStrap web application. To uninstall glass box both of these must be removed. This task requires access to your application server.

### **Procedure**

1. Undeploy GBootStrap web application:
  - a. Log in to Tomcat Manager. The default location is `http://<server_name>:<port_number>/manager/html`
  - b. In the application list, select **GBootStrap.war**, then click **Commands > Undeploy**.
  - c. When asked to confirm, click **OK**.
2. Shut down Tomcat server.
3. Remove the Java agent from CATALINA\_OPTS. During manual installation, CATALINA\_OPTS is edited to include gbAgent.jar. Refer to the installation instructions to remove gbAgent.jar from CATALINA\_OPTS, restoring CATALINA\_OPTS to its original state.
4. Delete the glass box directory from the Tomcat server, unless it is still to be used by other web application servers.
5. Restart the Tomcat server.

## **Uninstall manually on a Tomcat Service server:**

This section describes how to manually uninstall the glass box agent on a Tomcat Service server.

### **About this task**

When glass box is installed on the application server, the installation includes a Java agent and a GBootStrap web application. To uninstall glass box both of these must be removed. This task requires access to your application server.

### **Procedure**

1. Undeploy GBootStrap web application:
  - a. Log in to Tomcat Manager. The default location is `http://<server_name>:<port_number>/manager/html`
  - b. In the application list, select **GBootStrap.war**, then click **Commands > Undeploy**.
  - c. When asked to confirm, click **OK**.
2. Shut down Tomcat Service.
3. Remove the Java agent from JVM arguments:
  - a. Locate Tomcat JVM at ...\\Tomcat 7.0\\bin\\tomcat7w.exe
  - b. Double-click on it, and select **Properties > Java** tab.
  - c. In the Java Options area, remove the following line (if it exists): `-javaagent:c:/path/to/gbAgent.jar`
  - d. Click **OK**.
4. Restart Tomcat Service.

## **Uninstall manually on a WebLogic server:**

This section describes how to manually uninstall the glass box agent on a WebLogic server.

### **About this task**

When glass box is installed on the application server, the installation includes a Java agent and a GBootStrap web application. To uninstall glass box both of these must be removed. This task requires access to your application server.

### **Procedure**

1. Undeploy GBootStrap web application:
  - a. Log in to WebLogic Administration Console. The default location is `http://<server_name>:<port_number>/console/`
  - b. In the Domain Structure pane, click **Deployments**, then select the **GBootStrap** checkbox, and click **Delete**.
  - c. In the Domain Structure pane, click **Deployments**, and verify that GBootStrap has been removed.
2. Shut down WebLogic server.
3. Remove the Java agent from JAVA\_OPTIONS: During manual installation, JAVA\_OPTIONS is edited to include gbAgent.jar. Refer to the installation instructions to remove gbAgent.jar from JAVA\_OPTIONS, restoring JAVA\_OPTIONS to its original state.
4. Delete the glass box directory from the WebLogic server, unless it is still to be used by other web application servers.
5. Restart the WebLogic server.

## **Uninstall manually on a WebLogic Service server:**

This section describes how to manually uninstall the glass box agent on a WebLogic Service server.

### **About this task**

When glass box is installed on the application server, the installation includes a Java agent and a GBootStrap web application. To uninstall glass box both of these must be removed. This task requires access to your application server.

### **Procedure**

1. Undeploy GBootStrap web application:
  - a. Log in to WebLogic Administration Console. The default location is `http://<server_name>:<port_number>/console/`
  - b. In the Domain Structure pane, click **Deployments**, then select the **GBootStrap** checkbox, and click **Delete**.
  - c. In the Domain Structure pane, click **Deployments**, and verify that GBootStrap has been removed.
2. Shut down WebLogic Service.
3. Remove the Java agent from JAVA\_OPTIONS: During manual installation, JAVA\_OPTIONS is edited to include gbAgent.jar. Refer to the installation instructions to remove gbAgent.jar from JAVA\_OPTIONS, restoring JAVA\_OPTIONS to its original state.
4. Restart the service.

## **Uninstall manually on a WebSphere server:**

This section describes how to manually uninstall the glass box agent on a WebSphere Service server.

## About this task

When glass box is installed on the application server, the installation includes a Java agent and a GBootStrap web application. To uninstall glass box both of these must be removed. This task requires access to your application server.

## Procedure

### 1. Undeploy GBootStrap web application:

- a. Log in to WebSphere's Integrated Solutions console. The default location is `http://<server_name>:9043/console/`

**Note:** If more than one profile is defined, log in to the console of the appropriate one.

- b. Select **Applications > Application Types > WebSphere Enterprise Application**.
- c. In the Enterprise Applications area, select the **GBootStrap** box, and click **Uninstall**.
- d. Follow the wizard's instructions to remove the GBootStrap application.  
GBootStrap is removed from the list of applications.
- e. Select **Servers > Application Servers**.
- f. In the right pane, click on your server's name.
- g. Click on the **Configuration** tab to bring it to the front.
- h. In Server Infrastructure, click **Java and Process Management > Process Definition**.
- i. In Additional Properties, click **Java Virtual Machine**.

### 2. In Generic JVM Arguments, remove the glass box agent by clearing `-javaagent:c:/path/to/gbAgent.jar`

**Note:** This example applies to Windows operating systems; for other systems adjust the path as appropriate.

3. Click **Apply**, and save the configuration if asked.
4. Restart WebSphere.

## For .NET platforms

Installing and using the glass box agent on a .NET server.

### Installing the glass box agent

This section describes how to install the glass box agent on a .NET server.

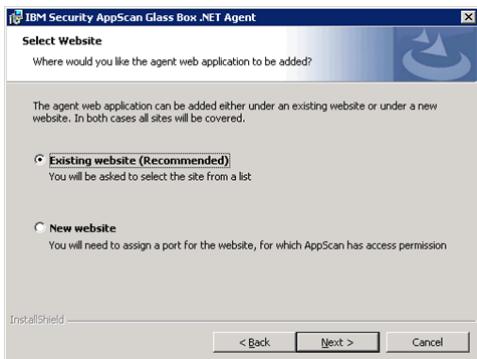
### Before you begin

The AppScan glass box agent must be installed on your application server (or servers), by copying certain files from the AppScan installation to the server. To do this you will need access to your application server. The following systems and technologies are supported:

Item	Details
Operating System	Supported operating systems (both 32-bit and 64-bit editions): <ul style="list-style-type: none"><li>• Microsoft Windows Server 2012</li><li>• Microsoft Windows Server 2012 R2</li><li>• Microsoft Windows Server 2008 SP2</li><li>• Microsoft Windows Server 2008 R2</li></ul>
Other	Microsoft IIS 7.0 or later  Microsoft .NET Framework 4.0 or 4.5 must be installed, and IIS must be configured at the root level to work with this version of ASP.net

## Procedure

1. Open ...\\Program Files\\IBM\\AppScan Standard\\Glass box  
The absolute path will depend on where you installed AppScan.
2. Locate GB\_DotNET\_Setup.exe and copy it to your web server machine.
3. Double-click on it to start the wizard.
4. Follow the wizard steps.



5. When you are asked whether to install on a new or existing website, select the recommended option (unless you are sure the alternative is better).

**Note:** Which of the two options is recommended depends on your system. Where possible an "existing website" will be recommended, but this is not always possible, and "new website" may be recommended in some cases. In general "existing website" is preferred because it does not require a new port, which would then need to be configured in AppScan, and might also require additional firewall configuration to be accessible to AppScan.

6. This step depends on your choice in the previous step:
  - a. **Existing site:** If you chose "Existing site", you are now asked to select a site from a list of existing sites.

**Note:** When there is more than one site, the wizard will list them in order of preference. If the wizard identifies some sites as potentially problematic, it will list them lower down, and label them "not recommended". However, the order, and even the "not recommended" label are only suggestions, and in some cases you may decide to select a site that is labeled "not recommended".

**Tip:** Whichever selection you make will enable monitoring of all sites, but where possible try to select the site which you plan to configure as the starting URL of your *first* scan.

6. This step depends on your choice in the previous step:
  - b. **New site:** If you chose "New site" you are now asked to assign a port. You must define a port that will allow access to AppScan when scanning from a remote machine.

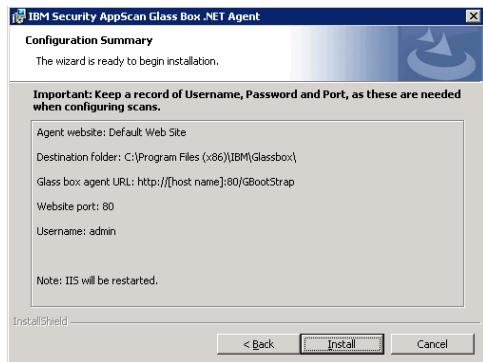
**Important:** You will need to supply the port number when configuring AppScan.

**Note:** You can use the **Test** button to verify that the selected port is currently available. However, this will automatically be done when you click **Next**.

7. Set the Agent Username and Password for the glass box web application agent.

**Important:** You will need to supply these credentials when configuring a scan with glass box.

8. The summary screen appears before the actual installation takes place.



**Important:** Before clicking **Install** make sure you have a record of the Username, Password and Port, as you will need these when configuring AppScan.

#### 9. Click **Install**.

**Note:** The installation includes a restart of your IIS.

When the process completes a final message appears indicating its success or failure.

**Note:** If the installation fails for any reason, the final dialog will provide a link to the installation log for use in troubleshooting. It may be worth attempting to install on a different or new site on the server, even a different site that was labeled "not recommended". In all cases you must uninstall before attempting installation again. The location of the two installation logs is the %temp% folder. The names of the two installation logs are:

glassbox\_setup\_api.txt  
glassbox\_setup\_msi.txt

Note that the uninstall process includes a restart of your IIS.

### Defining the glass box agent in AppScan

This section describes how to configure AppScan to access a server-side glass box agent.

#### About this task

Once you have installed the glass box agent on your application server, you must define it to AppScan so it can access the agent. After this is done AppScan will automatically use the agent for glass box scanning in any relevant scan (unless you deselect it in the scan configuration).

**Restriction:** You can define the same server agent on several AppScan machines, however an agent can be used for only one scan at a time.

**Restriction:** You can define multiple agents, but only one agent can be selected in a scan configuration.

#### Procedure

- In **Configuration > URL and Servers tab**, set the **Starting URL**. Note that in this example, the URL includes port 8080.  
`https://altoro.testfire.net:8080/`
- Click **OK** to close the Configuration dialog box.
- Click **Tools > Glass box agent management**.  
The Glass Box Agents dialog box appears, listing any agents that have already been defined.
- To add a new agent to the list, click .

The Glass Box Agents Definition dialog box appears. The Glass Box Agent URL is automatically filled, based on the Starting URL you entered previously.

**Important:** The Glass Box Agent URL automatically contains a port value extracted from the Starting URL, (in this case 8080). If you installed your agent on a different port, replace the port value with the correct one. In this example the port value has been changed to 8888.

<http://altoro.testfire.net:8888/GBootStrap/>

5. Fill in the remaining fields and options as needed.

Option	Description
<b>Username and password</b>	Type in the username and password that were defined when the agent was installed on the server.
<b>Agent log settings:</b>	(Optional:) Click to open these settings. <b>Note:</b> Agent log settings are saved in the agent on the server-side.
<b>Max. log rows</b>	(Optional:) Use the slider to limit the size of the log.
<b>Log content</b>	(Optional:) Select the level of information to be include in the log: <b>Errors:</b> Include only error messages; <b>Warnings:</b> Include error and warning messages; <b>Info:</b> Include errors, warnings and informational messages; <b>Debug:</b> Include all messages.

AppScan tries to connect to the agent server. If successful, a green "Connected" icon appears. If there is any problem, a red icon will appear with a message such as "Unable to connect to agent" or "Credentials needed".



**Note:** If your site offers both HTTP and HTTPS, verify that the starting URL is HTTPS (and change it to HTTPS if it is not).

**Note:** If you get an error message, refer to

<http://www.ibm.com/support/docview.wss?uid=swg21567723>

6. Click **OK** to close the definition dialog box.  
The agent is added to the list.
7. Click **OK** to close the list.  
The agent is now defined on this machine.

## Configuring a glass box scan

Although the scan is configured automatically, this section describes options you may want to change.

### Procedure

1. Configure your scan as usual.
2. In **Configuration > Glass Box tab**, select one of the defined glass box agents from the drop-down list, and optionally adjust the settings:

Setting	Details
Use this glass box agent	<p>If the glass box agent has been installed on your application server, and defined in AppScan, you can select it for use in the scan. If you have entered a Starting URL, AppScan attempts to select the appropriate agent automatically.</p> <p>When an agent is selected, AppScan attempts to connect to it, and indicates whether this was successful.</p> <p><b>Note:</b> If you select an agent and get the message "Credentials needed", check that the credentials supplied in Tools &gt; Glass Box Management are correct.</p> <p>If the required server does not appear in the drop-down list, you can define it by clicking the <b>Glass box agent management</b> link.</p> <p><b>Restriction:</b> Only one glass box agent can be selected for use in a scan. If the application being scanned has more than one server, you must scan using each server agent separately.</p>
Use glass box in the Explore stage	<p>(Selected by default.)</p> <p>This function can increase coverage of the site, by examining the server-side source code for the presence of parameters that affect the behavior of the server, but do not appear in the response.</p> <p>Example server-side code:</p> <pre>String debugOn = request.getParameter("debug"); if (debugOn == "true"){     response.getWriter().println(SECRET_SERVER_DATA); }</pre> <p>In this example the developer has left the parameter "debug" in the code. It does not appear in any link on the site, but if an attacker were to send a request containing it, SECRET_SERVER_DATA could be obtained.</p>
Use glass box in the Test stage	(Selected by default.) Select this check box to send glass box tests during the Test stage of the scan. This function can verify the success or failure of certain tests, such as Blind SQL Injection, with greater accuracy, and also reveal the existence of certain security issues that cannot be detected by black box techniques.
Skip equivalent black box tests	(Cleared by default.) This means that both glass box tests and black box tests for the same vulnerability (WASC Threat Classification) are sent. This is because although the glass box tests are generally both more accurate and give more detailed results, occasionally a glass box test may fail while the equivalent black box test succeeds. If the results for your application are unchanged when black box tests are skipped, you can reduce scan time by selecting this check box.

The status bar indicates that glass box scanning is enabled, and you are ready to start the scan.



## Scanning with glass box

This section describes glass box scanning.

### About this task

Once you have defined the glass box agent to AppScan, glass box scanning is enabled by default. You can use the Scan Configuration dialog box to verify that the correct server agent is selected, and glass box scanning is configured to run as part of the scan.

Glass box scanning can discover hidden URLs in the Explore stage, and additional issues and information during the Test stage.

## Procedure

1. Click Configuration > Glass Box view.
2. Select the agent you want to use from the drop-down list.

**Note:** If your agent does not appear in the list, click the **Glass box agent management** link, and define it.

3. Verify that one or both of the two main glass box scan options are selected:
  - Use glass box in the Explore stage
  - Use glass box in the Test stage

**Note:** The **Skip equivalent black box tests** check box is cleared by default. This means that both glass box tests and black box tests for the same vulnerability (WASC Threat Classification) are sent. This is because although the glass box tests are generally both more accurate and give more detailed results, occasionally a glass box test may fail while the equivalent black box test succeeds. If the results for your application are unchanged when black box tests are skipped, you can reduce scan time by selecting this check box.

4. Click Scan > Full Scan to start the scan.

The scan starts and the status bar messages indicate that glass box scanning is active.



The scan results will include glass box data in the Issue Information tab where available.

## Uninstalling the glass box agent

This section describes how to uninstall the glass box agent from your .NET server.

### About this task

To perform this task you will need access to your application server.

## Procedure

Do one of the following:

- Use the Microsoft Windows Add/Remove Programs feature.
- Run GB\_.NET\_Setup.exe again, and select the Uninstall option in the wizard.

---

## Partial scans

Several options on the Scan menu let you perform part of a scan, or scan part of your site.

### Explore Only

#### About this task

You can let AppScan Explore your site and build the site model, but not proceed with the Test stage of scanning. That way you can examine the site model and application data gathered to verify that it meets your requirements, and also explore manually if necessary, before proceeding with a Full Scan.

## Procedure

After configuring the scan, on the toolbar, click Scan ➔ > Explore Only ➔ .

AppScan explores the site, builds a site model in preparation for the Test stage. When you have examined the application data gathered, it is recommended to proceed with a Full Scan (see “Starting scans from the Scan menu or the toolbar” on page 127

## Test Only

### About this task

If you previously ran Explore Only, or if you stopped a full scan before it was complete, you can let AppScan Test your site using existing Explore results. This can save scan time if, for example, the site's structure has not changed, but you want to see the effect of security changes that have been made to it.

**Note:** Note that during a Full Scan there are typically parts of the site that are only revealed during the Test stage. For these AppScan runs an additional phase of Exploring and testing. There may be several such phases in a scan. If you scan using the Explore Only and Test Only functions, the result is a single-phase scan.

### Procedure

After configuring the scan, on the toolbar, click **Scan**  > **Test Only** .

AppScan tests the site using the existing Explore results.

## Re-Scan

### About this task

The Re-Scan option discards all or part of any existing scan results depending on the selection you make. There are three options:

- **Re-Scan (Full):** Deletes all scan results, and runs a full scan using the current configuration
- **Re-Explore:** Deletes all scan results and runs an Explore stage only using the current configuration
- **Re-Test:** Deletes Test results and runs a new Test stage using the current configuration and Explore results

### Procedure

On the Scan menu click **Scan** >, then select the relevant option (only valid options are active, others are grayed out):

- **Re-Scan (Full)**
- **Re-Explore**
- **Re-Test**

## Scan Multi-Step Operations Only

You can define a multi-step operation and use it to test just that part of the site.

### About this task

A multi-step operation is a sequence of requests that must be sent in a particular order to reach certain parts of the application. (For more details, see "Multi-Step Operations view" on page 79.) In some cases these operations may be the only part of the application you want to scan.

### Procedure

1. Configure the scan, including at least one multi-step operation. (See "Multi-Step Operations view" on page 79.)
2. (Recommended:) Run Scan Expert to evaluate the configuration. (See "Scan Expert" on page 134.)
3. On the Scan menu, click **Scan Multi-Step Operations Only**.

AppScan scans the configured operations, and displays the results.

**Note:** Note that only tests directly related to the sequence are sent. (Site infrastructure tests, for example, are not sent.) This means that when the scan is over, the Status Bar (bottom left) may indicate that there are more tests to be sent. If you are using the Scan Multi-Step Operations function as a way of running a *full* scan on *part* of the site, you should continue with Test Only (next step), to send these tests.

4. (Optional:) Click **Scan > Test Only** to complete a full scan of the part of the site being tested. Infrastructure and other remaining tests are sent to the site, and the results are added to the scan results.

## Changing the configuration during a scan

If you change the configuration after you have started a scan, you will need to rerun the scan, or at least the Test stage, to see the effect of the changes. In general:

- If you changed Explore configurations, you should fully re-scan the application, by running Explore and Test (**Scan > Re-Scan > Re-Scan (Full)**).
- If you changed the Test configurations, you need not run the Explore stage again (if it is complete), and can simply re-test the application (**Scan > Re-Scan > Re-Test**).

---

## Result Expert

Result Expert consists of various modules that are used to process scan results. The processed results are added to the Issue Information tab of the Detail pane, making the information displayed there more comprehensive and detailed.

Result Expert can be configured to run automatically following a full scan, or can be run manually at any time, on full or partial scan results.

When time is limited and the volume of results is large, you may decide not to run Result Expert, or to disable one or more of its modules.

- To run Result Expert manually on existing results, click **Tools > Run Result Expert**.
- To configure Result Expert to run automatically at the end of a scan, click **Tools > Options > Preferences**, and select **Run Result Expert after scan**.
- To configure which Result Expert modules run, click **Configuration > Result Expert**.

See “Issue Information tab” on page 178

For configuration details, see “Result Expert view” on page 98

---

## Exporting scan results

When a scan is complete, the results are displayed on the main window. The different views (Issues, Remediations, Application Data) offer the scan results, filtered for usage.

You can export the scan results from AppScan in different ways:

- Configure and generate an AppScan report; export to PDF or other readable, portable format. See Chapter 10, “Reports,” on page 197.
- Select test variants from Issues and allow AppScan to attach a zip file of the variant information to a new e-mail. See Chapter 8, “Results: Security Issues,” on page 173.
- Generate a database or an XML file from the complete scan results. See “Generating scan result DB and XML files” on page 162 below.

# Generating scan result DB and XML files

## Before you begin

You can export the complete scan results as an XML file, or as a relational database. (The database option exports the results into a Firebird database structure. This is open source, and follows ODBC and JDBC standards.)

The schema for the XML output is named **ScanExport.xsd** can be found in your AppScan \Docs folder, for example:

C:\Program Files (x86)\IBM\AppScan Standard\Docs\ScanExport.xsd

## Procedure

1. Click **File > Export** and select **XML or DB**.
2. Browse to the location you want, and type in a name for the file.
3. Click **Save**.

## What to do next

"Accessing the information in an exported database file"

## Accessing the information in an exported database file

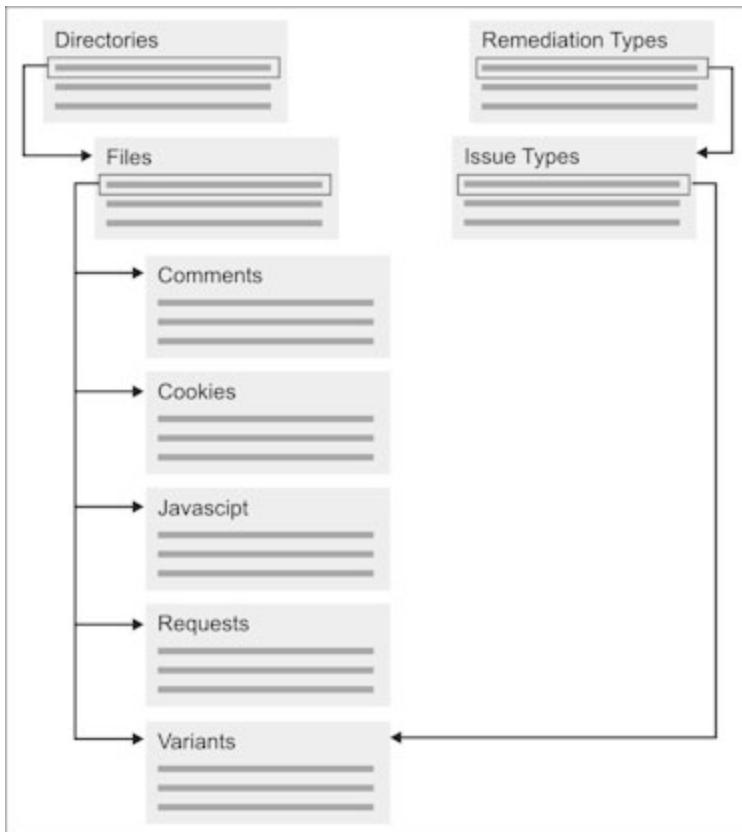
### Procedure

1. Download the Firebird database engine: <http://firebird.sourceforge.net/index.php?op=files&id=engine>
2. Download a Firebird driver (choose one):
  - **Firebird ODBC driver:** <http://firebird.sourceforge.net/index.php?op=files&id=odbc>
  - **Firebird JDBC driver:** <http://firebird.sourceforge.net/index.php?op=files&id=jaybird>
3. Open Firebird with username **SYSDBA** and password **masterkey**, using any ODBC or JDBC compliant client.

## Firebird database structure

When scan results are exported into a Firebird database structure, they can be viewed using one of many ODBC and JDBC database viewers. The structure of the relevant database components is illustrated in the figure below, and described in the following sub-sections.

**Note:** Some fields in the database are internal AppScan fields and not relevant to the user. These fields are marked "AppScan Internal" in the tables following.



## Directory

The Directory section of the database contains a line for each directory or sub-directory from the scan.

Field Name	Explanation/Comments
ID	AppScan ID for the directory
NAME	Directory name
PARENTID	Directory in which this directory is contained (if this is a sub-directory)
PATH	Directory path
DIRTYTYPE	Directory type: Host/Application

## Files

The Files section of the database contains a line for each file.

Field Name	Explanation/Comments
ID	AppScan ID for the file
FILENAME	Filename
PARENTID	ID of the directory that contains this file

## Comments

The Comments section contains a line for each HTML comment found in the site page.

Field Name	Explanation/Comments
ENGINEID	AppScan ID for the comment
FILEID	ID of file where comment was found
SHORTTEXT	Text of comment (may be truncated)

## Cookies

The Cookies section contains a line for each cookie found.

Field Name	Explanation/Comment
ID	AppScan ID for the cookie
REQCOOKIEID	AppScan Internal
RESPCOOKIEID	AppScan Internal
NAME	Name of cookie file
COOKIEVALUE	Value of cookie
SETINURL	Source URL for the cookie
FILEID	AppScan ID of the file where the cookie is saved
PATH	Path attribute of cookie
COOKIEDOMAIN	Domain attribute of cookie
EXPIRES	Date cookie expires
SECURE	Secure attribute of cookie

## Issue types

The Issue Types section contains a line for each issue found.

Field Name	Explanation/Comment
ENGINEID	AppScan ID for this issue
NAME	Issues name as it appears in the GUI
INVASIVE	Whether issue is invasive: Y/N
SEVERITY	Severity code of this issue as shown in the AppScan GUI: 4 = All 3 = High 2 = Medium 1 = Low 0 = Informational
THREATCLASS	Threat classification as it appears in the GUI
THREATCLASSREFERENCE	URL to Internet reference on this classification (if relevant)
REMEDIATIONTYPEID	AppScan ID for the Remediation Type
ADVISORYID	AppScan Internal

Field Name	Explanation/Comment
ENTITYTYPE	AppScan Internal
INFRASTRUCTURE	AppScan Internal

## Javascript

The Javascript section contains a line for each Javascript found in the site pages.

Field Name	Explanation/Comments
ENGINEID	AppScan ID for the Javascript
FILEID	ID of file where Javascript was found
SHORTTEXT	Text of Javascript (may be truncated)

## Remediation types

The Remediation section contains a line for each Remediation.

Field Name	Explanation/Comments
ENGINEID	AppScan ID for the remediation type
NAME	Name of the remediation type as it appears in the GUI
REMEDIATIONPRIORIY	Remediation priority code (1=highest)

## Requests

The Requests section contains a line for each test request sent.

Field Name	Explanation/Comments
ENGINEID	AppScan ID for the test request
FILEID	AppScan ID of the file to which the request was sent
URL	URL of the request
QUERY	Parameters sent in the request
STATUS	AppScan Internal
REASONID	AppScan Internal
XMLTYPE	AppScan Internal
LOGINTYPE	AppScan Internal

## Script parameters

This section contains a line for each parameter that was sent in test request.

Field Name	Explanation/Comments
ENGINEID	AppScan ID for the parameter
FILEID	ID of file where parameter was sent
NAME	Actual parameter that was sent
PARAMETERTYPE	Parameter type: Get/Post

## **Variants**

The Variants section contains a line for each variant.

Field Name	Explanation/Comments
ENGINEID	AppScan ID for the variant.
ENTITYNAME	Name of cookie to which variant sent (is applicable)
FILEID	ID of file to which variant was sent
ISSUETYPE	Issue type name as it appears in the GUI
REMEDIATIONTYPE	AppScan Internal
SEQUENCEINDEX	AppScan Internal

---

## Chapter 7. Results: Application Data

AppScan offers three ways of viewing and working with scan results: Application Data, Security Issues, and Remediation Tasks. This section describes Application Data View.

This view is useful for verifying that all the parts of the site that you want covered by the scan are indeed explored, before you begin the Test stage of the scan. It shows results from the Explore stage only, without reference to the Test stage.

On the **View Selector**, click:



---

### Application Data: Application Tree

The **Application Tree** shows explored folders, URLs, and files.

After the Explore stage, you can review the **Application Tree** to easily view your application and make sure that everything was explored.

The node that you select in the **Application Tree** acts as a filter for the data listed in the **Result List**. If there are no results for a specific type of data, select a higher node in the tree. If you select **My Application**, all data of a specific type are listed.

---

### Application Data: Result List

The Result List shows lists of URLs, parameters, and scripts that were discovered during the Explore stage, *for the selected node in the Application tree*. The table below lists the categories of data.

Data Type	Description
"Requests" on page 168	URLs that AppScan accessed.
"Parameters" on page 168	Scripts and their associated parameters that AppScan discovered.
"Cookies" on page 169	Cookies that AppScan discovered.
Pages	All pages explored by AppScan.
"Failed Requests" on page 170	Links that did not respond to requests.
"Filtered URLs" on page 170	URLs that were not explored, either because of default AppScan settings or because of your defined Explore Filters (see "Exclude Paths and Files view" on page 56).
"User Interaction Needed" on page 171	URLs that require user input which AppScan could not provide automatically. To learn how to define form parameter input, see "Login / Logout tab" on page 46.
"Comments" on page 171	Comments on your web pages that are accessible to users.
"JavaScript" on page 172	JavaScripts that AppScan discovered.

In the Result list:

- Mouse-over each data category (Requests, Parameters, and so on) to see the number of items in that category (for the selected node in the application tree).
- Click on a category to display the items in that category (in the Result list)
- Click on an item in the Result list to display its details in the Detail Pane

The Result List and the Detail Pane for each data type are described in the following sections.

## Requests

This view lists requests for which AppScan received a valid response. Based on these responses, AppScan generates tests, to reveal weaknesses in the site, that will be sent during the Test stage.

If the request or the response body contains XML (including XHTML or SOAP), the Visited URLs icon is replaced with the XML icon: 

### Requests in the Result List

The **Result List** shows the URL of each page that AppScan visited, as well as method and parameters.

- You can view a visited URL by right-clicking on a request and clicking **Show in Browser**, or selecting the URL and clicking the **Show in Browser** link in the **Detail Pane**.
- You can create a manual test for the URL by right-clicking the **Visited URL** item and clicking **Manual Test**, or selecting the URL and clicking the **Manual Test** link in the **Detail Pane**. (See “Manual tests” on page 189 to learn more.)

### Requests in the Detail Pane

The **Detail Pane** includes links to **Show in Browser** and **Manual Test**, which operate in the same manner as the right-click commands in the **Result List**.

The **Request/Response** tab is displayed and shows the request and immediate response of the URL selected in the **Result List**.

## Parameters

These are requests that included one or more parameters.

The **Result List** shows all the parameters found during the Explore stage. The URLs in this list are the ones most likely to be vulnerable to malicious attacks. This list can be crucial in evaluating whether the scan has generated a useful set of test requests.

For each parameter in the Script Parameters list, AppScan shows the name, type, value, and URL (Results pane) and value (Detail Pane), and whether it is tracked. One parameter name may be listed more than once, if it is on different URLs, or if it has different values on the same URL.

The table below shows the options available when you right-click on an item in the list.

*Table 3. Right-click options*

Option	Function
Copy URL	To copy the selected URL to the clipboard.
Add to list in Parameters and Cookies tab	To add the selected parameter name (all values) to the list in the Configuration dialog box by right-clicking and selecting <b>Add to Parameters and Cookies</b> list. The Parameter Definition dialog box opens, to configure AppScan’s treatment of this parameter.

Table 3. Right-click options (continued)

Option	Function
Exclude this path-parameter-value combination from the scan	<p>To exclude a specific parameter value from the scan when it occurs in a particular URL. When you select the option, the Edit Exclusion or Exception dialog box opens, with the relevant data filled in.</p> <p><b>Example</b></p> <p>Consider a site with the following URL: <code>http://site/command</code> and a post parameter named 'action'. Each value triggers a different response from the server:</p> <ul style="list-style-type: none"> <li>• <code>action=login</code>, redirects to login page</li> <li>• <code>action=logout</code>, session expires</li> <li>• <code>action=clean</code>, server deletes user data</li> </ul> <p>For AppScan to be able to scan this site it must exclude <code>http://site/command</code> when <code>action=logout</code> or <code>clean</code> but not when <code>action=login</code> or other values. This can be done using this feature to exclude <code>http://site/command</code> with parameter name <code>action</code> and value <code>logout</code> or <code>clean</code>.</p> <p>For more details, see "Adding new exclusions or exceptions" on page 57</p>
Do not test selected parameter(s)	<p>To exclude one or more parameter names (all values) from the Test stage of the scan. This setting applies to all values of the specified parameter. It does not affect the Explore stage.</p> <p>The parameter name is added to the list in Parameters and Cookies view of the Configuration dialog box, with its <b>Test Exclude</b> value set as "Yes".</p> <p>For more details, see "Parameters and Cookies view" on page 65</p>

## Cookies

*Cookies* lists all cookies that AppScan found during the scan, whether set by responses, generated by Javascripts, or already on the host before the scan.

- The Result List shows all cookies found during the Explore stage. For each cookie the list shows whether tracked, name, set-cookie URL, values, whether excluded from testing, path, domain, expire date, whether secure, context.

**Note:** The URL listed is the one that leads to the response that set the cookie (even if there were others that included the Set Cookie command). If the cookie was not set by a response (for example, if it was generated by a Javascript, or already present on the host) N/A is displayed in the Response URL field.

- You can add any cookie to the list in the Configuration dialog box by right-clicking and selecting **Add this cookie to the Parameters and Cookies tab** list. The Parameter Definition dialog box opens, to configure AppScan's treatment of this cookie.
- When a particular cookie is selected the Detail pane shows:

Item	Description
Path	The specific folder or subfolder in your application to which this cookie is sent. The path attribute is used to specify the subset of URLs in a domain for which the cookie is valid. If a cookie has already passed domain matching (next item), then the pathname component of the URL is compared with the path attribute, and if there is a match, the cookie is considered valid and is sent along with the URL request.
Domain	To which domain or sub-domain this cookie will be sent. (If Domain is not set, the cookie is sent to the domain which issued the Set Cookie command, and all sub-domains.)
Expires	The date and time that the cookie will expire and be removed from the user's machine.

Item	Description
Secure	Yes (secured) or No. If a cookie is marked <b>Secure</b> , it will only be transmitted if the communications channel with the host is secure (currently only to HTTPS servers). If <b>Secure</b> is not specified, a cookie is considered safe to be sent over all channels.
Request URL	The first request that AppScan sent with the cookie.

## Failed Requests

These are requests sent that did not return a valid response. This usually happens when the site is down or when other communication problems exist; or when a request returns an error response status instead of the requested page.

**Note:** If a URL in the application tree contains *only* error responses, the URL is displayed with strike through formatting (with a line through it).

- Where an error page is the expected response
- You can resend the request to a broken link, rather than repeat the entire scan. If the application has been changed or fixed since the Explore stage was done, perform the following procedure to re-explore the broken links.

**Note:** If your application requires a login, it is recommended that you manually log AppScan in (see “Login / Logout tab” on page 46) before doing the following procedure; otherwise the broken links will be explored outside of typical usage patterns.

- You can view all broken links by selecting **Failed Requests** in the Result pane.
- If you click **Retry All Failed Requests**, these links are removed from the Broken Links list and added to the Unvisited Links list. AppScan then continues the Explore stage, visiting the links in the Unvisited Links list whenever possible. This Explore stage will end when all the Unvisited Links have been explored.

**Note:** If there are communication problems between AppScan and your server during a scan, some links might be marked as Broken Links. When AppScan has communication problems, it attempts to resend its requests for 90 seconds. If the connection is not achieved within that time, the scan stops. The Notice panel on the main window notifies you of the problem and shows the timeout count down. If you see this notification, you should fix the connection between AppScan and your application before attempting to troubleshoot the links.

- The Detail Pane shows the response page of specific broken link, by clicking the **Show in Browser** button.

## Filtered URLs

These are URLs that AppScan did not visit, because they were filtered out of the Explore: either by standard filters, or by filters you defined when you configured the scan (see “Exclude Paths and Files view” on page 56).

### Filtered URLs in the Result List

The **Result List** shows the URLs that were not explored and provides the Filter Type (the reason why this page was filtered out).

You can view a filtered URL by right-clicking a Filtered URL and then clicking **Show in Browser**.

## Filtered URLs in the Detail Pane

The **Detail Pane** offers a link to show the URL in a browser: **Show in Browser**. This is equivalent to the right-click > **Show in Browser** in the **Result List**.

The **Request/Response** tab shows the request that would have been sent to the URL if it had not been filtered out.

## User Interaction Needed

These are requests that were not sent because they require input from the user that AppScan could not supply. You can configure AppScan to provide input; see “Automatic Form Fill view” on page 75. If you missed some of your application parameters, or if you chose not to use the automatic form filler, AppScan will have a list of interactive URLs for you to review.

- You can examine the interactive URLs list. If you want these pages to be scanned, supply the required user information in a Manual Explore.
- It is recommended that you carefully examine the list of interactive URLs, fill in the required data, and send these requests. AppScan will then include these URLs during the Test stage.
- By enabling AppScan to send these requests, whole new parts of the site which were previously inaccessible may be reached. Therefore, after you visit interactive URLs, you should re-explore your application (**Scan > Re-scan > Explore**).

See “Manually exploring interactive URLs”).

### Manually exploring interactive URLs

#### About this task

Interactive URLs are requests that were not sent because they require input from the user that AppScan could not supply. If you missed some of your application parameters, or if you chose not to use the automatic form filler, AppScan will have a list of interactive URLs at the end of a scan.

#### Procedure

1. In **Data view > Result List**, click **User Interaction Needed**.

The list of URLs with interactive input is displayed.

**Note:** A single URL may appear more than once, each instance being from a different form.

2. Right-click a URL in the list and click **Manually Explore this URL**.

The browser, with the **Manual Explore** buttons, appears, open to this URL. Complete the manual explore; see “Recording a Manual Explore” on page 116.

After you finish the manual explore, AppScan analyzes the new Explore.

If new URLs were found, a message appears, recommending that you continue the Explore (**Scan > Continue Scan > Explore**).

- If no new URLs were found but new tests were created, a message appears, recommending that you go on to the Test stage (**Scan > Continue Scan > Test**).
- If both new URLs were found and new tests were created, a message appears, recommending that you continue both the Explore and the Test (**Scan > Continue Scan > Full Test**).

## Comments

*Comments* are HTML comments that AppScan found during the Explore stage. Comments hidden in an HTML page can contain information that hackers find useful: sometimes developers, intentionally or accidentally, leave comments for themselves or other developers in the final page. A hacker can harvest useful internal information from these comments, such as debug passwords.

## Comments in the Result List

The list of comments shows the first line of the comment, and the first URL where it is located. If AppScan finds the same comment more than once, only the first instance is listed.

## Comments in the Detail Pane

The **Detail Pane** shows the entire comment of the selected item in the **Result List**. Review the comments here to determine which should be removed from the final application.

## JavaScript

*JavaScript* lists the JavaScript code that AppScan found during the Explore stage.

### JavaScript in the Result List

The list shows the first line of the JavaScript, and the first URL where it was found. If the same script is to be found on more than one URL, only the first instance is listed.

### JavaScript in the Detail Pane

The **Detail Pane** shows the entire script of the selected item in the **Result List**. Review the code here for comments that should be removed from the final application.

---

## Application Data: Detail Pane

In Application Data view, the Detail Pane shows a variety of data and toolbar options, depending on the data type selected.

Data type	Toolbar options
Requests	Show in Browser, Set as Error Page, Manual Test, Search (for word/phrase), Search (for word/phrase).
Parameters	Requests (toggle through), Show in Browser, Set as Error Page, Manual Test, Search (for word/phrase).
Cookies	None.
Pages	Page information tab. Request/Response tab: Requests (toggle through), Show in Browser, Set as Error Page, Manual Test, Search (for word/phrase).
Failed Requests	Show in browser, Set as Error Page, Search (for word/phrase).
Filtered	Show in browser, Set as Error Page, Search (for word/phrase).
User interaction needed	Manually explore this URL, Search (for word/phrase).
Comments	None.
JavaScripts	Search (for word/phrase).

---

## Chapter 8. Results: Security Issues

AppScan offers three ways of viewing and working with scan results: Security Issues, Remediation Tasks, and Application Data. This section deals with Security Issues View.

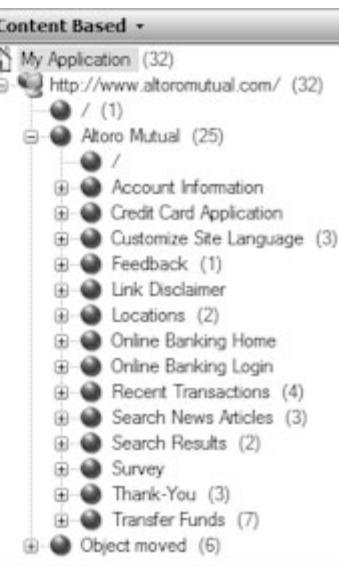
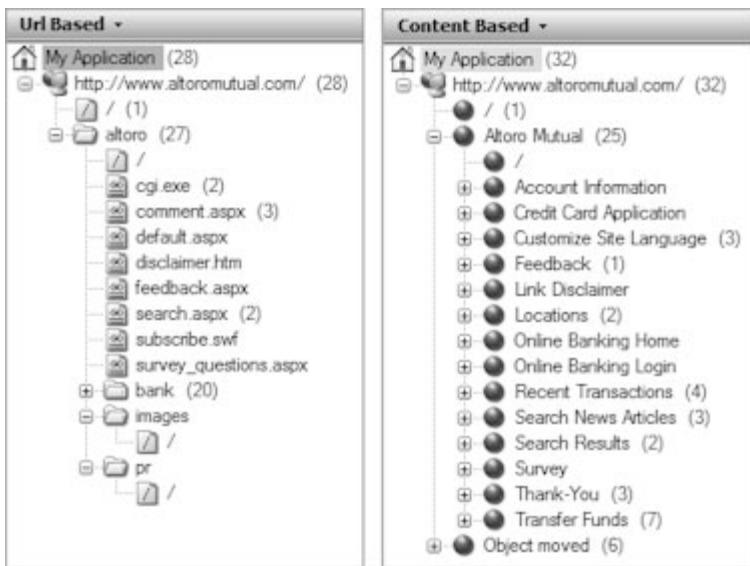
Security Issues View provides access to the results of a scan. You can view results at a high level or select specific tests or objects and access more details. These details include advisories, fix recommendations, requests/responses, and differences between the test variants that resulted in issues. You can manipulate the severity of issues, resend tests (with or without modifications), and create reports based on Issues.

In the **View Selector**, click:



### Security Issues: Application Tree

The **Application Tree** displays the folders and files of the scanned application. Each node in the tree has a counter, which shows how many issues the node contains.



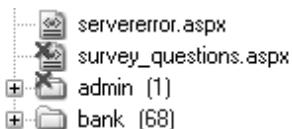
In the Application Tree you can:

- Select a node to filter the issues displayed in the **Result List**.
- Use the right-click menu to View in Browser, Manual Explore, Manual Test, Copy URL to Clipboard, and Exclude URL from Scan. (See “Application Tree right-click menu” on page 22.)
- If you have defined content-based rules, you can toggle between URL-Based view and Content-Based view by clicking the combo-box at the top of the pane. (See “Content-Based Results view” on page 82.)

### Exclude URL from scan

Any URL or node in the Application Tree can be excluded from future scans by right-clicking on it and selecting **Exclude from Scan**. (To reinstate the URL or node, simply right-click again and select **Include in Scan**).

When a URL or node is excluded from scan a red X appears on its icon in the Application Tree.



When you exclude URLs in this way, "Exclude" items are added to the **Exclude or Include Paths** list (see "Exclude Paths and Files view" on page 56).

## Security Issues: Result List

The Result List displays the issues relevant to the node selected in the Application Tree. If you selected the My Application node, the Result List shows all issues found in your web application.

Issues are grouped by Type. Under each Type, all URLs are listed. Under each URL, all issues are listed. (Individual variants of issues are not shown in the Result List, but can be seen from the Detail pane.)

Each node in the tree has a severity icon, indicating issue severity; and a counter, indicating how many issues of this type were found. In the case of Type and URL, the severity icon indicates the severity of the most severe issue included under the node.



You can change the way the issues are sorted, and you can manipulate their severity value (see "Severity levels").

You can designate issues you do not want to deal with currently as Noise, either removing them from the result display altogether, or having them appear with strikethrough (see "Issue state: Open or Noise" on page 175).

The total number of security issues (at the top of the list) is a measure of the vulnerable locations in the site, and depends in part on how site is structured. If you define a content-based structure (see "Content-Based Results view" on page 82), the total number of issues in the application tree may not be the same as it is for the URL-based application tree (for the same results). When site structure is content-based (rather than URL-based), and content-based view is configured correctly, the issue count in content-based view represents more accurately the number of "vulnerable locations" that exist in the site. The total number of *variants* (at the top of list in parentheses) is independent of site structure, and does not change between content-based and URL-based views.

## Severity levels

In the case of issues, the severity icon indicates the severity level of the issue; in the case of Issue Types and URLs, it indicates the most severe of all issues under that node.

Icon	Indicates	Description	Examples
!	High severity	Direct danger to your application, web server, or information	Executing commands on server, stealing customer information, denial of service
!	Medium severity	Threat through unauthorized access to private areas, though the database and operating system are not at risk	Script source disclosure, forceful browsing
!	Low severity	Allow for unauthorized reconnaissance	Server path disclosure, internal IP address disclosure
i	Informational	Issues you should know about, not necessarily security issues	Insecure methods enabled

#### Related tasks:

"Changing severity levels"

### Changing severity levels

#### Procedure

You can change the severity assigned to any node by right-clicking on the node and selecting **Severity >**, followed by the new value.

**Note:** If you change the severity level of a Type or URL node, all issues under that node are changed to the new level.

### Issue state: Open or Noise

Issues not relevant to your application can be designated as "Noise" and removed from the results.

#### About this task

If a specific issue that AppScan discovers is not relevant for your application (meaning that for your application it is effectively a "false positive" result), for example an issue that exists only in the development environment but will not exist in the deployment environment, you may decide to classify it as "Noise".

There are two display options for issues marked as noise: Included in the Result list but as gray text with a strikethrough, and Not included in the result list at all.

#### Procedure

To toggle between the two display options, click **View > Show Issues marked as Noise**.

### Results

When a checkmark appears next to the menu item, issues marked as Noise will be included in the Result list but with gray text with a strikethrough.

### Example

Noise definitions are applied in the following way:

- On the workstation where the scan is configured, noise designations are saved, and automatically applied to future scans, *when the scan is saved*. (The location of the Noise Classification file is defined in Tools > Options > General tab.)

- When you open a saved scan *even on a different workstation*, these issues will be designated as noise *for this scan*, even if the workstation has different definitions. (However, note that if you *save* the scan on the second workstation its definitions will be saved to that workstation, overwriting any previous Noise Classification file there.)

## What to do next

See also:

"Changing Issue State"

"Test Options view" on page 92

### Changing Issue State

You can change the state assigned to any node.

#### Procedure

Right-click on the node and select **State >**, followed by the new value ("Open" for relevant issues; "Noise" for issues not relevant to your application).

**Note:** If you change the state of a Type or URL node, all issues under that node are changed to the new state.

### Exporting and importing Issue states

If you have designated certain Issues as "Noise" (not relevant to your application), you can export these designations for use at other workstations.

#### About this task

To export Noise designations for use at other workstations:

#### Procedure

Click **File > Export > Cross-Scan Data**, and save the data as an XML file.

## What to do next

To import the states at another workstation, click **File > Import > Cross-Scan Data**.

### Resending tests

#### About this task

You can resend a test without running a complete Full Scan or Test stage. For example, if the results of a test seem inconsistent with the results of a previous scan, you can resend that one test.

#### Procedure

- On the **Result List**, right-click a node.
- On the menu that appears, click **Re-test**.

AppScan sends all the test requests included in the selected node, and the new results are added to the **Result List**.

## Right-click menu

The right-click menu in the Security Issues Results List includes the following options:

Item	Explanation
Severity	Change the Severity value for the selected item (select High, Medium, Low, Informational)
State	Issue state is always Open by default. If for some reason you do not need to relate to certain issues, you can define them as Noise. (Options: Open/Noise)
Re-test	Resend the selected test and add the result to the scan results
Manual Test	Create a manual test (see "Manual tests" on page 189)
Delete	Delete the selected item from the test results (cannot be reinstated)
Set as Non-vulnerable	Set the result as non-vulnerable, so it will not appear in the test results (but can be viewed, and reinstated, from the Non-Vulnerable Variants list, see "Non-Vulnerable Variants List" on page 191)
Report False Positive	Zip and email test information to AppScan Support or to members of your own organization, see ("Report false positive test results" on page 187)

## Filtering Security Issues in Result List

You can filter the **Result List** for types of issues, or you can search for a specific issue.

### Procedure

1. On the **Edit** menu, click **Find** (or press **Ctrl + F**).  
The **Find** bar appears in the main window, below the **Result List**.
2. On the **Application Tree**, select a node.
  - If you select the **My Application** node, Find searches through all the results.
  - If you select a node within the tree, Find searches through the selected node and its sub-nodes.
3. In the **Look for** text box of the **Find** bar, type a string or partial string.
4. In the **Search in** combo box, select where in the results to search for the string. Options are: ID number of test, Security Issue, URL, CVE ID, CWE ID, XFDI (X-Force ID), Request/Response data, Variant description, Parameter/Cookie Name, or All of these.
5. Click **Find Now** or press **Enter**.

Results appear in the **Result List**, overwriting the list that was previously displayed.

If you enter another **Find** string and click **Find Now** again, the new search is done on the selected node in the **Application Tree**, not on the displayed results of the previous search.

To find:	Look for:	Search in:
The variant with ID: "5016" (see <b>Properties</b> sub-tab of <b>Detail Pane</b> )	"5016"	ID
Cross-Site Scripting issues	"cross"	Test Name
Login pages with issues	"login"	Test URL
Issues relevant to password manipulation	"passwd"	Request/Response

## Sorting the Result List

### About this task

You can re-arrange issues in the Result List. By default, issues are sorted by severity.

## Procedure

1. Click the **Arranged By** column header.
2. Click a command on the menu to sort the issues:
  - **Severity** - test names are listed by severity order: High Severity first, down to Informational.
  - **Count** - number of issues in each test. Tests that found the most issues are listed first.
  - **Name** - alphabetical order by test name.

The issues are re-sorted in the **Result List**.

To reverse the order of the sort (ascending or descending), click the second column of the **Result List**. Click the second column again to reset the order.

---

## Security Issues: Detail Pane

The Detail Pane shows relevant information for a selected test, and all its variants, selected in the **Result List**.

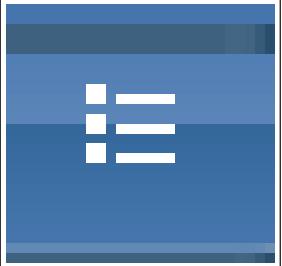
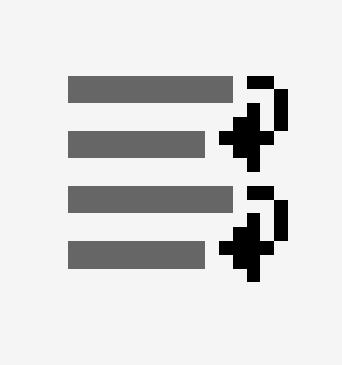
The Detail Pane has four tabs (clicking on a tab brings its contents to the front): “Issue Information tab,” “Advisory tab” on page 183, “Fix Recommendations tab” on page 185, “Request/Response tab” on page 185.

### Issue Information tab

The first tab of the Detail pane summarizes the information available.

During scanning, as issues are discovered and added to the tree, the Issue Information tab provides a summary of the information available on the other Detail pane tabs, as well as valuable additional information such as CVSS Metric scoring for the issue, and relevant screen captures, that can be saved with the results and included in your reports.

Area/Icon	Description
Header	Issue header including URL, Entity and Security Risk (High, Medium, Low or Informational).
CVSS Metrics score	An average score based on the three CVSS metrics groups: Base, Temporal and Environmental. Click on the link to edit (see “CVSS settings” on page 180).

Area/Icon	Description
	Enables you to include screen captures, other relevant images of your choice, and your own comments with the scan results, and include them in reports (see "Editing Issue Information").
Tip (yellow box)	This information refers to the Content area (below) and explains what to look out for in the image(s) or HTML that appear there.
Content (screen capture(s) or HTML code)	<p>Depending on the issue, this area may include a screen capture, two screen captures for comparison, a screen capture with a simulated pop-up, or HTML code. In the case of  HTML, you can toggle text wrap on and off by clicking at the top right of the content area.</p>
Reasoning (blue box)	Explains what AppScan did, and why it thinks this is an issue.
Technical summary (gray box)	The technical details of what AppScan did to test for this issue, and how it validated the response.

#### Note:

If any Result Expert modules were disabled during the scan, the Issue Information may be incomplete. You can update the data at any time by enabling the missing modules in **Configuration > Result Expert**, and then clicking **Tools > Run Result Expert**.

#### Issue Information toolbar

The toolbar above the Detail pane shows the current severity of the selected issue, and its state, and lets you toggle between issues.

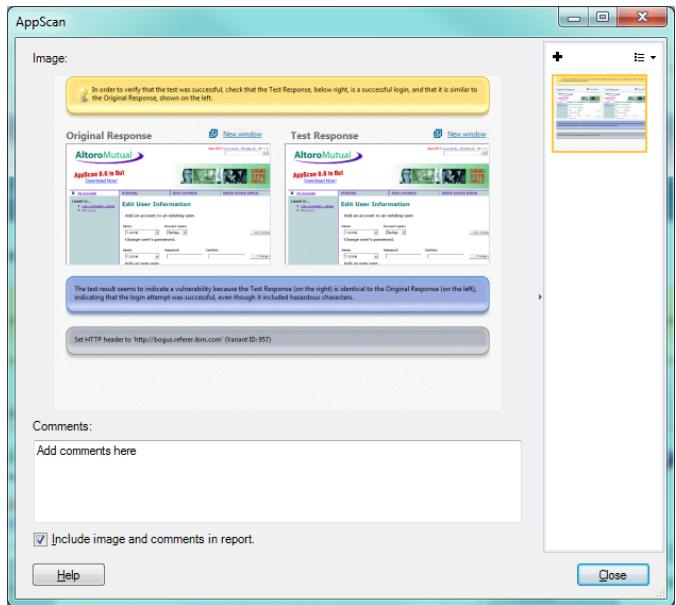
- **Severity:** Select one of the four standard severity settings, or manually adjust the CVSS settings for this issue.
- **State:** Options are "Open" or "Noise". Default is Open. Select Noise for issues that are unimportant to you. By default, results designated as Noise appear in the Result List with a line through them. To remove them from the display altogether, deselect **Tools > Show Issues Marked as Noise**.

#### Editing Issue Information

Add images and comments to the issue information, for inclusion in reports.

Issue Information tab screen captures, other related images of your choice, and your own comments can be saved with the scan results, and included in reports.

- Click  at the top right hand corner of the Issue Information tab to edit issue information.



Area/Icon	Description
Image	Shows the image currently selected from the available thumbnails in the right-hand pane.
	Click to add an image to this issue in the scan results.
	Click to save the current image to your computer.
	Click to delete the current image from the scan results.
Comments	Type in comments about the currently displayed image, to be saved with the scan.
Include image and comments in report	Select to include the <i>currently displayed image and comments</i> in reports. You can configure each image separately. All images are included by default.

## CVSS settings

You can manually fine-tune the severity setting for a specific issue based on CVSS metrics. This is done from the Issue Information toolbar, by clicking **Severity > CVSS Settings**.

txtSearch - search.aspx - Link Injection (facilitates Cross-Site Request F... X

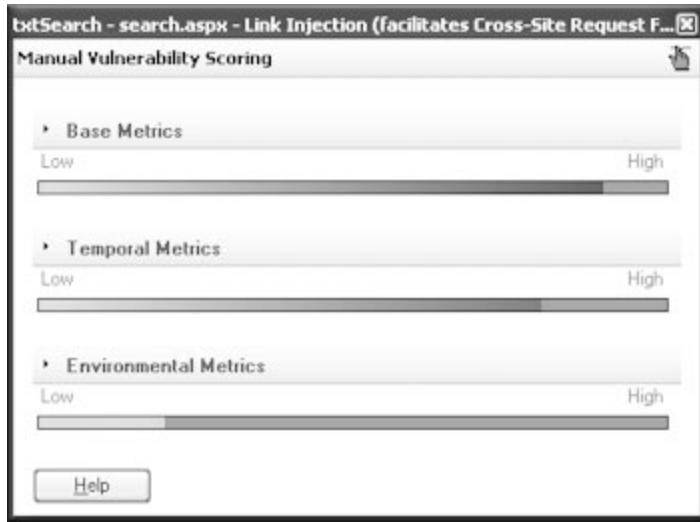
Manual Vulnerability Scoring 

► Base Metrics  
Low  High

► Temporal Metrics  
Low  High

► Environmental Metrics  
Low  High

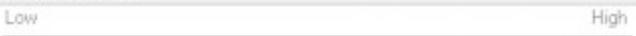
[Help](#)



From the CVSS window, clicking on the name of one of the three sections opens that section for configuration. You can restore the default settings by clicking  , which becomes active when changes are made.

txtSearch - search.aspx - Link Injection (facilitates Cross-Site Request F... X

Manual Vulnerability Scoring 

► Base Metrics  
Low  High

► Temporal Metrics  
Low  High

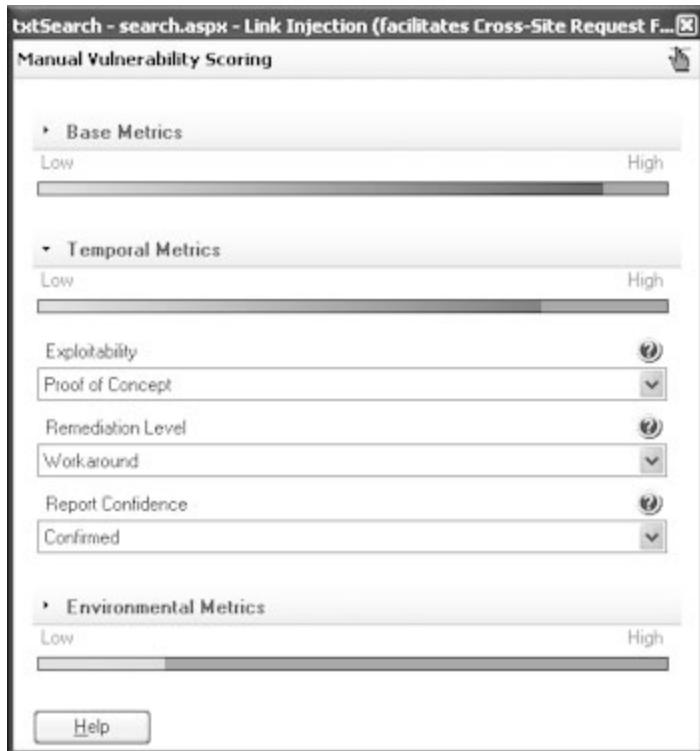
Exploitability   
Proof of Concept 

Remediation Level   
Workaround 

Report Confidence   
Confirmed 

► Environmental Metrics  
Low  High

[Help](#)



## Base metrics

These are metrics of the vulnerability that are constant over time and across user environments.

Metric	Explanation	Options
Access Vector	Whether the vulnerability can be exploited only locally, also from adjacent networks, or from any network connection ("remotely exploitable").	Local, Adjacent Network, Network
Access Complexity	The difficulty involved in exploiting this vulnerability.	High, Medium, Low
Authentication	The number of times an attacker must authenticate to exploit the vulnerability.	None, Single, Multiple
Confidentiality Impact	The impact on confidentiality if this vulnerability is successfully exploited.	None, Partial, Complete
Integrity Impact	The extent to which system integrity (the accuracy of information supplied by the application) is compromised if this vulnerability is successfully exploited.	None, Partial, Complete
Availability Impact	The impact on the availability of information resources if this vulnerability is successfully exploited.	None, Partial, Complete

## Temporal metrics

These are metrics of the vulnerability that may change over time.

Metric	Explanation	Options
Exploitability	The current state of exploit techniques utilizing this vulnerability.	Unproven, Proof-of-Concept, Functional, High, Not Defined
Remediation Level	The level of remediation available to protect against the vulnerability.	Official Fix, Temporary Fix, Workaround, Unavailable, Not Defined
Report Confidence	The degree of confidence in the existence and technical details of the vulnerability.	Unconfirmed, Uncorroborated, Confirmed, Not Defined

## Environmental metrics

These metrics reflect the application environment, and should be set globally using the Configuration dialog box > Environmental Metrics tab. Change them here only if this vulnerability is specific to a part of the application environment that has different characteristics.

Metric	Explanation	Options
Collateral Damage Potential	The potential for damage or theft if the application is vulnerable.	None, Low, Low-Medium, Medium, Medium-High, High, Not Defined
Target Distribution	The proportion of systems in the environment that are potential targets.	None, Low, Medium, High, Not Defined
Availability Requirement	The relative importance of availability (of information).	None, Low, Medium, High, Not Defined
Confidentiality Requirement	The relative importance of confidentiality (of user information).	None, Low, Medium, High, Not Defined
Integrity Requirement	The relative importance of integrity (accuracy) of information.	None, Low, Medium, High, Not Defined

## Restoring default severity settings

### Procedure

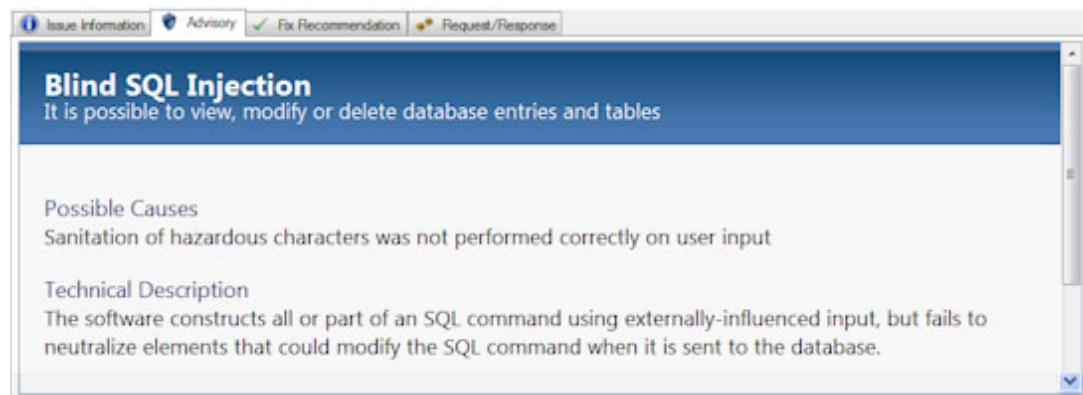
If you have manually changed the severity setting for a particular issue (by selecting a different severity or by adjusting the CVSS settings) you can restore the previous setting.

- To replace your manual setting (High/Medium/Low/Informational) with a setting based on the current CVSS settings for this issue: On the Issue Information toolbar, click **Severity > Use CVSS to calculate severity**.
- To restore the *default* CVSS settings: On the Issue Information toolbar, click **Severity > CVSS Settings**, and in the CVSS settings window that opens click 

## Advisory tab

The second tab of the Detail pane is the advisory.

The information on the Advisory tab provides technical details on the selected issue and reference links for more information. This information is essential when you need to explain what has to be fixed and why.



The **Advisory** tab may include any of the following sections:

#### Test Name

The name of the test as it appears in the Result List.

#### Severity

The severity assigned to this vulnerability.

#### Type

Whether this vulnerability is application-level or infrastructure-level.

#### WASC Threat Classification

An Internet link to the page of the web Application Security Consortium that describes this class of threat.

#### CVE ID(s)

The Industry Standard number(s) for this type of vulnerability (see "CVE support" on page 184.)

#### CWE ID(s)

The Industry Standard number(s) for this issue (see "CWE support" on page 184.)

#### XFID

The X-Force ID for this issue (see "X-Force support" on page 185.)

#### Security Risk

An explanation of how this issue is a security risk to your application.

### **Training Module**

An Adobe Flash presentation that explains the and demonstrates the issue.

### **Possible Causes**

Suggests how the issue came to exist in your application

### **Technical Description**

Detailed technical description of the issue

### **Affected Products**

Third Party products that could be affected by the issue.

### **References and Relevant Links**

Links to additional information.

## **CVE support**

CVE (Common Vulnerabilities and Exposures) is an industry standard list that provides common names for publicly known information security vulnerabilities and exposures. This makes it easier to share data across separate databases and tools. (For more details see the CVE website at: <http://cve.mitre.org/>)

AppScan advisories for vulnerabilities that have been assigned a CVE ID, include the reference, with a link to the description on the CVE website. (CVE IDs begin with the letters CVE for accepted vulnerabilities or CAN for candidate ones.)

You can:

- View the CVE ID of a test result in the advisory for the test (see “Advisory tab” on page 183)
- Click on the CVE ID to link to its description on the CVE website
- Include CVE IDs (as part of the advisory) in reports
- See the CVE ID that is tested for by any particular test, in Test Policy view (Scan Configuration > Test Policy; see “Test Policy view” on page 87)
- List all tests that have CVE IDs by searching for CVE abd CAN strings in Test Policy view
- Search for a particular CVE by searching for its ID in Test Policy view

## **CWE support**

CWE (Common Weakness Enumeration) is an industry standard list that provides common names for publicly known software weaknesses. This makes it easier to share data across separate databases and tools. (For more details see the CWE website at: <http://cwe.mitre.org/>)

AppScan advisories for vulnerabilities that have been assigned a CWE ID include the reference number, and a link to the description on the CWE website. Where specific vulnerabilities have their own CWE ID (in addition to the ID for the *issue*), this is shown in the Variant Details pane.

You can:

- View the CWE ID for an issue in the advisory for that issue (see “Advisory tab” on page 183)
- View the CWE ID for a variant, and its parent issue, in the Variant Details tab (see “Variant details” on page 187)
- Click on the CWE ID to link to its description on the CWE website
- Include CWE IDs (as part of the advisory) in reports
- Search for a particular CWE by searching for its reference number in Test Policy view

For the version of the CWE database used in the current version of AppScan, refer to the Release Notes<sup>®</sup> (generally located in C:\Program Files\IBM\AppScan Standard\Docs). If a daily update changes the database used, the change will be listed in the Update Log (**Help menu > Update Log**).

## X-Force support

X-Force is one of the world's most comprehensive threats and vulnerabilities database. All AppScan advisories for vulnerabilities include their Xfid, and a link to the X-Force site where you can find additional information about the security issue.

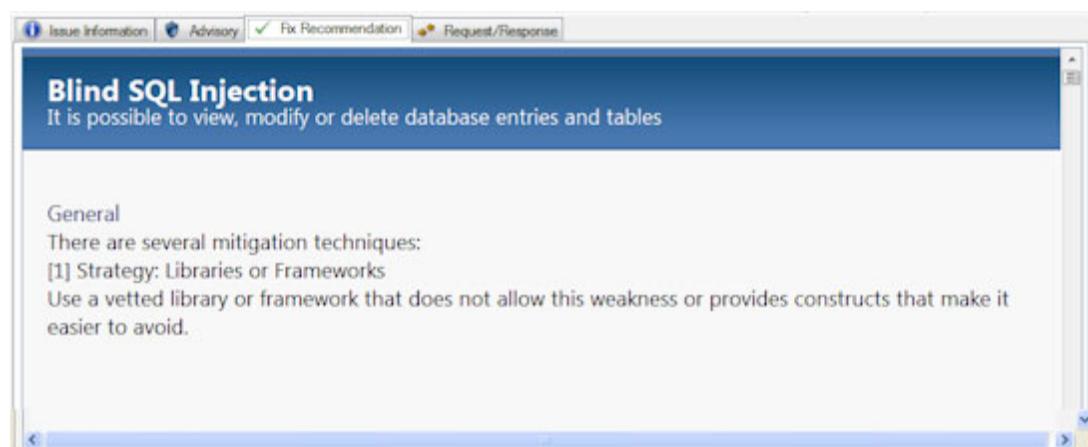
You can:

- View the Xfid for an issue in the advisory for that issue (see "Advisory tab" on page 183)
- View the Xfid for a variant, and its parent issue, in the Variant Details tab (see "Variant details" on page 187)
- Click on the Xfid to link to its description on the X-Force website
- Include Xfids (as part of the advisory) in reports
- Search for a particular Xfid by searching for its reference number in Test Policy view

## Fix Recommendations tab

The third tab of the Detail pane shows the fix recommendation.

The information on the Fix Recommendations tab is the exact tasks that should be done to make your web application secure against the specific selected issue.



The **Fix Recommendations** tab shows known recommendations for fixing the selected issue. These solutions may be quite involved, step-by-step instructions.

Fix Recommendations are sorted:

- **General** - always selected
- **.Net** - Microsoft® .NET
- **Java EE** - Sun® Java Platform, Enterprise Edition

**Note:** You can set AppScan to hide the Fix Recommendations that you do not find relevant. See "Preferences tab" on page 218 for details.

## Request/Response tab

The fourth tab of the Detail pane is the Request/Response tab.

The **Request/Response** tab provides information about the tests, and their specific variants, that were sent to your web application to discover where it has weaknesses. A test may have multiple variants. A *variant* is a slight change to the original test request that AppScan sends to your web application server. (AppScan first sends a request that is meant to be legal and to follow the business logic of your

application. Then it sends a similar request, modified for the purpose of discovering how your application handles illegal or mistaken requests. Each test request may have a number of variants; as many variants as needed to cover all the security rules in the extensive AppScan database.)

For example, consider a test sent to check that you have enforced user input rules for a specific parameter. One variant might checks that apostrophes are not valid input, another that quotation marks are not allowed.

The variant itself is shown in red text, and the validation (the part of the response that indicates the existence of a security issue) is highlighted in yellow.

Besides a large amount of explanatory information, the **Request/Response** tab provides advanced features for understanding and using the results of a scan.

The **Request/Response** tab has two panes and its own toolbar along the top. The toolbar and tabs are shown below, and summarized in the table following.

Tool	Function
Variant < >	Indicates the number of variants of the current test. Clicking the < and > icons toggles to the previous and next variant respectively.
Test/Original	Toggles between the Original and Test information.
Next Highlight	(Available where validation text is highlighted). Moves cursor to the next highlighted text.
Show in Browser	Opens the built-in browser to show the current page, with the option of taking a screen capture from the browser.  When the browser opens you can take a screen capture of the page by clicking the camera icon  on the browser toolbar. The screen capture is added to the Issue Information tab.
Options > Report False Positive	Use to email the current variant to the AppScan support team, or within your enterprise. (See “Report false positive test results” on page 187.)
Options > Manual Test	Modify test and save it as a manual test. (See “Manual tests” on page 189.)
Options > Delete Variant	Permanently deletes the selected variant from the test results (irreversible). This can also be done by right-clicking on the variant in the Result pane.
Options > Set as Non-vulnerable	Changes the definition of the selected variant to Non-vulnerable.  Positive responses that were changed by the user to non-vulnerable are removed from the scan results, and do not appear in reports, but can be viewed (and reinstated) via the Non-Vulnerable Variants list. (See “Non-Vulnerable Variants List” on page 191.)
Options > Set as Error Page	Adds the current page to the list of error pages (Scan Configuration dialog-box > Error Pages) and updates the results to reflect the fact that this response is an error page.
Options > Add to Issue Information	Runs Result Review on the current issue, and adds any new information available to the Issue Information tab.
Find	Type in text to search for a specific string. (See “Filtering Security Issues in Result List” on page 177.)
Variant details	The right pane shows details for the current variant: ID, Description, Difference (the difference between this variant and the original request), Reasoning, and CWE ID.

## Viewing variants

### About this task

Each test can have multiple relevant variants; each variant changes a request slightly to check your application's security against numerous attack techniques.

### Procedure

1. Click **Test**.
2. Click the right or left arrow buttons to view the variant requests.

For each variant test sent, the modified part of the request is highlighted in red. For a more descriptive explanation of the variant, see the **Variant Details** tab.

### Variant details

Variant Details is a side tab in the Request Response tab of the Detail pane.

The **Variant Details** tab, within the **Request/Response** tab of the **Detail Pane**, describes the variant and explains its purpose.

Section	Description
ID	An ID number is assigned to each variant to simplify search and management.
Description	A brief description of the test.
Difference	Shows which changes were made to the original request for this test. The changes are highlighted in red. (Changes may include changing the value of a parameter, cookie or method; changing a path; deleting a parameter, deleting or adding an HTTP header; removing or adding a parameter to the body.)
Reasoning	Explains why this test result indicates a vulnerability.
CVE ID	The CVE ID of the vulnerability (see "CVE support" on page 184).
CWE ID	The CWE ID of the variant and its parent issue. (See "CWE support" on page 184).

## Taking screen captures

You can take screen captures of your application that demonstrate the vulnerability, to include in reports.

### Procedure

1. Select an item in the **Result List** and toggle through to the required variant.
2. In the **Detail Pane** > **Request/Response** tab, click **Show in Browser**.
3. In the browser, click the **Camera icon**  A screen capture of the current page is added to the data in the Issue Information tab, and will be included in reports.

## Report false positive test results

You can email test information to AppScan Support in order to:

- Report a result that AppScan classified as positive (indicates a vulnerability) but which you believe is negative (does *not* indicate a vulnerability).
- Ask AppScan Support why a result was classified as positive.

You can also use the feature to conveniently zip and email the result to the developers and auditors in your own organization.

**Note:** By default AppScan saves the data in an encrypted format, accessible only to Support personnel. If you are sending the file within your own organization, you must configure AppScan to save the information as a .zip file. In the **Tools** | **Options** | **General tab**, deselect the **Encrypt Attachments** check box.

## Report a single false positive variant

### Procedure

1. Select an item in the **Result List**.
2. In the **Detail Pane** > **Request/Response tab**, browse through the variants of the selected issue.
3. When the variant you want to send is displayed, click **Report False Positive** on the Detail pane toolbar.  
The Report False Positive dialog box opens.
4. Click **Save File** and save the file to disk.  
Depending on your Encryption Setting (see below), the file is saved in encrypted or .zip format.
5. To send the file to AppScan Support, click the **Browse to your support provider** link, log in and upload the file.

## Report a false positive variant set

### About this task

You can report all the variant information of any Issue, URL, or child item in one attachment, using the right-click menu in the Result List.



### Procedure

1. In the **Result List**, right-click on an issue, URL or parameter, and select **Report False Positive**.  
The Report False Positive dialog box opens.
2. Click **Save File** and save the file to disk.  
Depending on your Encryption Setting (see below), the file is saved in encrypted or .zip format.

**Note:** By default the file is encrypted. To change this setting go to **Tools** > **Options** > **General** > **Report False Positive**, and deselect the **Encrypt attachments** check box.

3. To send the file to AppScan Support, click the **Browse to your support provider** link, log in and upload the file.

## Encryption of false positive reports

### About this task

By default the Report False Positive feature saves the data in an encrypted format that only AppScan Support personnel can open.

If you will be sending the attachment within your own organization, the recipient will not be able to decrypt it, so you need to disable the encryption functionality.

### Procedure

1. Click **Tools** > **Options** > **General** > **Report False Positive**.
2. Select/deselect the **Encrypt attachments** check box as required.

## Manual tests

### About this task

The **Manual Test** feature lets you send your own tests and save them as security issues, to be included in your reports.

You can create a manual test as soon as you have Explore results. Manual tests are saved for the current scan only. Note that if you re-scan the test will be lost.

You can base your manual test on an existing test, or you can create a new one from scratch.

### Procedure

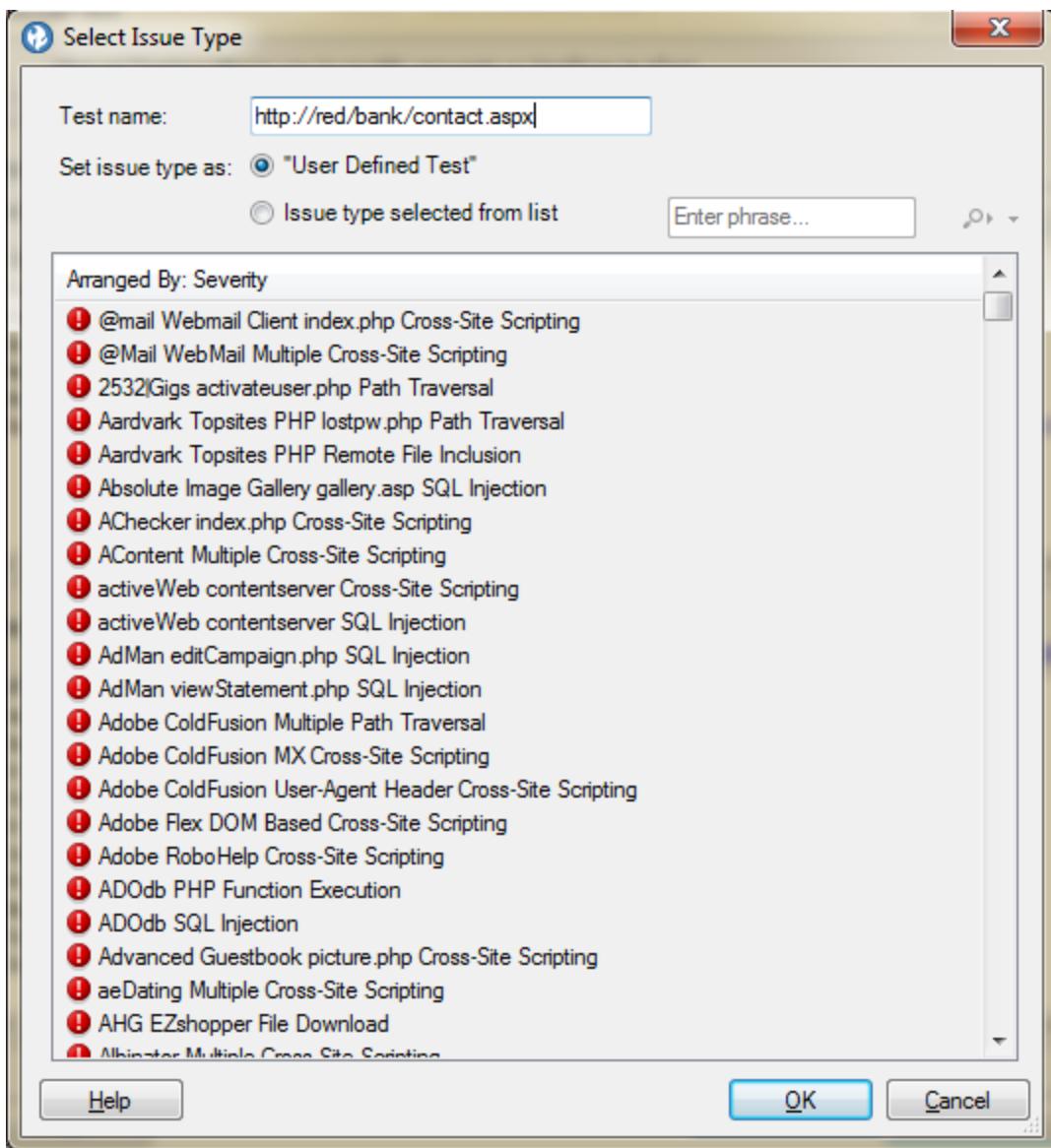
1. To base the manual test on an existing variant:
  - On the **Result List**, click a test variant, or
  - On the **Result List**, click a test, then in the **Detail Pane** use the toolbar to browse to the required variant.

Alternatively, to create a new variant from scratch, simply open the **Manual Test** dialog box (next step) without selecting an existing variant.

2. Open the **Manual Test** dialog-box:
  - On the **Tools** menu, click **Manual Test**, or
  - Right-click on a node in the **Application Tree** or the **Result List**, and select **Manual Test** from the pop-up menu, or
  - Click the **Manual Test** button in **Application Data > Detail Pane** for the selected variant.The **Manual Test** dialog box appears, displaying the properties of the selected test variant.
3. In the **Hostname/IP address** field, enter the server to which the test will be sent.
4. In the **Port** field, enter the port to be used by AppScan to reach the server.  
The default port is **80**; unless SSL is selected, in which case **443** is the default port.
5. If necessary, you can edit the **Request** itself.
6. In the **Options** list, select or clear the following options:

Option	Click to
SSL	Send request in SSL.
Login before sending request	Send login request to the application before sending the manual test.
Auto content length	Automatically update the <b>Content-Length</b> HTTP header in the request, to the value that equals the request content as you edit the request.  If this option is selected, the user cannot edit the content length value.  If the request header does not have the <b>Content-Length</b> parameter, this option has no effect.

7. Click **Send**.  
The request is sent, and the response is displayed in the **Response** text area (lower pane).
8. To view the response in the embedded browser, click **Show in Browser**.
9. To add this manual test to the current scan, click **Save**.  
The **Select Issue Type** dialog box appears, with the **User-Defined Test** radio button selected by default.



10. In the **Test name** field, you can leave the default name (the URL) or define a new name.
11. To save the test under an existing test type (rather than the default User-Defined test type), select the second radio button, and then click on a test type in the list.

**Note:** To locate an issue in the list using the Search feature, type part of the issue name into the Search field. Click on the magnifying glass icon to move from one search result to the next.

12. Click **OK**.

The dialog boxes closes. The new test is added to the results, and will be included when you continue the current scan (**Scan > Continue > Scan/Test**).

**Note:** If you rescan the test is not included.

## Non-vulnerable variants

During a scan, AppScan sends many thousands of test variants to the site it is testing. The responses to many of these indicate that they do not pose a security threat of any kind, and by default AppScan discards all these "non-vulnerable" results.

- If necessary, you can configure AppScan to save all non-vulnerable variants
- You can also change the status of individual results to "non-vulnerable"

"Saving all non-vulnerable variants"

"Defining variants as non-vulnerable"

"Non-Vulnerable Variants List"

"Deleting variants" on page 192

## Saving all non-vulnerable variants

### About this task

You can configure AppScan to save all "non-vulnerable" test variants, if you want to review them.

#### CAUTION:

Saving non-vulnerable test variant information may slow down AppScan performance and significantly increase the disk space required.

### Procedure

In Scan Configuration > Test Options select the Save Non-Vulnerable Test Variant Information check box.

## Defining variants as non-vulnerable

### About this task

When you define a test variant as *non-vulnerable* it will not be displayed in the scan results or included in reports, but you maintain the option of viewing its details (via the "Non-Vulnerable Variants List") and reinstating it later if necessary.

### Procedure

Do one of the following:

- On the Result List, right-click and select **Set as Non-vulnerable**
- Select the variable in the Result List, and on the Request/Response toolbar, click **Set as Non-vulnerable**.

The variant is removed from the scan result display, and will not be included in reports.

## Non-Vulnerable Variants List

The Non-Vulnerable Variants List lets you view details of non-vulnerable variants that have been saved with the scan results, and reinstate them if necessary.

There are two ways that variants are added to this list:

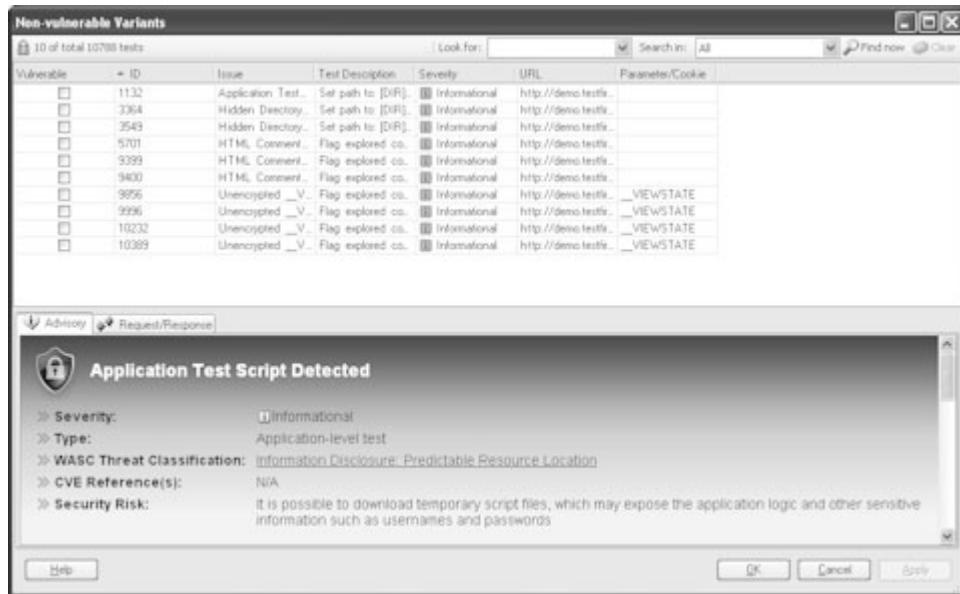
- AppScan has been configured to save non-vulnerable test variant information (see "Test Options view" on page 92)

- You manually changed a result that AppScan categorized as "vulnerable" to "non-vulnerable" (as described in the previous section)

## Viewing non-vulnerable variants

### Procedure

On the View menu, click Non-vulnerable Variants.



## Reinstating non-vulnerable variables as vulnerable

### Procedure

1. Select the check box for one or more variants in the non-vulnerable Variants list, that you want to reinstate as vulnerable.
2. At the bottom of the dialog box, click **Apply**.
3. Click **OK** to confirm.

## Deleting variants

### About this task

When you delete variants (rather than define them as non-vulnerable) they are deleted from the scan results altogether and cannot be reinstated later. (To access them again you would need to run a new scan.)

### Procedure

Do one of the following:

- On the Result List, right-click and select **Delete**.
- Select the variant in the Result List, and on the Request/Response toolbar, click **Delete Variant**.

The variant is removed from the scan results, and will not be included in reports.

---

## Chapter 9. Results: Remediation Tasks

AppScan offers three ways of viewing and working with scan results: Security Issues, Remediation Tasks, and Application Data. This section deals with Remediation Tasks view.

Remediation Tasks view provides solutions designed to address issues that were found in the scan. One remediation task usually addresses a number of security issues.

On the **View Selector**, click:



---

### Remediation Tasks: Application Tree

The **Application Tree** displays the folders and files of the scanned application. Each node in the tree has a counter, which shows how many remediation tasks are held in the node. The count for each node will be the same as, or less than, the count for the **Issues** view, as multiple issues may be addressed by one remediation task.

The **Application Tree** displays remediation tasks in the following levels:

- **Task name**
- **URL**
- **Parameter or Cookie**

A single task designed for an issue found on several URLs is listed once, with the URLs under it.

Select a node in the **Application Tree** to filter the **Result List**, which will display only the results of the selected node.

---

### Remediation Tasks: Result List

The **Result List** displays the Remediation tasks relevant to the node selected in the **Application Tree**. If you selected the **My Application** node, the **Result List** shows all remediation tasks that are relevant to your application.

Remediation tasks are consolidated by the type of remediation that can be done to address issues. Each remediation item has an icon, indicating priority for the task to be done; and a count, indicating how many files, parameters, or cookies will be affected by this remediation.

-  Ensure that accessed files reside in the virtual path and have certain extensions; remove special characters from user input. (1)
-  **Filter out hazardous characters from user input.** (3)
-  Disable HTTP TRACE support in your web server. (1)
-  Encrypt all login requests. (1)
-  Modify the server configuration to deny directory listing, and install the latest security patches available. (6)
-  Cancel output of debugging error messages and exceptions and contact vendor for a security patch. (1)
-  Contact the vendor of your product to see if a patch or a fix has been made available recently. (1)
-  Disable WebDAV, or disallow unneeded HTTP methods. (5)
-  Encrypt sensitive information sent to the server. (1)
-  Install patch Q296576. (1)
-  Remove sensitive information from HTML comments. (2)
-  Verify that parameter values are in their expected ranges and types. Do not output debugging error messages and exceptions. (2)

Each task may hold URLs, which may hold files, parameters, and cookies. You can change the way the remediations are sorted, and you can manipulate their priority value.

## Searching Remediation Tasks in Result List

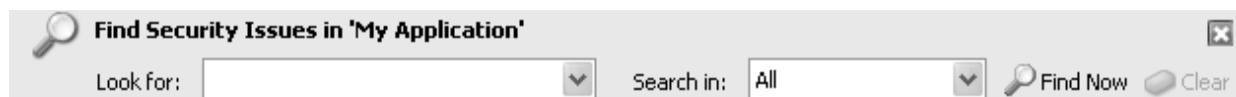
### About this task

You can filter the **Result List** for types of remediation tasks, or you can search for a specific remediation.

### Procedure

1. On the Edit menu, click **Find** (or press **Ctrl + F**).

The **Find** bar appears in the main window, above the **Result List**.



2. On the **Application Tree**, select a node.

- If you select the **My Application** node, Find searches through all the results.
- If you select a node within the tree, Find searches through the selected node and its sub-nodes.

3. In the **Look for** text box of the **Find** bar, enter a string or partial string.

4. In the **Search in** list, select the part of the remediation that you want to search in:

Search in	To find string within
Remediation	remediation name as it appears in the Result List
URL	pathname of URLs for which remediation results are relevant
Details	remediation task details
All	all of the above options

5. Click **Find Now** or press **Enter**.

Results appear in the **Result List**, overwriting the list that was previously displayed. If you enter another **Find** string and click **Find Now** again, the new search is done on the selected node in the **Application Tree**, not on the displayed results of the previous search.

### Example

To find	Look for	Search in
Remediation tasks that concern virtual directories:	"virtual direc"	Remediation
Remediation tasks for login pages:	"login"	URL

## Sorting Remediation Tasks

### About this task

You can re-arrange remediation tasks in the **Result List**. By default, tasks are sorted by priority.

### Procedure

1. Click the **Arranged By** column header.  
A menu appears.
2. Click a command on the menu to sort the remediations:
  - **Priority** - task names are listed by priority order: High Priority first, down to Low.
  - **Count** - number of affected URLs, parameters, or cookies. Tasks that solve the most issues are listed first.
  - **Name** - alphabetical order by task name.

The remediation tasks are re-sorted in the **Result List**.

To reverse the order of the sort (ascending or descending), click the second column of the **Result List**. Click the second column again to revert to the previous order.

## Manipulating priority levels

### About this task

The icons of the remediation tasks indicate their priority levels. The following table explains the icons.

Icon	Indicates
	High priority task
	Medium priority task
	Low priority task

Priority is assigned to remediation tasks based on the issue severity. If the severity is High, the priority is High; Medium severity is Medium priority; and Low and Informational severity is Low priority.

You can view default priority settings as a starting point. You can change the priority assigned to remediation tasks. The new priority setting affects all remediations with the same Task name.

### Procedure

On the **Result List**, right-click a remediation task and select **Priority > High/Medium/Low**. The remediation task icon changes to match the new priority.

## Deleting Remediation Tasks from the Result List

### About this task

You can delete a task from the **Result List**, which will delete the selected node and all the objects contained within it.

### Procedure

1. On the **Result List**, right-click a node.
2. On the menu that appears, click **Delete**.

A message appears, asking you to confirm the delete and reminding you that the data of the tests will be permanently deleted.

3. Click **Yes**.

The remediation task is deleted, but this does not affect the issues; issues will still be available from the Issues view.

---

## Remediation Tasks: Detail Pane

The **Detail Pane** for the Remediations view has one tab. It shows the remediation task currently selected in the **Result List**.

The information in the **Detail Pane** includes: Task Name, Issues (list of scan results addressed by this task), and Details (one or more possible solutions).

---

# Chapter 10. Reports

This section describes how to generate reports from scan results.

## Report overview

After AppScan has assessed your site's vulnerability, you can generate customized reports configured for the various personnel in your organization; from developers, internal auditors, and penetration testers, to managers and executives.

There are five basic types of report, as described below. The Security Report includes many options that can be included or excluded depending on who the report is intended for.

You can open and view the reports from within AppScan, and you can save a report as a file to be opened with a third-party application, such as Acrobat Reader.

Icon	Name	Short Description
	"Security reports" on page 204	Report of security issues found during the scan. Security information may be very extensive, and can be filtered depending on your requirements. Six standard templates are included, but each can easily be tailored to include or exclude categories of information, as necessary.
	"Industry standard reports" on page 206	Report of the compliance (or non-compliance) of your application with a selected industry committee, or your own custom standards checklist.
	"Regulatory compliance reports" on page 208	Report of the compliance (or non-compliance) of your application with a large choice of regulations or legal standards, or with your own custom Regulatory Compliance template).
	"Delta analysis reports" on page 209	The Delta Analysis report compares two sets of scan results and shows the difference in URLs and/or security issues discovered.
	"Template-based reports" on page 211	Custom report containing user-defined data and user-defined document formatting, in Microsoft Word .doc format.

## Configuring report layout

The layout tab of the Create Report dialog box enables you to customize the appearance of your reports. (This feature is optional, as you can simply generate reports using the default layouts.)

### Procedure

1. In the **Create Report** dialog box, click the **Layout** tab.
2. Select the layout options that you want and enter appropriate values:

Layout Options	Description
Include Cover Page	Add a cover page to your report. If selected, cover page options are enabled.

Layout Options	Description
Company Logo	Include your company logo at the top left of the cover page. (Click on the Plus  icon in the Company Logo area, and browse to the logo file on your computer). The AppScan logo is the default logo.
Additional Logo	Include an additional logo at the top right of the cover page. (Click on the Plus  icon in the Company Logo area, and browse to the logo file on your computer).
Report Type	Include the Report Type (editable text) on the lower half of the cover page.
Report Title	Include the default title, or a title that you type in, as the main title in the center of the cover page.
Description	Include the default description, or a description that you type in, as the description on the cover page.
Report Date	Include the date in the footer to each inside page of the report.
Header/Footer	Add a header and/or footer to each inside page. Type the text you want to appear.
Table of Contents	Include a table of contents in the report.
Save as Default Layout	Save the layout settings and texts for future use.

## Viewing and saving reports

Reports can be generated and viewed in AppScan's Report Viewer, and saved in various formats.

### Procedure

1. Configure the report type, template, and filters as needed.
2. To see a preview in the Report Viewer, click **Preview**.
3. To save the report, click **Save Report**, type in a name for the report, and select a Format: PDF (Adobe Acrobat Reader), HTML (web browser), RTF (Microsoft Word), or TXT (text editor).

**Note:** If you have previously saved a report in one format and now want to save it in another, you must save it with a different filename. For example: If you have previously saved a report named Report458.pdf and now want to save it in RTF format, you cannot save it as Report458.rtf but could save it as Report\_458.rtf

## Creating partial reports

You can create a Security Report or a Template-Based Report for a subset of the scan results by right-clicking on the URL or folder for which you want to create the report.

### Procedure

1. In the Application Tree, right-click on the URL or folder for which you want to create the report, then select:
  - **Report for this node > Security**
  - **Report for this node > Template-based**
 The Create Report dialog box opens with report options for the selected node.
2. Continue as for a regular report.
  - “Security reports” on page 204
  - “Template-based reports” on page 211

# Creating user-defined report templates

## About this task

You can create user-defined templates for either the Industry Standards or Regulatory Compliance reports. AppScan report templates have the **.asreg** file extension. Supplied templates are stored in the **\Regulations** folder of your AppScan installation directory; templates you create should be stored in your AppScan User Files folder.

You can create a new template from scratch and save it with the **.asreg** extension, or copy an existing file and make changes as required. (The procedure below describes creating a template based on an existing one.)

## Procedure

1. Open the **\AppScan\Regulations** folder and copy an existing **.asreg** file.

**Note:** By default the AppScan directory is located in **C:\Program Files\IBM\AppScan Standard** unless you specified a different location during installation.

2. Paste the file into your AppScan User Files folder, and give it a new name.

**Note:** By default the AppScan User Files folder is **\My Documents\AppScan** unless you specified a different location in **Tools > Options > Preferences tab > File Locations > User Files folder**.

3. The root tag is **Regulation**, with the attribute of **format\_version**:

```
<Regulation format_version="2.0">
```

4. The next tag should be the title of your template:

```
<Title>Our Organization's web Application Requirement Compliance Report  
</Title>
```

5. Enter a description of the regulation or standard using the **Description** tag:

```
<Description>  
  <Subtitle>Sub Section</Subtitle>  
  
  <p>This regulation addresses ...</p>  
  <p>It is important because...</p>  
  
  <Subtitle>Sub Section 2</Subtitle>  
  
  <p>This section of the regulation addresses ...</p>  
</Description>
```

6. By default, there is a **<Disclaimer>** tag, which ensures that you do not take legal responsibility for the contents of the report.

7. Create the requirement sections (**<Section>** tag) for your regulation template and define which AppScan issues are relevant for each section, by using the **<Cause>** and **<Risk>** tags.
  - The sections are of your own definitions, of what you decide your requirements should be. The **name** attribute of the **Section** tag defines the section.
  - The causes inside each section are taken from the "List of causes" on page 200. Each cause describes an incomplete or incorrect configuration, missing validation, or similar status.
  - The risks inside each section are taken from the "List of risks" on page 201. Each risk is a "worst case scenario".
  - The threat classes inside each section are taken from the "List of threat classes" on page 202. A threat class is a category of tests.

For example:

```

<Section name="My Application login must be secured">
  <Cause>inputLengthNotChecked</Cause>
  <Risk>denialOfService</Risk>
  <Risk>siteDefacement</Risk>
</Section>

```

8. Close the file with the Regulation end tag: </Regulation>

## List of causes

Cause	Description
hazardousCharactersNotSanitized	Sanitation of hazardous characters was not performed correctly on user input.
formatStringsVulnerability	User input is used directly as a formatting string input for C/C++'s sprintf and similar functions.
hiddenParameterUsed	Parameter values were 'hardcoded' in the HTML as a parameter of type 'hidden'.
boundsCheckingOnParamValues	Proper bounds checking were not performed on incoming parameter values.
incorrectDataType	No validation was done to ensure user input matches expected data type.
inputLengthNotChecked	User input length is not limited, thereby enabling buffer overflows.
errorMessagesReturned	Exceptions and error messages, which may contain sensitive debugging information, are presented to users.
debugInfoInHtmlSource	Debugging information was left by the programmer in web pages.
backDoorLeftBehind	A backdoor or a debugging option was left behind by programmers.
clientSideValidation	User input validation is done at the client-side and may be bypassed.
usOfClientSideLogic	The web application uses client-side logic to create web pages.
cookiesCreatedAtClientSide	Cookies are created at the client-side.
javaScriptPassWordMechanism	The web application uses a client-side password authentication.
sqlBuiltByJavaScript	The web application uses client-side logic to create SQL queries.
dotDotNotSanitized	User input is not checked for the '..' string.
weakTokenUsed	A weak token algorithm is used by the web application.
missingPatchesForThirdPartyProds	Latest patches or hotfixes for 3rd party products were not installed.
tempFilesLeftBehind	Temporary files were left in production environment.
improperFileDirPermissions	Improper permissions/ACLs were set to file/directory.
nimdaWormBackdoor	The Nimda worm was found on the system.
sampleScriptsFound	Default sample scripts or directories were installed on the website.
insecureThirdPartySoftware	A vulnerable third party software, which does not have a known patch, is installed on the website.
directoryBrowsingEnabled	Directory browsing is enabled.
managementConsoleAccess	web management console is accessible from the web.
insecureWebServerConfiguration	The web server or application server is configured in an insecure way.
frontPageServerUnsecureInstall	FrontPage server extensions were installed with improper security settings.
insecureWebAppConfiguration	Insecure web application programming or configuration.
vulnSOAPserializer	The SOAP serializer used by your web services server does not validate SOAP input properly.

Cause	Description
sensitiveDataNotSSL	Sensitive input fields such as usernames, passwords, and credit card numbers are passed unencrypted.
nonSecureCookiesSentOverSSL	The web application sends non-secure cookies over SSL.
sessionCookieNotRAM	The web application stores sensitive session information in a permanent cookie (on disk).
redirectionFromWithinSite	The web application performs a redirection to an external site.

## List of risks

Risk Name	Description
tempScriptDownload	It is possible to download temporary script files, which may expose the application logic and other sensitive information such as usernames and passwords.
sourceCodeDisclosure	It is possible to retrieve the source code of server-side scripts, which may expose the application logic and other sensitive information such as usernames and passwords.
pathDisclosure	It is possible to retrieve the absolute path of the web server installation, which may help an attacker to develop further attacks and to gain information about the file system structure of the web application.
directoryListing	It is possible to view and download the contents of certain web application virtual directories, which may contain restricted files.
envVariablesExposure	It is possible to expose server environment variables, which may help an attacker to develop further attacks against the web application.
anyFileDownload	It is possible to view the contents of any file (e.g. databases, user information or configuration files) on the web server (under the permission restrictions of the web server user).
userImpersonation	It is possible to steal customer session and cookies, which may be used to impersonate a legitimate user, allowing the hacker to view or alter user records, and to perform transactions as that user.
remoteCommand Execution	It is possible to execute remote commands on the web server. This usually means complete compromise of the server and its contents.
cacheFilesDownload	It is possible to view the contents of cache files, which may contain sensitive information regarding the web application.
debugErrorInformation	It is possible to gather sensitive debugging information.
eShoplifting	It is possible to steal goods or services (eShoplifting).
denialOfService	It is possible to prevent the web application from serving other users (denial of service).
privilegeEscalation	It is possible to escalate user privileges and gain administrative permissions over the web application.
genericWorstCase	It is possible to undermine application logic.
configurationFile Downloadable	It is possible to download or view the contents of a configuration file, which may contain vital information such as usernames and passwords.
sensitiveInformation	It is possible to gather sensitive information about the web application such as usernames, passwords, machine name and/or sensitive file locations.
genericWorstCase JavaScript	It is possible to exploit JavaScript; the extent of the risk depends on the context of the page modified at the client side.

Risk Name	Description
genericWorstCase JSCookie	It is possible to exploit JSCookie code; the extent of the risk depends on the context and role of the cookies that are created at the client side.
emailSpoofing	It is possible to send emails through your web application, using spoofed email addresses.
siteDefacement	It is possible to upload, modify or delete web pages, scripts and files on the web server.
databaseManipulations	It is possible to view, modify or delete database entries and tables (SQL Injection).
authBypass	It is possible to bypass the web application's authentication mechanism.
siteStructureRevealed	It is possible to retrieve information about the site's file system structure, which may help the attacker to map the website.
publisherInformation Revealed	It is possible to retrieve sensitive FrontPage publishing information.
dataResource Download	It is possible to access information stored in a sensitive data resource.
sensitiveNotOverSSL	It is possible to steal sensitive data such as credit card numbers, social security numbers etc. that are sent unencrypted.
loginNotOverSSL	It is possible to steal user login information such as usernames and password that are sent unencrypted.
unsecureCookieInSSL	It is possible to steal user and session information (cookies) that was sent during an encrypted session.
sessionCookieNot RAM	It is possible to steal session information (cookies) that was kept on disk as permanent cookies.
phishing	It is possible to persuade a naive user to supply sensitive information such as username, password, credit card number, social security number etc.

## List of threat classes

Threat class	Description
Authentication: Brute Force	An automated process of trial and error used to guess a person's username, password, credit-card number or cryptographic key.
Authentication: Insufficient Authentication	Occurs when a website permits an attacker to access sensitive content or functionality without having to properly authenticate.
Authentication: Weak Password Recovery Validation	Occurs when a website permits an attacker to illegally obtain, change or recover another user's password.
Authorization: Credential/Session Prediction	A method of hijacking or impersonating a website user, by deducing or guessing the unique value that identifies a particular session or user.
Authorization: Insufficient Authorization	When a website permits access to sensitive content or functionality that should require increased access control restrictions.
Authorization: Insufficient Session Expiration	When a website permits an attacker to reuse old session credentials or session IDs for authorization.
Authorization: Session Fixation	An attack technique that forces a user's session ID to an explicit value. After a user's session ID has been fixed, the attacker will wait for them to login. Once the user does so, the attacker uses the predefined session ID value to assume their online identity.

Threat class	Description
Client-side Attacks: Content Spoofing	An attack technique used to trick a user into believing that certain content appearing on a website is legitimate and not from an external source.
Client-side Attacks: Cross-site Scripting	An attack technique that forces a website to echo attacker-supplied executable code, which loads in a user's browser.
Command Execution: Buffer Overflow	Attacks that alter the flow of an application by overwriting parts of memory with data that exceeds the allocated size of the buffer.
Command Execution: Format String Attack	Attacks that alter the flow of an application by using string formatting library features to access other memory space.
Command Execution: LDAP Injection	An attack technique used to exploit websites that construct LDAP statements from user-supplied input.
Command Execution: OS Commanding	An attack technique used to exploit websites by executing Operating System commands through manipulation of application input.
Command Execution: SQL Injection	An attack technique used to exploit websites that construct SQL statements from user-supplied input.
Command Execution: SSI Injection	A server-side exploit technique that allows an attacker to send code into a web application, which will later be executed locally by the web server.
Command Execution: XPath Injection	An attack technique used to exploit websites that construct XPath queries from user-supplied input.
Information Disclosure: Directory Indexing	Automatic directory listing/indexing is a web server function that lists all of the files within a requested directory if the normal base file (index.html/home.html/default.htm) is not present. Unintended directory listings may be possible due to software vulnerabilities combined with a specific web request.
Information Disclosure: Information Leakage	When a website reveals sensitive data, such as developer comments or error messages, which may aid an attacker in exploiting the system.
Information Disclosure: Path Traversal	This is a technique that forces access to files, directories, and commands that potentially reside outside the web document root directory.
Information Disclosure: Predictable Resource Location	An attack technique used to uncover hidden website content and functionality, by making educated guesses.
Logical Attacks: Abuse of Functionality	An attack technique that uses a website's own features and functionality to consume, defraud, or circumvents access controls mechanisms.
Logical Attacks: Denial of Service	An attack technique with the intent of preventing a website from serving normal user activity.
Logical Attacks: Insufficient Anti-automation	When a website permits an attacker to automate a process that should only be performed manually.
Logical Attacks: Insufficient Process Validation	When a website permits an attacker to bypass or circumvent the intended flow control of an application.
Application Privacy Tests	n/a
Application Quality Tests	n/a
User-Defined Tests	n/a

## Earlier versions of report templates

Earlier versions of some Industry Standard and Regulatory Compliance templates are saved in the "Old Versions" folder.

## About this task

The Reports dialog box lists the latest versions of the report templates. However, in the case of some Industry Standard and Regulatory Compliance reports, earlier versions are saved in a special folder. The folder is located in \Regulations\Old Versions in your AppScan Program folder.

## Procedure

1. In the Report dialog box, select **Industry Standard** or **Regulatory Compliance**.
2. In the Report Type tab, select **User Defined**.
3. Click **Browse...**, and go to \Regulations\Old Versions in your AppScan Program folder.  
The default location is C:\Program Files\IBM\AppScan Standard\Regulations\Old Versions  
For Japanese, the default location is C:\Program Files\IBM\AppScan Standard\ja-JP\Regulations\Old Versions
4. Select the required file, and continue as for regular report creation.

---

## Security reports

### About this task



The Security report provides information about security issues discovered. You can create a security report that covers the whole scan, or for a particular URL or folder in the application tree.

There are various different templates for Security reports. Each template is a set of content topics that are relevant to different audiences within your organization. The topics contain scan results from each of the views (Security Issues, Remediation Tasks, Application Data), formatted for easy printing, readability, and rapid comprehension of what the results mean, why they are relevant, and how to fix them.

### Security Report Options

The table following summarizes the options in the Security Reports dialog box.

Option	Description
Template	Select one of several templates for the report, or define your own, by selecting/clearing check boxes in the right hand pane, as described in the table following. <ul style="list-style-type: none"><li>• <b>Executive Summary:</b> A high level summary with highlights of security risks found in your web application, and statistics of scan results, formatted in tables and charts.</li><li>• <b>Detailed:</b> A thorough report that includes the executive summary, as well as security issues, advisories and fix recommendations, remediation tasks and application data.</li><li>• <b>Remediation Tasks:</b> Remediation tasks: actions designed to address the issues discovered in the scan.</li><li>• <b>Developer:</b> Security Issues, variants, advisories and fix recommendations, without the Executive Summary or Remediation Tasks sections.</li><li>• <b>QA:</b> Security Issues, advisories and fix recommendations, and application data, without detailed variant information, or the Executive Summary or Remediation Tasks sections.</li><li>• <b>Site Inventory:</b> Application data only.</li></ul>
Min. Severity	Select the lowest level of severity for issues to be included in the report.

Option	Description
Test Type	Select which types of test results to include in the report: <b>All</b> , <b>Application</b> , <b>Infrastructure</b> , or <b>Third-Party Web Component</b> tests.
Sort by	Select whether to sort issues by type or URL.
Limit number of variants per issue	You can reduce the length of the report by limiting the number of variants listed per issue, if this level of detail is unlikely to be useful to the recipient of the report.
Add page break after each issue	This setting applies only to PDF output. It can make the report clearer to read.
View when done	If you select this check box, the report will be opened in an appropriate viewer after it is generated. <b>Note:</b> This will only work if you have a program installed that can open the generated report.

After selecting any template as a basis, you can customize the individual report structure by selecting/deselecting the fields of information to be included. If you do this the template name changes to "Custom".

## Security Report Sections

The table following summarizes the standard content of the various Security Reports. In all cases the actual content can be changed as required by selecting/clearing check boxes in the Report Content pane.

**Note:** A full detailed report could be hundreds of pages long, so be sure to include only the sections that are relevant to your audience.

Report Section	Description
Introduction	A short section that provides some general information about the scan, including such details as overall number of issues found (High, Medium, Low and Informational), and login settings. This section is included in all reports.
Executive Summary	A series of tables summarizing the following information about the scan, or the part of the scan included in the report: <ul style="list-style-type: none"> <li>• Issue types (includes number of issues found for each type, and their severity)</li> <li>• Vulnerable URLs (includes number and type of issues per URL)</li> <li>• Fix recommendations</li> <li>• Security risks</li> <li>• Causes</li> <li>• WASC threat classification</li> </ul>
Security Issues	Issues found in your application: <ul style="list-style-type: none"> <li>• Basic: If you select neither of the following two check boxes, basic information only is included</li> <li>• Additional: Includes more detailed information, including screen captures, similar to the Issue Information tab content</li> <li>• Variants: Includes specific variant information: <ul style="list-style-type: none"> <li>– Request/Response</li> <li>– Difference: The difference between the original request and the test request, as shown in the Detail pane &gt; Request/Response tab</li> </ul> </li> </ul>
Advisories and Fix Recommendations	Technical explanations of the issues found and recommendations for fixing them. <b>Note:</b> To include fix recommendations specific to .NET, Java EE and PHP environments, go to Tools > Options > Preferences and select the required options.
Remediation Tasks	Suggested tasks for improving site security based on the issues found. One task may solve more than one issues.

Report Section	Description
Application Data	List of data that AppScan found in your web application: Application URLs, Script Parameters, Broken Links, Comments, JavaScripts, Cookies, and Filtered URLs.

## Procedure

1. Select the scan content on which to base the report:
  - To create a report for the whole scan, click **Tools > Report > Security Report**
  - To create a report for a particular URL or folder that was included in the scan, right-click on the node in the application tree, and then select **Report for this node > Security**
2. Select the relevant template, or define your own report content by selecting/clearing check boxes in the right pane.
3. Select the options required.
4. To customize the layout of the report, click the Layout tab. See “Configuring report layout” on page 197 for details.
5. Select the output format required: PDF, HTML, TXT, RTF, or XML.
6. Click **Save Report**.

## Limiting the size of Security reports

Tips for reducing a large Security Report to a more manageable size.

### About this task

Security reports can be very large. If, when you attempt to generate a Security Report, you receive a warning message that the file will be hundreds of pages long, or if the report creation process times out, you can try the following tips to reduce report size while including the information that is important to you.

## Procedure

1. One way of reducing report size is to limit the report to a part of the application. You can do this by selecting the relevant node in the application tree, right-clicking it, and selecting **Report for this node > Security**. This will create a report for all parts of the application under the selected node.
2. If you see that your scan results contain thousands of issues, consider generating an **Executive Summary** only, or the **Default Report** with the **Additional Issue Information** check box cleared.
3. By default *all* Test Types are included in reports (Min. Severity = Informational). Try increasing the **Min. Severity** setting so that only High, or High and Medium, severity issues are included.
4. Verify that the **Max. Variants** setting is 1, so that no more than a single variant is included for each issue.
5. Be aware that including **Variants > Request/Response**, and **Advisories and Fix Recommendations** can add a lot of volume to reports. Select these options only if needed.

---

## Industry standard reports

### About this task



The Industry Standards report lets you know if your application complies with standards of a selected industry committee. The list of templates to choose from is updated by IBM as new standards are created for different industries and your AppScan is updated automatically on a regular basis.

If you do not find the industry standard that you need in the list, you can create your own Industry Standard report template (see “Creating user-defined report templates” on page 199).

The Industry Standard reports include the following sections:

Section Title	Information Presented
Description	Explanation of the standard.
Compliance Summary	<p>List and count of non-compliance issues.</p> <p>Note that a single issue may indicate non-compliance of more than one section, and therefore the number of issues per section, for all the sections, may be more than the number of unique issues.</p>
Unique Compliance Issues	<p>List of non-compliant URLs, relevant parameters or cookies, and test names.</p> <p>Each issue appears once only.</p>
Compliance Issues by Section	Detailed explanations of how your application is non-compliant, with remediation to address issues.

The figure below shows a sample of part of an Industry Standard Report.

## Compliance Issues by Section

### 1) Unvalidated input (A1)

3 Issues

#### Cross-Site Scripting

##### Security Risks

- It is possible to steal customer session and cookies, which may be used to impersonate a legitimate user, allowing the hacker to view or alter user records, and to perform transactions as that user

##### Causes:

- Sanitation of hazardous characters was not performed correctly on user input

##### Remediation:

Filter out hazardous characters from user input

##### Issues:

Issue ID	URL	Parameter/Cookie
1	http://bern/bank/search.aspx	searchitems

## Procedure

1. Click Tools > Report > Industry Standard
2. Do one of the following:

- Select one of the listed **Industry Standard Report Templates**.
  - Select the **User Defined** radio button and enter or browse to a custom Industry Standard template file (\*.asreg). See “Creating user-defined report templates” on page 199 for details.
3. If you want to control the appearance of the report, open the **Layout** tab (see “Configuring report layout” on page 197).
4. Click **Preview** to generate and view the report within AppScan; or click **Save Report** to generate and save it to a file.

---

## Regulatory compliance reports

### About this task



The Regulatory Compliance report lets you know if your application complies with regulations or legal standards. There is a long list of templates from various countries to choose from, each template providing a compliance report for a different regulation.

If you do not find the regulation that you need in the list, you can create your own Regulatory Compliance report template (see “Creating user-defined report templates” on page 199 for details).

The Regulatory Compliance reports include the following sections:

Section Title	Information Presented
Description	Explanation of the regulation.
Compliance Summary	List and count of non-compliance issues.  Note that a single issue may indicate non-compliance of more than one section, and therefore the number of issues per section, for all the sections, may be more than the number of unique issues.
Unique Compliance Issues	List of non-compliant URLs, relevant parameters or cookies, and test names.  Each issue appears once only.
Compliance Issues by Section	Detailed explanations of how your application is non-compliant, with remediation to address issues.

The figure below shows a sample from a Regulatory Compliance Report.

# Compliance Summary

34 unique issues across 44 sections of the regulation:

Section	No. of Issues
1. Implement Internet Protocol (IP) masquerading to prevent your internal address from being translated and revealed on the Internet. (Requirement 1.5)	3
2. Do not use vendor-supplied defaults for system passwords and other security parameters. (Requirement 2)	19
3. Always change the vendor-supplied defaults before you install a system on the network. (Requirement 2.1)	14
4. Develop configuration standards for all system components. Make sure these standards address all known security vulnerabilities and industry best practices. (Requirement 2.2)	15
5. Disable all unnecessary and insecure services and protocols. (Requirement 2.2.2)	14
6. Configure system security parameters to prevent misuse. (Requirement 2.2.3)	14
7. Remove all unnecessary functionality, such as scripts, drivers, features, subsystems, file systems. (Requirement 2.2.4)	15
8. Encrypt all non-console administrative access. Use technologies such as SSH, VDM, or	4

## Procedure

1. Click Tools > Report > Regulatory Compliance
2. Do one of the following:
  - Select one of the listed **Regulatory Compliance Report Templates**:
  - Select the **User Defined** radio button and enter or browse to a custom Regulatory Compliance template file (\*.asreg). See “Creating user-defined report templates” on page 199 for details.
3. If you want to control the appearance of the report, open the **Layout** tab (see “Configuring report layout” on page 197).
4. Click **Preview** to generate and view the report within AppScan; or click **Save Report** to generate and save it to a file.

---

## Delta analysis reports

### About this task



The Delta Analysis report compares two sets of scan results and shows the difference in URLs and/or security issues that were discovered in them. You select Base and Target scans, and AppScan compares the two sets of results to let you review how the security situation has improved or deteriorated between time of the two scans.

You can compare the currently loaded scan with a saved scan, or you can compare two saved scans.

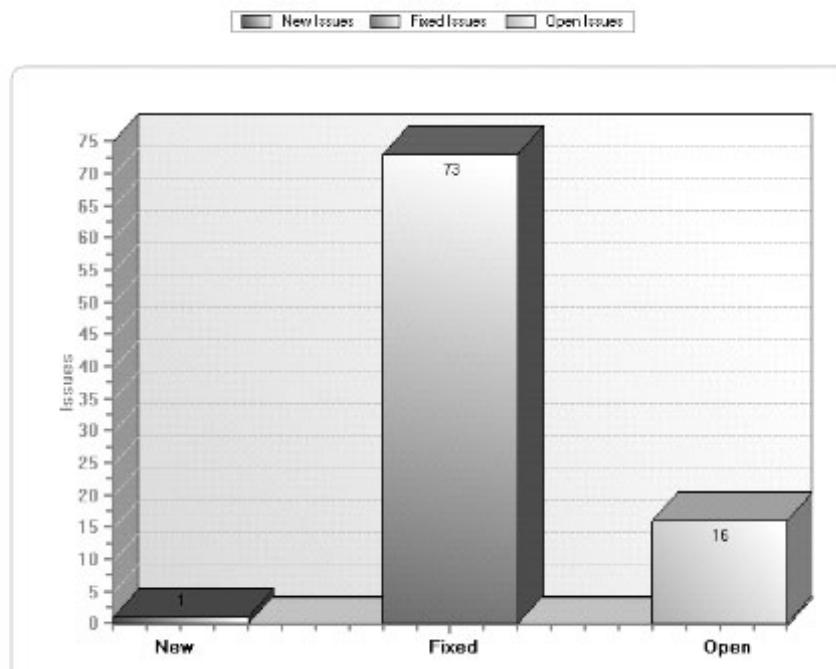
When comparing scans you will generally use the earlier scan as the Base scan. The Delta Analysis report will then indicate how the Target scan results differ from those of the Base scan.

Delta Analysis reports include the following sections:

Section Title	Information Presented
General Information	Lists the names and locations of the Base and Target scans.
Description	Description of the information included in the scan.
Issues per Host	Table showing the number of High, Medium, Low, Informational and Total issues found in each scan.
Application URLs	(If included) Bar chart showing number of New/Removed/Remaining URLs, followed by a full list of each type.
Security Issues	(If included) Bar chart of New/Fixed/Remaining Security Issues, followed by a bar chart showing the distribution of Severities in both scans (High/Medium/Low/Informational), followed by a full listing of all New/Fixed/Remaining issues.

The figure below shows a sample from a Delta Analysis Report.

## Security Issues



## Procedure

1. Click Tools > Report > Delta Analysis

2. Select the Base scan (usually the earlier of the two scans being compared). In the Base Scan area, do one of the following:
  - Click the **Current Scan** radio button
  - Click the **Saved Scan** radio button, and then browse to the location of the saved scan file.
3. Select the Target scan (usually the later of the two scans being compared). In the Target Scan area, do one of the following:
  - Click the **Current Scan** radio button
  - Click the **Saved Scan** radio button, and then browse to the location of the saved scan file.
4. In the Report Content area select one or both of the check boxes for the type of information you want the report to compare:
  - **Application URLs**, and/or
  - **Security Issues**
5. If you want to control the appearance of the report, open the **Layout** tab (see “Configuring report layout” on page 197).
6. Click **Preview** to generate and view the report within AppScan; or click **Save Report** to generate and save it to a file.

## Template-based reports

The Template Based tab of the Create Report dialog box enables you to create reports in Microsoft Word .DOC format, with exactly the data you want, and the document formatting you define.



Several sample templates are included with AppScan. These can be used:

- to generate reports (without editing)
- as a basis for creating your own custom report templates
- as a tool for understanding how to create your own report templates

The dialog box has two panes:

- The left pane lists the templates currently available. Sample templates supplied with AppScan appear in italicized text and are suffixed by the word "(sample)"; any user-created templates appear in regular (non-italic) text.
- The right pane shows an editable "Preview Image" to help you identify the template you require quickly (see table following).

Button or Link	Click to:
Import	Import a Word report template to the list in the left pane (e.g. one that you have previously customized). Imported templates are saved in: <b>My Documents &gt; AppScan &gt; Report Templates &gt; Word</b>  For more details, see “Importing a custom template” on page 216
Export	Export a template on the list to a different location (e.g. for editing).
Remove	Delete a template from the list. This does not delete the template, but moves it to the folder: <b>My Documents &gt; AppScan &gt; Report Templates &gt; Word &gt; Deleted</b>
Edit in MS Word	Open the selected template in MS Word for editing content and layout.  For more details, see “Custom template tutorial” on page 213

<b>Button or Link</b>	<b>Click to:</b>
View Field Reference	Open an HTML list of field references that can be used in creating Word Report Templates.
Select Report Preview Image	Browse-to and select an image that will represent the template in the right pane (when the template is selected in the left pane).
Revert to Default Image	Reinstate the default image (a blank page) for the selected template.
Preview	Generate and display a report on the current scan results, using the template selected in the template pane.  MS Word will open and you can edit the content or layout if required. Note that AppScan creates only a temporary file; if you want to save it you must do so in Word.
Save Report	Generate and save a report on the current scan results (without opening MS Word to display the report contents), using the template selected in the template pane.  AppScan asks you to specify a name and location for saving the file. Once the file has been saved it opens so you can view it.

*See also:*

"Creating a custom report template"

"Custom template tutorial" on page 213

"Importing a custom template" on page 216

"Creating partial reports" on page 198

## Creating a custom report template

You can use field codes for defining your own custom reports to generate in MS Word.

### Introduction to "merge fields"

AppScan Merge fields fall into two categories:

- **Repeaters (or loops):** These consist of a Start and an End tag. They do not produce any data themselves, but they "loop through" any plain fields placed between the Start and End tags, and draw the relevant data for these groups.

Example: <<AS:IssueTypeRepeaterStart>> <<AS:IssueTypeRepeaterEnd>>

This repeater loops through all IssueTypes in the scan results.

- **Plain fields:** These consist of a single field, and they produce the actual data. (Plain fields do not necessarily have to be placed within a repeater, but they cannot have other plain fields as "children".)

Example: <<AS:IssueTypeName>>

This field represents the names of all issues found in the scan.

When you insert a plain field between the Start and End tags of an appropriate loop, a report can be created. Using the two examples above, we can create a report of all the IssueTypes found by the scan:

<<AS:IssueTypeRepeaterStart>> <<AS:IssueTypeName>><<AS:IssueTypeRepeaterEnd>>

To better understand how this works in practice, see the "Custom template tutorial" on page 213 following.

**Note:** A complete list of valid merge fields can be opened by clicking **Tools > Report > Template Based > View Field Reference**.

## Custom template tutorial

This section describes how to make a simple custom template.

### About this task

In this section we will create a simple custom template for generating a table of remediation tasks for all issues discovered during the scan. The template output will have the structure shown below, though obviously the actual data will depend on the scan you use when generating the report.

Index	Name	Count	Priority
1/2	RemediationTaskA	4	High
2/2	RemediationTaskB	2	High
1/3	RemediationTaskC	5	Medium
2/3	RemediationTaskD	2	Medium
3/3	RemediationTaskE	7	Medium
2/2	RemediationTaskF	3	Low

High priority tasks will be listed first, followed by Medium and then Low. For each task the table will indicate the task Index (e.g. 1/n), Name, Count (number of issues that the task applies to), and its priority.

**Note:** Basic familiarity with field codes is assumed in this tutorial. For information about field codes refer to your MS Word documentation.

**Note:** A complete list of valid merge fields can be opened by clicking **Tools > Report > Template Based > View Field Reference**.

### Procedure

1. Open the list of valid merge fields by clicking **Tools > Report > Template based > View Field Reference**. You will need this as a reference from which to copy field names.
2. Open an MS Word document, type in the heading "Remediation Tasks", and save it.
3. **Create the table header:** Create a table with 4 columns and 1 row and type in the headers "Index", "Name", "Count", and "Priority".
4. **Add the fields that will generate the High priority entries in the table:**
  - a. From the Field Reference table, copy the code: AS:RemediationTypeRepeaterStart<Priority=High>
  - b. In the Word document, place your cursor after the header table.
  - c. Paste the merge field into the document *as a field* (MS Word 2003: **Insert > Field > MergeField > Field name field**; MS Word 2010: **Insert > Quick Parts > Field > MergeField > Field name**).

The merge field is created in the document in the form:

```
{MERGEFIELD AS:RemediationTypeRepeaterStart<Priority=High>\* MERGEFORMAT}
```

**Note:** By default Word displays the tags in a truncated form which, since they are long, may omit relevant information. To see the full tag names "show field codes" by clicking [Alt] + [F9] (toggles field codes on/off).

- d. Click **Enter** and create another table with 4 columns and 1 row. This table will later be populated with all the rows of High priority tasks.
- e. After the table add the following merge field: AS:RemediationTypeRepeaterEnd

This appears as:

```
{MERGEFIELD AS:RemediationTypeRepeaterEnd\* MERGEFORMAT}
```

You have now entered the opening and closing merge fields for the section of the document listing High Priority remediation tasks. These two fields form a "loop" that will create a listing based on the fields inserted between them. You can now enter the four fields that will create the *content* in the four columns in the table you added above.

- f. In the left hand column, add the merge field: AS:RemediationTypeRepeaterIndex. This will create a counter (in the form: 1/n, 2/n, etc.) for each of the n tasks in this section.
- g. In the second column, add the merge field for RemediationTypeName.
- h. In the third column, add the merge field for RemediationTypeName.
- i. In the fourth column, type the word "High".

The High priority section of the table is now complete. It will produce multiple rows of data: one for each High priority remediation task in your scan results. Your document should now look like this:

Index	Name	Count	Priority
{ MERGEFIELD AS:RemediationTypeRepeaterStart<Priority=High> \* MERGEFORMAT }			
{ MERGEFIELD AS:RemediationTypeRepeaterIndex \* MERGEFORMAT }	{ MERGEFIELD AS:RemediationTypeName \* MERGEFORMAT }	{ MERGEFIELD AS:RemediationTypeCount \* MERGEFORMAT }	High
{ MERGEFIELD AS:RemediationTypeRepeaterEnd \* MERGEFORMAT }			

5. Repeat Step 4 for Medium and Low priority tasks (in the Priority column, type "Medium" and "Low" respectively). Make sure that each table row (High, Medium and Low) has a Start merge field before it and an End merge field after it, as shown in the image below.

**Note:** The field content of the three rows is the same, so you can copy-and-paste the "High" priority row in between the Start and End tags for Medium and Low priority tasks, simply changing the text in the "Priority" column.

**Note:** Since the Name column will contain text strings, while the Index, Count and Priority columns will contain only numbers or a short word, you may want to adjust the column widths with this in mind.

Index	Name	Count	Priority
{ MERGEFIELD AS:RemediationTypeRepeaterStart<Priority=High> \* MERGEFORMAT }			
{ MERGEFIELD AS:RemediationTypeRepeaterIndex \* MERGEFORMAT }	{ MERGEFIELD AS:RemediationTypeName \* MERGEFORMAT }	{ MERGEFIELD AS:RemediationTypeCount \* MERGEFORMAT }	High
{ MERGEFIELD AS:RemediationTypeRepeaterEnd \* MERGEFORMAT }			
{ MERGEFIELD AS:RemediationTypeRepeaterStart<Priority=Medium> \* MERGEFORMAT }			
{ MERGEFIELD AS:RemediationTypeRepeaterIndex \* MERGEFORMAT }			
{ MERGEFIELD AS:RemediationTypeRepeaterIndex \* MERGEFORMAT }	{ MERGEFIELD AS:RemediationTypeName \* MERGEFORMAT }	{ MERGEFIELD AS:RemediationTypeCount \* MERGEFORMAT }	Medium
{ MERGEFIELD AS:RemediationTypeRepeaterEnd \* MERGEFORMAT }			
{ MERGEFIELD AS:RemediationTypeRepeaterStart<Priority=Low> \* MERGEFORMAT }			
{ MERGEFIELD AS:RemediationTypeRepeaterIndex \* MERGEFORMAT }	{ MERGEFIELD AS:RemediationTypeName \* MERGEFORMAT }	{ MERGEFIELD AS:RemediationTypeCount \* MERGEFORMAT }	Low
{ MERGEFIELD AS:RemediationTypeRepeaterEnd \* MERGEFORMAT }			

6. Add explanatory text and apply any formatting to the rest of the document as required.
7. Save the file.

8. To create a report using this template, see "Importing a custom template" on page 216

## Tag hierarchy

When creating reports using AppScan tags you may want to place one tag "within" another. For example, you may want a separate section for each Issue Type, and within each section a list of Issues, and for each issue a list of Variants. To do this you place the start and end tags for the "child" within the start and end tags for the "parent", as shown in the figure following.

```
<<AS:IssueTypeRepeaterStart<Severity=High>>
<<AS:IssueTypeRepeaterIndex>>

<<AS:IssueRepeaterStart>>
<<AS:IssueRepeaterIndex>>

<<AS:VariantRepeaterStart<Limit=1>>
<<AS:VariantRepeaterIndex>>
<<AS:VariantRepeaterEnd>>

<<AS:IssueRepeaterEnd>>

<<AS:IssueTypeRepeaterEnd>>
```

The following repeater (loop) hierarchies are allowed:

- Issue Type > Issue > Variant
- Remediation Type > Remediation
- Vulnerable URL > Issue Type > Issue > Variant
- Remediation > Issue Type > Issue > Variant

Repeaters should be used either alone, or in one of these hierarchies. Using the repeaters in hierarchies other than these may result in an error when the report is created.

*See also:*

"Repeaters and fields"

## Repeaters and fields

Within each repeater (loop) there must be one or more plain fields, that will be the data input to the loop. (For example, within the Variant loop you could put the VariantDifference field, which will show the difference between the variant and the original for each variant in the loop.) By default, the Index field is added within each repeater, but you can delete it, or add additional fields (as in the tutorial).

Repeaters and Fields Example:

<<AS:IssueTypeRepeaterStart>>	[Start looping through issue types]
<<AS:IssueTypeRepeaterIndex>>	[For each type list index (e.g. 3/24) and name]
<<AS:IssueTypeRepeaterName>>	
<<AS:IssueRepeaterStart>>	[Start looping through individual issues]
<<AS:IssueRepeaterIndex>>	[For each issue list index (e.g. 3/24)]
<<AS:VariantID>>	[For each issue list ID and variants]
<<AS:VariantTestRequest>>	
<<AS:IssueRepeaterEnd>>	[End issue repeater]
<<AS:IssueTypeRepeaterEnd>>	[End issue type repeater]

## Field filters

### About this task

Some of the merge fields include filters. For example, the VariantRepeater limit can be used to set how many variants will be included per Issue.

**Note:** In the VariantScreenShot field, the filter defines the scale ("Scale=50") by percentage (50%). You can change this to any integer value up to 100 (for full size).

### Procedure

1. Select the field.  
`<>AS:VariantRepeaterStart<Limit=1> >>`
2. To see the full field names, click [Alt]+F9 (or right-click on the field and select **Toggle Field Codes** from the menu).  
`{MERGEFIELD AS:VariantRepeaterStart<Limit=1>\*MERGEFORMAT}`
3. Update the filter between the angled brackets.  
`{MERGEFIELD AS:VariantRepeaterStart<Limit=4>\*MERGEFORMAT}`
4. To hide the full field names, click [Alt]+F9 (or right-click on the field and select **Toggle Field Codes** from the menu).  
`<>AS:VariantRepeaterStart<Limit=4> >>`

## Importing a custom template

### About this task

If you have created and saved a custom template (see “Custom template tutorial” on page 213 ) you can add it to the list that appears in the Template Based Reports list.

### Procedure

1. In *Template Based* view of the *Report* dialog-box, click **Import**.
2. Browse to your custom template file and click **Open**.

The template is added to the list in the left pane, and can be used for generating custom reports.

---

## Chapter 11. Tools

This section explains how to use additional tools provided with AppScan.

### Options dialog box

This section describes options you can control, to customize AppScan, from the **Options** dialog box (**Tools > Options**).

Changes made in this dialog box apply to AppScan across sessions and scans. The dialog box has five tabs.

#### Scan Options tab

**Tools > Options > Scan Options tab** provides control over how AppScan operates during a scan.

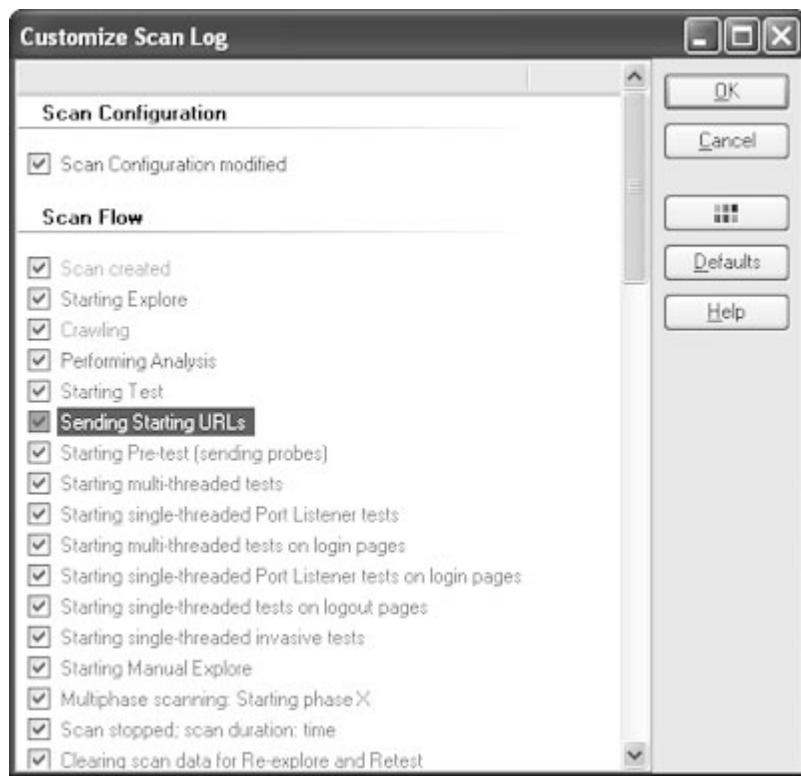
Option	Description
<i>Scan Monitoring:</i>	
Enable Scan Log	You can enable/disable the scan log, and can use the <b>Customize</b> button to choose exactly which items appear in the log, and their color.  See “Customizing the Scan Log.”
Enable request/response logging	You can enable the <b>Request/Response Log</b> , which can help technical support if you need assistance. However, turning on the log may affect performance. It should be disabled unless you need it.
Monitor memory consumption and disk space	AppScan checks available virtual and physical memory and disk space, and will send a warning if the available memory or disk space drops below the recommended amount.  If you clear the <b>Monitor memory consumption and disk space</b> check box (disable memory and disk space monitoring), AppScan does not perform the checks.
Indicate if scan is incomplete	If selected, Scan Incomplete appears at the bottom left corner of the Result List, when a scan is stopped before completion, to remind you that the results are only partial.
Automatically save during scan	Save a file of the scan (.scan) as it runs.  <b>Interval in Minutes:</b> Enter the number of minutes between saves; or use the slide bar to change the interval.
<i>Communications:</i>	
TLS/SSL support	If your application uses secure protocols to communicate with clients, configure AppScan to support the relevant protocols. Options are TLS 1, 1.1, 1.2 and SSL 2. Changing this option requires that you restart AppScan.
HTTP version	Choose the HTTP version of the site: 1.0 or 1.1
Use External Browser	AppScan scans using its own internal browser by default. You can select an external browser to be used instead.

#### Customizing the Scan Log

You can control which items appear in the scan log, and their color.

The “Scan Log” on page 273 lists actions performed by AppScan during the current scan. The **Customize Scan Log** dialog box (**Tools > Options > Scan Options tab > Customize**) contains a scroll-down list showing all items that can be included in the log, a check box indicating whether they are currently

included, and colored text indicating the color they appear in the log.



- To exclude/include an item from the scan log, simply deselect/select the check box next to it.
- To change the color an item appears in the scan log: Select the item, click , select a color from the palate that opens, then click **OK**.
- To restore the default settings, click **Default**.

## Preferences tab

Tools > Options > Preferences tab provides various controls over AppScan files, fix recommendations that appear in the GUI, and the Welcome screen.

Option	Description
Display Welcome screen on startup	The Welcome screen appears when AppScan launches. It offers quick access to configuring and loading scans.  If you find that you are closing the Welcome screen more often than using its features, you can stop it from opening by clearing this check box.
Check for updates on startup	AppScan will check for updated tests, security issues, and fixes when it launches. To disable this check, clear this check box; you can get updates whenever you want: <b>Help &gt; Check for Updates</b> .
Automatically install security updates	If this check box is selected, the updates are installed while you work, without your intervention.
Report False Positive: Encrypt files	This setting applies to “Report false positive test results” on page 187. AppScan will encrypt selected variant information files that have been zipped, before attaching them to a new post in your default e-mail client.  Clear this check box if the email you want to send is to anywhere other than the AppScan support team.

Option	Description
Run Scan Expert before Scan	(Selected by default) When selected, Scan Expert runs automatically whenever you run a full scan (Explore and Test), but not when you select Explore Only or Test Only.
Run Result Expert after Scan	(Selected by default) When selected, Result Expert runs automatically after scans.  Select how the application should behave when a scan is stopped before completion: <b>Prompt</b> (asks if you want to run Result Expert), <b>Run Result Expert automatically</b> , or <b>Don't run</b> .
Include advisory training videos	For some of the Advisories, short training videos are available that can be embedded in the Advisory tab for viewing when you view the advisory. Deselect if you need to reduce file size.
Fix Recommendation Settings	If your application does not use one or more of the environments listed, you do not need to see fix recommendations that are specific to those environments.  Clear the check boxes of any environments (.NET / Java EE / PHP) that are not relevant to your application.

## Recording Proxy tab

Use this tab to configure AppScan to act as a proxy for external browsers, or for manually exploring non-SOAP web services using either a remote device (such as a mobile phone), or a local application (such as a simulator or emulator).

Option	Description
Proxy Port	Specifies which port AppScan uses. When using AppScan as a proxy server you need to configure the external browser or mobile device to use this port.  Use the check box to select whether AppScan selects an available port automatically, or lets you choose the port. Note that if the port is chosen automatically it may change between sessions, and you will therefore need to re-configure your mobile device.
External Connections	This setting determines which connections to external domains are accepted.  <b>Reject all</b> (Default) Connection attempts from all external IPs will be rejected. Use this setting only if you will be exploring using an application on the same machine as AppScan.  <b>Accept white list only</b> Connections from external IPs that appear on the white list will be accepted; all others will be rejected.  <b>Accept white list and prompt for others</b> Connections from external IPs that appear on the white list will be accepted automatically; for all others the AppScan user will be prompted, with the option of adding the new IP to the white list. Note that prompts are seen only if the External Traffic Recorder is open.
White List	Connections from IPs listed here will be accepted automatically.  To add new IPs to the list, click  and select an option: <ul style="list-style-type: none"><li>• To add a single IP to the list, type in the IP and optionally a description. <b>Tip:</b> If you will be using a remote device but are not sure of its IP address, or if it changes frequently, select <b>White List and prompt for others</b>. The first time the device connects with a new IP, a pop-up appears giving you the option to add it to the white list.</li><li>• To add a range of IP addresses, add an IPv4 address and subnet mask, or an IPv6 address and subnet prefix length, and optionally a description.</li></ul>

Option	Description
AppScan SSL Certificate	<p>If the server uses HTTPS, since AppScan has to act as a proxy in order to record the traffic between the web service and the device you use to manually explore, it will be sending SSL certificates to the device instead of the web service's certificate. When a browser receives an unrecognized certificate it typically warns the user with a pop-up, but in the case of a mobile device the request is usually just ignored. It is therefore impossible to explore the application unless the AppScan certificate is accepted on the device sending the requests.</p> <p><b>Add</b> Adds the AppScan SSL certificate to the root certificates on this machine.      You must do this to allow sending requests to the web service. The AppScan certificate will be added to the root certificate, and requests from the web service to the simulator will not be rejected.  <b>Note:</b> After you have added the certificate, the button changes to <b>Remove</b>, and can be used to remove the certificate from the AppScan machine.</p> <p><b>Export</b> Saves the AppScan SSL certificate <i>that is currently installed on this machine</i>, as a ZIP file, so it can be added manually to the root certificates on a different device. Note that you do not usually need to do this, as you can import the certificate directly from the device in most cases.</p> <ol style="list-style-type: none"> <li>1. In AppScan, click <b>Scan &gt; Manual Explore &gt; Using External Device</b>          The External Traffic Recorder opens with status "Waiting for incoming connections".  <b>Important:</b> Leave it open for the next sub-steps.</li> <li>2. On the mobile device, browse to <a href="http://appscan">http://appscan</a></li> <li>3. In AppScan, if you are prompted to allow an incoming connection from your device, click <b>OK</b>.          When the device connects successfully to AppScan as its proxy, a message (on the device) confirms the connection, IP and port. If the certificate is installed on the AppScan machine, it also provides a button to install it on the device.  <b>Note:</b> If the button is grayed out, the certificate is not installed on the AppScan machine.  <b>Note:</b> The device's domain and request will appear in the External Traffic Recorder lists.</li> <li>4. On the mobile device, tap <b>Install AppScan SSL Certificate</b>          The certificate is installed.  <b>Note:</b> If the device is unable to access the application you are testing after this procedure, you need to install the certificate (onto the remote device or application) manually:             <ol style="list-style-type: none"> <li>a. In AppScan, open <b>Tools &gt; Options &gt; Recording Proxy</b></li> <li>b. Click <b>Export</b> and save the certificate as a ZIP file.</li> <li>c. Install the certificate as a root certificate on the device or application.</li> </ol> </li> <li>5. When finished, click <b>Cancel</b> on the External Traffic Recorder, to close it.</li> </ol> <p><b>Note:</b> This option is active only if the certificate is already added to the root certificates on <i>this</i> machine.</p> <p><b>Attention:</b> The AppScan certificate that is exported must be identical to the one installed locally. If you <b>Remove</b> the local certificate and then <b>Add</b> again, you must also reinstall it on the device, as the new certificate is not identical to the previous one.</p>

For more information, see “Using AppScan as recording proxy” on page 121

## General tab

Tools > Options > General tab provides options to control how AppScan prioritizes issues, gets updates, and displays dialog boxes.

Option	Description	
File Locations	During use, AppScan writes various files to disk. You can change the locations to which these files are saved by clicking the browse button to the right of the file type, and browsing a different location.	
	User files	*.scan (Scan), *.exd (Explore Data), *.xml (Report, Export), *.scant (Scan Template), *.asreg (Report Template), *.aspol (Test Policy)  Default location: ...\\My Documents\\AppScan
	Log files	*.log (Log), *.lic (License), *.dmp (Memory Dump), *.css (Packed Support)  Default Location: ...\\Program Files\\IBM\\AppScan Standard\\Logs
	Custom advisories	*.xml (User-created advisory file)  Default Location: ...\\My Documents\\AppScan\\Advisories
	Noise classification file	*.xml  Default Location: ...\\Program Files\\IBM\\AppScan Standard\\IssueManagement  (For details see "Issue state: Open or Noise" on page 175.)  Clicking the  button deletes the Noise Classification file, restoring default noise classifications.
Log File Size	Set the maximum log file size, in MBytes.  When the limit is reached the file is saved as a backup and a new log file is opened. (The next time a backup is saved the earlier backup is deleted.) If you have limited space on your hard drive, you may want to limit the size of the AppScan log file.	
Operating system priority level	By default, AppScan has a <b>Normal</b> priority among running applications. If other applications need more of your computer's resources, you can change the value of AppScan to <b>Idle</b> , to a lower priority, allowing it to run only if other applications do not take the resources.  <b>CAUTION:</b> <b>Changing the operating system priority level can dramatically affect the performance of other applications currently running on the same system.</b>	
Reset Dialog Suppress	If you have selected the "Don't Show this again" check box in any AppScan warning, you can reset them to display again by clicking the Reset Dialog Suppress button.	
Clear History	Click to clear all history lists: URL History, Recent Scans list, Recent Templates list, Recent search strings, and recent user-defined policies.	
Select Language	If AppScan can work on your workstation in more than one language, you can select the interface language here. After changing this setting you may be asked to close and reopen AppScan for the change to take effect.	

## Advanced tab

Tools > Options > Advanced tab lets you view and change the default values of advanced settings.

The Advanced tab lists the Type (String, DWord or Boolean) and current Value of many settings, and lets you change them. When you select a setting by clicking on it, a brief description of the setting and how to use it appears at the bottom of the list.

- To locate a particular Preference Name (or names), type a word or fragment into the filter field at the top of the dialog. Use the drill-down arrow to **Match case** and/or **Match whole word**
- To change a setting, click on the Value field, and select or type in the new value. Settings that have been changed from their default values appear in bold.

**Note:** If you try to enter an invalid value a warning appears and the value is not accepted.

- To restore the default value of a *single* Preference, right-click on the Preference and select **Restore Default**.
- To restore *all* settings to their default values, click the **Restore Defaults** button at the bottom of the dialog.

---

## Scan Scheduler

You can set scans to start automatically on a schedule. Scan schedules may be one time in the future or on a routine basis. When the time of a scheduled scan arrives, AppScan is automatically opened and it runs the scan.

Due to licensing issues, only one instance of AppScan can be open at any time. Therefore, scheduled scans will not run if AppScan is already open. If you are using AppScan when the time of scheduled scan arrives, a message opens to let you know that the scheduled scan will be skipped.

### Schedule a new scan

#### About this task

You can schedule scans using the configuration of the current scan or of a saved scan. The scheduled scans are given the name of the source scan with the date and time appended, and saved in the same folder as the source scan.

#### Procedure

1. On the Tools menu, click **Scan Scheduler**.  
The **Scan Scheduler** dialog box appears, listing any scans scheduled so far.
2. Click **New**.  
The **Schedule Settings** dialog box appears.
3. Enter a name for the schedule.
4. Decide whether you want to schedule the current scan or schedule a loading of a saved scan (\*.scan file).  
If you choose the current scan, and the current scan is unsaved, AppScan asks you to save the scan as a scan file, to be loaded by the scheduler.
5. Select the recurrence schedule: daily, weekly, monthly, or once on a specified date and time.
6. Select the date and time that the first scan will start.
7. You can optionally set a time limit for the scan, by selecting the **Limit Scan Time** check box, and entering the maximum duration of the scan in minutes.
8. Enter your domain name, user name, and password.  
These authentication values enable the task to run as if started by the given user; without this information, the task may not be able to run at all.
9. Click **OK**.  
The schedule name appears in the **Scan Scheduler** dialog box.

## Edit scheduled scan configuration

### Procedure

1. In the Scan Scheduler dialog box (**Tools | Scan Scheduler**), select a scheduled scan.
2. Click **Edit**.  
The **Schedule Settings** dialog box appears.
3. Edit the details as needed.
4. Click **OK**.

## Delete a scheduled scan

### Procedure

1. In the Scan Scheduler dialog box (**Tools > Scan Scheduler**), select a scheduled scan.
2. Click **Delete**.

## Schedule a Test stage only

### About this task

A full scan consists of two stages: Explore and Test. If it is not necessary to run the Explore stage each time, you can configure a scheduled scan to run the Test stage only. This is done using *Scheduled Tasks* on the *Windows Control Panel*.

### Procedure

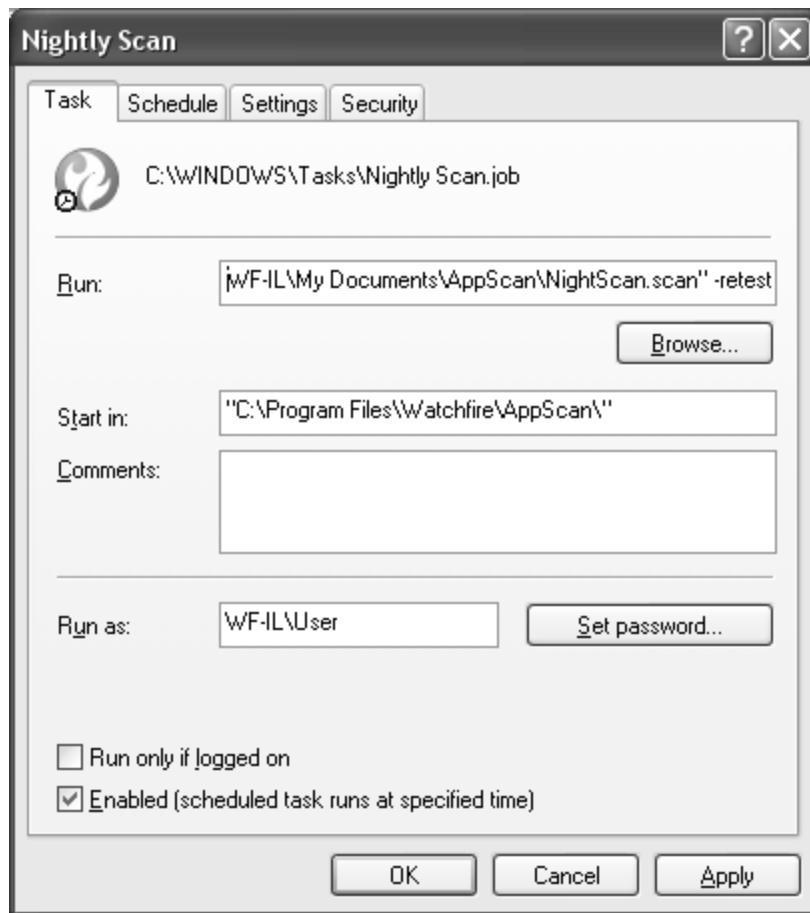
1. In AppScan, “Schedule a new scan” on page 222 (**Tools | Scan Scheduler | New**).
2. On the Windows **Start** menu, click **Control Panel | Scheduled Tasks**.

**Note:** If the *Scheduled Task* option does not appear on your *Windows Control Panel*, you may be in *Category View* which does not display all options. If necessary switch to *Classic View*.

The Windows Scheduled Tasks dialog box opens, with the task that you created in AppScan on the list of tasks.

3. Right-click on the AppScan scheduled scan and select **Properties** from the pop-up menu.

The *Scheduled Task Properties* dialog box opens.



4. In the *Run* field, at the very end of the text string (after the final quotation marks), type in:  
[space][hyphen]retest
  5. Click **OK**.
- When the scheduled time arrives the Test stage only will run.

## Schedule a scan in installments

### About this task

If you have limited "windows of time" in which to run scans (for example, you have just three hours a day, between 1.00 and 4.00 am), and your scans need more than this amount of time to complete, you can configure a scheduled scan to run in installments. In order to do this you need to tell AppScan:

- To save the scan with the original filename (without the timestamp as part of its name)
- To continue the next scheduled scan from where the previous scan left off

This is done by adding command line parameters via *Scheduled Tasks* on the *Windows Control Panel*.

### Procedure

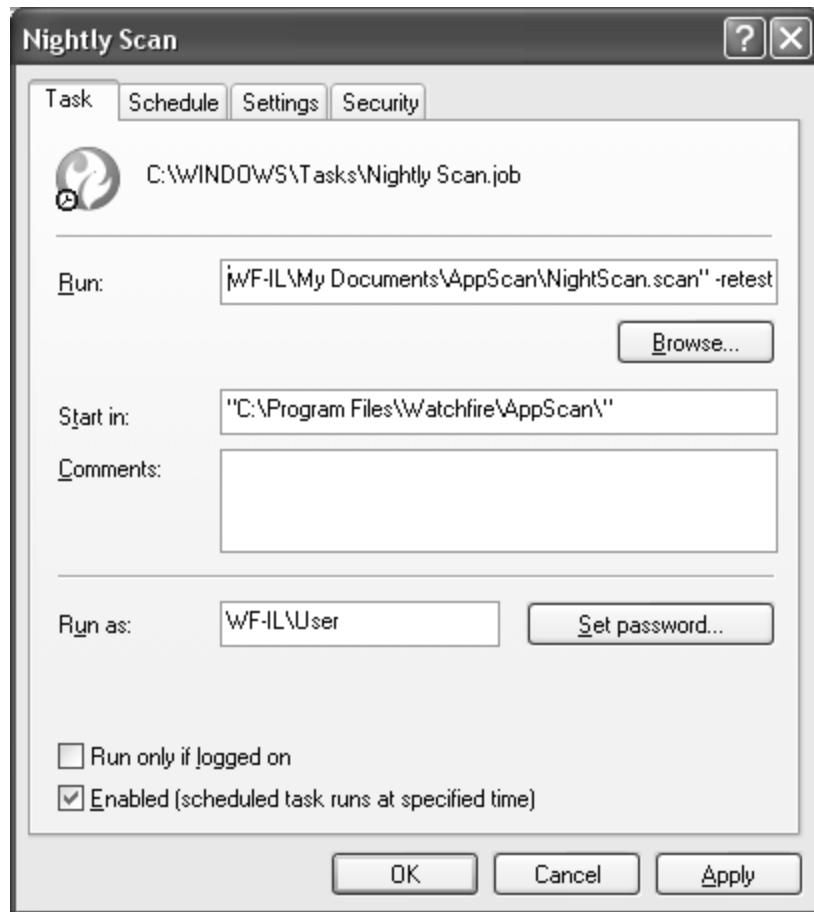
1. In AppScan, "Schedule a new scan" on page 222 (**Tools > Scan Scheduler > New**).
2. In the Windows Start menu **Run** field, type Task Scheduler  
The Windows Scheduled Tasks dialog box opens, with the task that you created in AppScan in the list of tasks.
3. Right-click on the scheduled scan, and select **Properties**.  
The Scheduled Task Properties dialog box opens.

- In the Triggers tab, configure the task as needed. Refer to Microsoft Windows documentation for further information.

## Scheduled task command line parameters

The table below shows the available command line parameters for use in the *Scheduled Tasks* dialog box of the *Windows Control Panel*.

Parameters are added after the main string in the Run field of the Task tab.



Command Line Parameter	Function
[space][number]	This number is the timeout, in minutes, for the scheduled scan.
[space][hyphen]retest	<p>Configures the scheduled scan to start from the Test stage, omitting the Explore stage.</p> <p>This might be useful for re-testing a site at intervals, using a source scan that has already explored the site.</p>

Command Line Parameter	Function
[space][hyphen]continue	<p>Configures the scheduled scan to continue from where it left off rather than starting again.</p> <p>This might be useful for re-testing a site at intervals, using a source scan that has already explored the site.</p> <p>Note that when using this option to scan <i>in installments</i> you must also use the <i>saveName</i> switch, so that the scheduled scan overwrites the original one each time it runs. (See “Schedule a scan in installments” on page 224.)</p>
[space][hyphen]saveName[filename]	Defines the name that the scheduled scan will be saved with. When no <i>saveName</i> is defined, AppScan saves the scheduled scan with the source filename followed by a timestamp.

## User-defined tests

### About this task

AppScan provides a database of thousands of tests. However, if your web application has issues that are specific to it, or if you want to write your own advisories for fixing issues, you can create your own tests. These tests are saved and included in your AppScan database of tests.

Each test looks for one specific issue. For example, one test will modify a path in a request; another test will modify user input to include a character that should be invalid.

### Procedure

Click **Tools > User-Defined Tests**.

The User-Defined Tests dialog box appears, listing tests that have been defined, and their type. The check box next to each test indicates whether or not it is currently included in scans.

Option	Description
Checkbox	Select/clear to include/exclude a test from the scan.
View Test	Select a test <i>row</i> (not its check box) and then click <b>View Test</b> , to open <b>User-Defined Test Details</b> .
Export	Click to export all current user-defined tests, and their selected/cleared status, as a UDT file that can be imported to another scan.
Import	Click to import a previously saved UDT file. The tests in the imported file are added to the current list of tests. <b>Note:</b> If the imported file contains a test with the same name as one of the current tests, a number is appended to its name.
[+]	Click to open the User-Defined Test wizard to create a new test.
-	Click to delete the selected test.

## User-Defined Test Wizard

Wizard for creating user-defined tests.

### About this task

*Step 1 of 5*

The five-step User-Defined Test wizard help you create user-defined tests in addition to the tests AppScan creates automatically.

## Procedure

1. In the User-Defined Tests dialog box, click **Add Test**.  
Step One of the **User-Defined Test** wizard appears.



2. In the **Test Name** text box, enter a name for your test.
3. In the **Test Type** area, select the type of test you want to create, then click **Next** and follow the instructions for the type of test you have chosen.

The next two steps of the wizard depend on which test you select (see table below).

Test	Description
<b>Infrastructure Test</b>	Creates a test that attempts to reach a specific URL during the scan. See "Defining infrastructure tests" on page 228
<b>Parameter Modification</b>	Creates a test that changes the value of a specific parameter in the application to a different value. See "Defining parameter modification tests" on page 229
<b>Parameter Addition</b>	Creates a test that sends an unexpected parameter to the application. See "Defining parameter addition tests" on page 231
<b>Pattern Search</b>	Creates a test that looks for a particular pattern (such as a credit card number format) in all Explore-stage responses from the application. See "Defining pattern search tests" on page 232

See also:

- "Creating your own advisory" on page 233
- "Managing user-defined tests" on page 235

## Defining infrastructure tests

### About this task

*Step 2 of 5*

The Infrastructure Test attempts to reach a URL that you specify (such as a known issue in a third-party product) during the scan.

### Procedure

1. Select **Infrastructure Test** and click **Next** in Step 1 of the wizard.

The Define Test step appears.



2. In the text box, enter the URL to be tested.

3. Click **Next**.

The Severity and Validation step appears.

### What to do next

Next step: "Setting severity and validation of infrastructure tests"

## Setting severity and validation of infrastructure tests

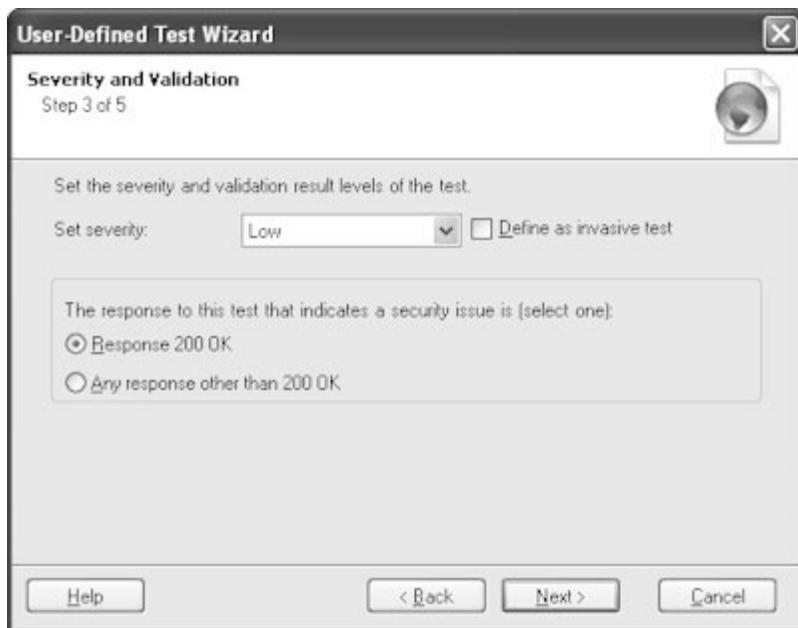
### About this task

*Step 3 of 5*

In this step, you define the severity level (High, Medium, Low, or Informational) of a positive test result, whether or not the test is Invasive, and which type of result AppScan should consider a positive result (successful attack).

- The severity level you assign will be shown in test results.
- Depending on the type of test you have created, a response with a 200 OK status code (valid, expected response) may indicate a successful or an unsuccessful attack. You must define which.

**Note:** If you define the test as Invasive, it will only be sent if Test Policy (see "Test Policy view" on page 87) is configured to include Invasive Tests that could damage the application.



### Procedure

1. On the **Set Severity** list, select the level of severity that you want AppScan to assign the test result, if your application is found to be vulnerable to this test.
2. If running this test could pose a damage risk to your application, select the **Define as Invasive Test** check box.
3. In the **Response 200 OK** area, define whether a response with a 200 OK status code (valid, expected response) is a positive or negative test result. This setting is mandatory.
  - **Response 200 OK:** Select this if the correct response to this test by your application should be an error message or similar response (but not with 200 OK). If the application responds with 200 OK, AppScan will list it as a security issue.
  - **Any response other than 200 OK:** Select this if 200 OK is the correct response to this test. If any other response is received, AppScan will list it as a security issue.
4. Click **Next**.

Continue with "Creating your own advisory" on page 233.

## Defining parameter modification tests

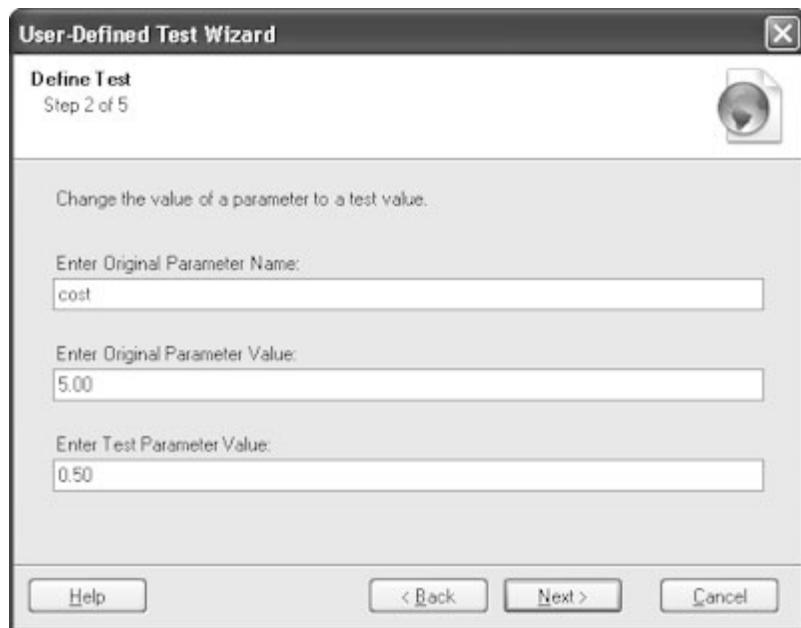
### About this task

#### Step 2 of 5

The Parameter Modification Test changes the value of a specified parameter and evaluates the application's response. The dialog box accept both ordinary parameters and regular expressions as valid input.

### Procedure

1. Select **Parameter Modification** and click **Next** in Step 1 of the wizard.  
The Define Test step appears.



2. Enter the name of the parameter; the original value of the parameter that is to be changed; and the new value that you want to be used for the test.

**Note:** For the original values you can use regular expressions. For example: For **Original Parameter Name** you could enter **name**, to alter name parameters, or **.\*** to alter all parameters. For **Original Parameter Value** you could enter **6** to change that specific value only, or **.\*** to change all number parameters to the Test value.

**Note:** The Expression Test PowerTool (**Tools > Expression Test**) can be useful to verify the syntax of your regular expressions. If you need additional help you may find the following link useful:  
<http://www.regular-expressions.info/quickstart.html>

**Note:** AppScan automatically URL-encodes text entered in this dialog box (for example \$ is automatically converted to %24). Therefore, do not enter URL-encoded text (unless you want it to be "double encoded").

3. Click **Next**.

The Severity and Validation step appears.

## What to do next

Next step: "Setting severity and validation of parameter tests"

### Setting severity and validation of parameter tests

#### About this task

#### Step 3 of 5

The options for severity level and validation are the same for both types of parameter tests. In this step, you decide how AppScan should handle the results of the test: what should be an issue and how high the severity of the issue should be.



## Procedure

1. On the **Set Severity** list, click the level of severity that you want AppScan to mark this test, if your application is vulnerable to this test.
2. Decide whether running this test could pose a damage risk to your application (for example, if the test could crash the application). If so, select the **Define as Invasive Test** check box.
3. In the **Response 200 OK** area, decide whether AppScan should consider this test as an issue if the server responded with a 200 OK status code (valid, expected response).
  - If your application should have responded with an error message or other response (other than OK), select **Yes**.
  - If it is not an issue that your application responded with the 200 OK response, select **No**.You can set whether the response type should open an issue or not; or you can leave the default settings.
4. If you want to determine if AppScan calls the test results an issue, if the test causes the body of the response to be changed, select the **Response Body** check box.
  - If you consider a different response body to be a sign of an issue, select **Yes**.
  - If a change in the response body should not be seen as an issue, select **No**.
5. Click **Next**.

## What to do next

Continue with "Creating your own advisory" on page 233.

## Defining parameter addition tests

### About this task

#### Steps 2 and 3 of 5

The Parameter Addition test adds a specified parameter to requests sent to the site during the scan.

## Procedure

1. Select **Parameter Addition** and click **Next** in Step 1 of the wizard.  
The **Define Test** step appears.



2. Enter the name of the parameter to be added, and give it a value.

**Note:** AppScan automatically URL-encodes text entered in this dialog box (for example \$ is automatically converted to %24). Therefore, do not enter URL-encoded text (unless you want it to be "double encoded").

3. Click **Next**.

The **Severity and Validation** step (Step 3 of 5) appears. It is the same as for Parameter Modification (see "Setting severity and validation of parameter tests" on page 230).

4. Set the severity level and validation options.
5. Click **Next**.

## What to do next

Continue with "Creating your own advisory" on page 233.

## Defining pattern search tests

### About this task

#### Steps 2 and 3 of 5

The Pattern Search Test searches for a particular pattern (such as a credit card number with the format "xxxx-xxxx-xxxx-xxxx") in all Explore-stage responses from the site.

## Procedure

1. Select **Pattern Search** and click **Next** in Step 1 of the wizard.  
The **Define Test** step appears.



2. Enter the pattern to be detected, in the form of a regular expression.

**Note:** A regular expression (regexp) is a string that describes a set of strings, according to certain syntax rules. IBM Security's Expression Test PowerTool Tools > **Expression Test**) can be useful to verify the syntax of your regexps. If you need additional help writing regexps you may find the following link useful: <http://www.regular-expressions.info/quickstart.html>

3. Click **Next**.

The **Severity and Validation** step appears. It is the same as for Parameter Modification (see “Defining parameter modification tests” on page 229).

4. Set the severity level and validation options.
5. Click **Next**.

## What to do next

Continue with “Creating your own advisory”

### Creating your own advisory

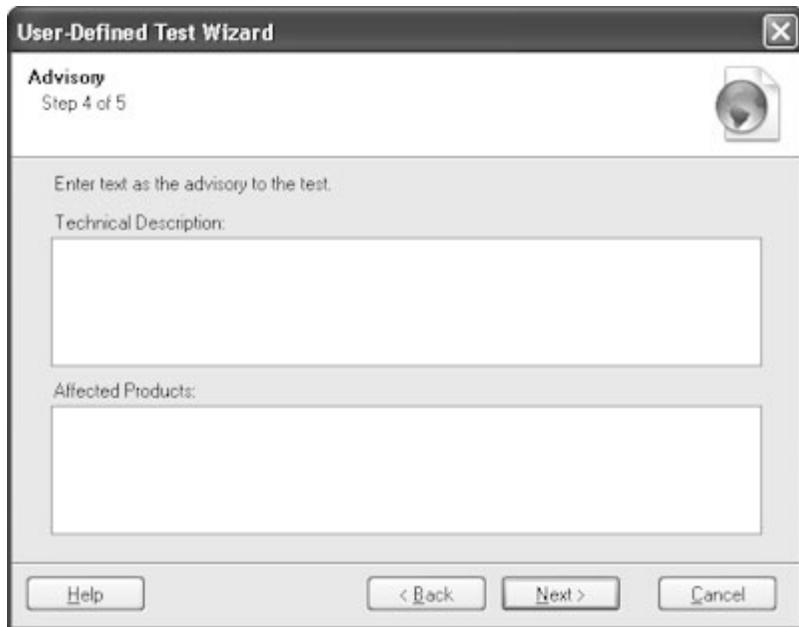
#### About this task

##### *Step 4 of 5*

After you define the test, create an Advisory for the security issue. If AppScan discovers an issue for your application as a result of the user-defined test, the Advisory information will help others understand the issue and provide them with fix information.

## Procedure

1. Click Next in the Severity and Validation step. The Advisory step appears.



2. Enter a technical description of this test. If this issue affects 3rd-Party products, list them in the **Affected Products** text box.
3. Click Next.

## What to do next

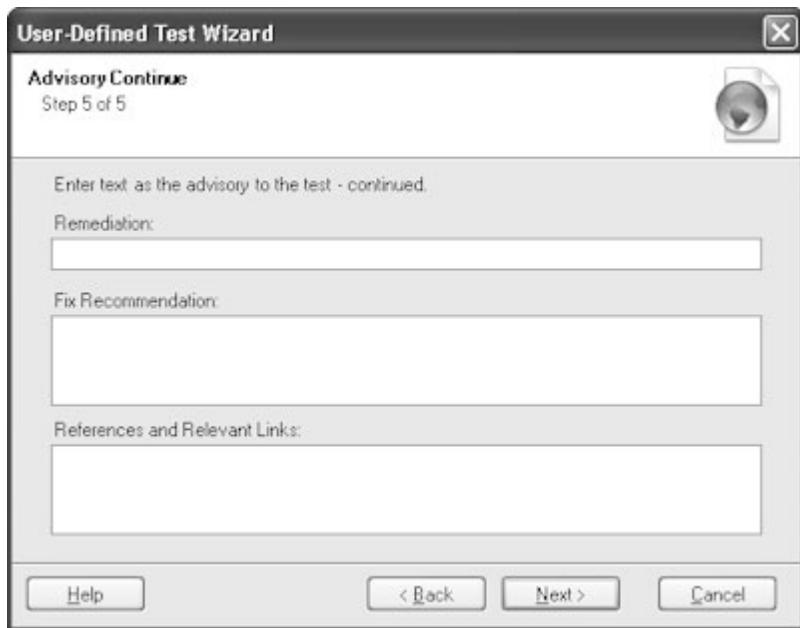
Next step: "(Advisory Continue step:) Fix recommendations and references"

### (Advisory Continue step:) Fix recommendations and references

#### About this task

*Step 5 of 5*

More options for the advisory are offered in the **Advisory Continue** step of the wizard.



## Procedure

1. In the **Remediation** text box, enter the general task to address the issue.
2. In the **Fix Recommendation** text box, enter the specific workaround or solution for this issue.
3. If you have references or external links that provide more information, enter them in the **References and Relevant Links** text box.
4. Click **Next**.

## Completing the user-defined test wizard

### About this task

After you click **Next** in the fifth step of the wizard, the final step of the wizard appears.

## Procedure

1. If you want to view the Test details when the wizard closes, select the **Show Test Details** check box.
2. Click **Finish** to save the new test and close the wizard.

## Managing user-defined tests

### About this task

Your test is saved to your AppScan database under the name you gave it in the first step. When the wizard closes, the **User-Defined Tests** dialog box is on top again, and your new test appears in the list.

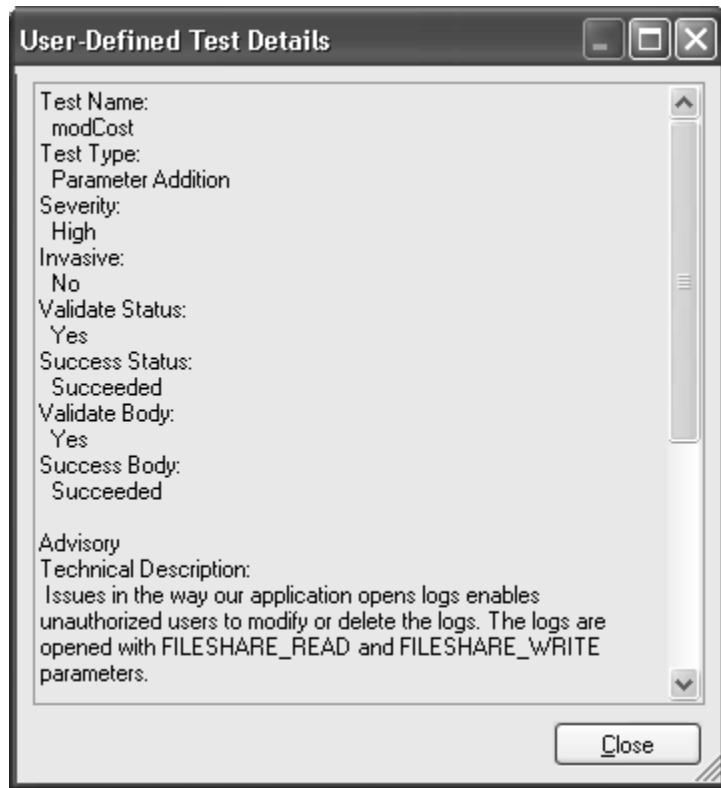
- To include the test in future scans, select the check box next to its name. It is now active.
- To exclude the test from future scans, clear its check box. The test is still saved, but is now inactive. (You can activate it when required.)

**Note:** You cannot delete user-defined tests once they have been created; but you can activate and deactivate them via the check boxes in this list, as required.

## Procedure

1. Select the row of the test in the **User Defined Tests** dialog box (not the check box).
2. Click **View Test**.

The **User-Defined Test Details** dialog box appears.



## PowerTools

AppScan offers access to IBM Security PowerTools. Each PowerTool is a utility providing a specific feature to help you manage your application security or to help you use AppScan.

Each PowerTool has its own Help file.

You can open the PowerTools from AppScan (**Tools > PowerTools >**) or from the Windows Start menu (IBM > AppScan Standard > PowerTools).

### Authentication Tester

This is a testing utility that uses the "brute-force" technique to reveal weak username-password combinations that could be used to gain access to your web application. (A brute force attack is an automated process of trial and error used to guess authentication credentials, causing a server to acknowledge an imposter as a legitimate user.)

Using brute force, a malicious user will cycle through combinations until stumbling upon credentials that gains access to the authorized area. Using a brute force application, a malicious user can employ a dictionary file, or even simply try all possible combinations of the accepted character set (depending on the username and password formats that the site accepts). Such an attack can generate thousands, even millions, of incorrect combinations before successfully gaining access, and typically takes hours, weeks, or more.

Ensuring that your web application enforces the use of strong passwords can greatly reduce the feasibility of brute force attacks.

- You can run Authentication Tester from AppScan, by clicking **Tools > PowerTools > Authentication Tester**

- You can run Authentication Tester on its own, from the Windows Start menu, by clicking **All Programs > IBM Security AppScan Standard > PowerTools > Authentication Tester**

## Authentication methods

In the Authentication Tester main window, you choose the authentication method used by your web application. Options are:

- **Form Authentication:** (authentication is performed by a custom web-page)  
“Form authentication”
- **HTTP Authentication:** (authentication as defined in the protocol)  
“HTTP authentication” on page 239

The scan options available change depending on which method you select.

**Note:** If the application uses *both* types of authentication (e.g. HTTP authentication in general and Form authentication for specific administrator pages), you must provide Authentication Tester with the *actual* username and password for the HTTP authentication, to enable it to test the Form authentication on these pages. (See “HTTP authentication over form authentication” on page 239 for details.)

### Form authentication

#### About this task

If you select the **Form Authentication** radio button, perform the following procedures:

#### Procedure

1. “Provide a typical login.”
2. “Describe the application's login responses” on page 238.

#### Provide a typical login:

#### About this task

If you select **Form Authentication** in the **Authentication Method** section of the main window, the **Setup** button appears. This is used to configure Authentication Tester with the correct Login procedure.

#### Procedure

1. Click **Setup**.  
The Authentication Tester browser opens.
2. Browse to the login page of your web application.
3. Perform the Login procedure using these credentials (you can cut-and-paste them from the upper part of the browser window):

username:	BruteUsername
password:	BrutePassword

Authentication Tester requires that you model the site login procedure using these values. During the testing stage, these strings will be replaced with possible username and password combinations as Authentication Tester attempts to gain access to the site by 'Brute Force'. When you complete the Login procedure now, however, Authentication Tester does not actually attempt to log in using these credentials, but simply examines the login request.

#### CAUTION:

If the strings "BruteUsername" and "BrutePassword" are not allowed by your client-side verification, no Login request will be created for Authentication Tester to examine! In such a case you must change the placeholder values for the username and password strings. See “Form Authentication tab” on page 241.

When you have completed the login process, Authentication Tester "captures" the login request, and a confirmation message appears.

4. On the confirmation message, click **OK**

The browser closes and the **Successful Login Detection** window opens. This is used to describe the login responses. See "Describe the application's login responses"

**Describe the application's login responses:**

**About this task**

In the Successful Login Detection window, you enable Authentication Tester to recognize login requests as either successful or failed. This information is necessary to know when the web application has accepted credentials as valid.

Some default Success responses appear in the list by default, but you should edit the list to include any and all responses that are specific to your application.

**Procedure**

1. Select the type of response that you want to describe:

- **Success Response:** The response to a valid login attempt
- **Error Response:** The response to an invalid login attempt

2. Enter a text string or regular expression (regexp) that matches some content on the response page. (Be sure to match only static content, not variables.)

For example, if you know that invalid credentials often receive a response of: "Username and password do not match" you can use this to let Authentication Tester know the result of its tests.

The use of regular expressions, rather than strings, allows you to configure Authentication Tester once for multiple runs during the development stages of your web application.

For example, if the design of the successful-login page has not yet been finalized between a large, full page **Welcome!** note or a small **welcome** string at the top of the home page, you can enter **(?i)welcome** to indicate that the search word is case-insensitive.

**Tip:** To learn more about regexps, and the metacharacters that can be used to indicate more than literal strings, see "About metacharacters." To test regular expressions before attempting to use them in Authentication Tester, try the Expression Test PowerTool.

3. Click **Add**.

The regexp is added to the list of responses.

You can add as many regexps as you want. Authentication Tester uses them with an OR operator: if one or more of the regular expressions matches content on a page of your site, that page is recognized as a result page (either successful login or error page, depending on your response type selection).

4. To remove unwanted regexps from the list, select the regexp and click **Remove**.

5. Click **OK**.

The **Successful Login Detection** window closes and you are returned to the main window. You can now run brute force tests using the current configuration (see "Running authentication tests" on page 240).

**About metacharacters:**

A metacharacter is one or more characters with a special, non-literal meaning in the context of a regular expression. For example, the circumflex character (^) is a metacharacter that means "search at the beginning". If you want to find the circumflex character, rather than the metacharacter pattern, you must protect (or "escape") it with a backslash: \^.

The table following lists examples of some common regular expression metacharacters.

	Metacharacter Description	Example
\	Protect the next character (interpret it literally and not as a metacharacter).	\! finds an exclamation mark (!) \. finds a period (.), rather than a character
^	Find at the beginning of a string.	^H finds <b>Home</b> but not <b>home</b> or <b>PHP</b>
.	Find any character (letter, number, symbol, whitespace), except newline.	(.*) finds any paragraph
( )	Find a pattern group.	(word) finds "In this <b>word</b> " ^(Word) finds " <b>Words</b> in this line" Welcome ((back) (home)) finds " <b>Welcome back</b> " and " <b>Welcome home</b> "
[ ]	Find a pattern range.	[a-z] finds any lower-case alphabetic character
*	Find the pattern zero or more times.	<(.*)> finds all HTML tags, with their content
+	Find the pattern one or more times.	(<.1>)+ finds <UL><OL>
?	Find the pattern zero or one time.	log(.?)in finds <b>login</b> and <b>log in</b>
(?i)	Find the next characters with a case-insensitive search.	(?i)word finds <b>word</b> , <b>Word</b> , <b>woRd</b> , <b>WORD</b>

## HTTP authentication

### About this task

If you select the **HTTP Authentication** radio button in the **Authentication Method** section, the **URL Address** and **Domain** fields appear.

### Procedure

1. Enter the URL of the login page in the **URL Address** text-box.
2. To test that this is the correct URL, click **Show**.  
The browser opens.
  - If a standard HTTP login window pops up in front of the browser, the URL is correct.
  - If the page appears without an HTTP login window, the URL is incorrect. Correct the URL.
3. Close the browser.
4. If the HTTP login window requires a domain, enter the correct domain name in the **Domain** field.  
You can now run brute force tests using the current configuration (see "Running authentication tests" on page 240).

## HTTP authentication over form authentication

### About this task

In some instances an application may utilize *both* types of authentication. HTTP authentication may be used generally on all pages, while Form authentication is used to protect specific administrator areas. In such cases you will probably want to run both types of test.

**HTTP Authentication:** This is tested as described above (see "HTTP authentication")

**Form Authentication:** To test this you need to provide Authentication Tester with the actual username and password for the HTTP authentication. This will enable it to "get past" the HTTP authentication and test the Form authentication on these pages.

## Procedure

1. In the Authentication Tester main window, select the **Form Authentication** radio button.
2. Configure Form Authentication (see “Form authentication” on page 237).
3. Click **Advanced**.  
The Advanced Configuration dialog box opens (with the General tab on top).
4. In the **HTTP authentication over Form authentication** area select the **Enable** checkbox.  
The Username, Password and Domain fields become active.
5. Type in valid HTTP authentication credentials for Authentication Tester to use when testing Form Authentication pages.
6. If the HTTP login window requires a domain, enter the correct domain name in the **Domain** field.
7. Click **OK** to close the dialog box.  
You can now run brute force tests using the current configuration (see “Running authentication tests”).

## Running authentication tests

After you have chosen the authentication method and configured the basic tests, you can start the Authentication Tester scan by clicking **Start**.

The progress bar on the main window shows:

- Number of username-password pairs to be tested, against the number of pairs to try.
- Number of threads Authentication Tester is using in this scan.
- Status of the scan, if not in progress: paused, resumed, stopped, finished.

If you **Pause** a scan and leave Authentication Tester open (do not exit), you can resume the scan later by clicking **Resume**. If you exit Authentication Tester after pausing, the scan is deleted.

If you **Stop** a scan, the data collected so far will be deleted.

## Scan results

When testing is complete, any username-password pairs that gained a successful login are listed in the table of Successful Logins.

You can save the results by exporting them in the form of an XML file. This feature is convenient for authentication testing during development, letting you compare the Authentication Tester results after implementing fixes.

Click **Export results** and give a name to the export file. An XML file is created, listing the credentials that successfully gained access.

## Advanced configuration

The Advanced Configuration dialog box lets you customize the behavior of Authentication Tester for your local network and the application you are testing.

**Tip:** If you will be testing both Form and HTTP Authentication, you can enter all information at the same time. Only information relevant to the type of testing currently being done (Form or HTTP) is used.

Clicking **Advanced** in the main window opens the **Advanced Configuration** dialog box opens. It has four tabs:

- “General tab” on page 241
- “Form Authentication tab” on page 241
- “Proxy tab” on page 241
- “Credential Generation tab” on page 242

### General tab:

The General tab of the Advanced Configuration window allows you to change network configurations.

Option	Description
Number of threads	Set the maximum number of connection threads Authentication Tester will test simultaneously. The default value is 12. More threads means a faster scan. If you find that Authentication Tester overloads your network you can reduce the number.
Request timeout (milliseconds)	Set the amount of time Authentication Tester should be given to reach the web application server before timing out the connection.
Internal proxy port	An available port is assigned automatically for the internal proxy. If necessary you can assign a different port. You will receive a warning if the port you select is unavailable.
HTTP Authentication over Form Authentication	If the web application uses HTTP authentication over form authentication, in order to test the <i>form</i> authentication you must select <b>Enable</b> , and enter valid values for the HTTP authentication <b>username</b> , <b>password</b> , and <b>domain</b> . This will allow Authentication Tester to access the Web application server to run the brute force tests on the Form Authentication mechanism.  These values do not affect HTTP Authentication testing; they are applied only if you select the <b>Form Authentication</b> method.

### Form Authentication tab:

This tab holds the Successful Login regular expressions that you enter to describe the page that is sent in response to an accepted username-password. (Information entered here applies only when Form Authentication is tested.)

Option	Description
<i>Successful Login Detection</i>	
Success Response	When selected, currently configured Success Responses are displayed in the pane below.
Error Response	When selected, currently configured Error Responses are displayed in the pane below.
Add/Remove	See “Describe the application’s login responses” on page 238 for instructions on adding responses to the lists; or “About metacharacters” on page 238 to learn more about writing regular expressions.
<i>Default Counterfeit Credentials</i>	
Username/Password	<p>These fields show the strings that you will be asked to enter when creating the login request to configure Form Authentication (see “Form authentication” on page 237). They do not need to be <i>valid</i> credentials as they are not used to actually log in to the site. They are simply used to identify to Authentication Tester the location of the credentials in the login request (for use in Brute Forcing the site).</p> <p>The default values are <b>BruteUsername</b> and <b>BrutePassword</b>. If client-side logic will not allow these values to be used in a login request to the site (for example, if the application requires an email as the username, and client-side logic enforces this rule when creating the login request), change these values to a valid format.</p> <p>When changing the default counterfeit credentials, make sure that neither value is a sub-string of the other. For example, if you enter <code>user@email.com</code> as the username, you may not use <code>user</code> as the password.</p>

### Proxy tab:

If your web application requires a web proxy connection, select the **Enable** checkbox on the **Proxy** tab and fill in the Server and Port. If the proxy requires authentication, fill in Username and Password, and Domain if required.

#### Credential Generation tab:

The options on this tab enable you to configure the usernames and passwords that Authentication Tester tries during the scan.

It has two areas:

- **Mode:** “Credential Generation: Mode area”
- **Configuration:** “Credential Generation: Configuration area”

#### *Credential Generation: Mode area:*

##### Procedure

Select which mode Authentication Tester will run:

- **Matrix:** Each username in the list is tried against every password in the list. Select this mode for a more comprehensive scan
- **Parallel:** Usernames and passwords are paired by index number. Select this mode for a faster scan

#### *Credential Generation: Configuration area:*

##### About this task

In this area you configure the options that Authentication Tester uses to generate the usernames and/or the passwords to be tested.

##### Procedure

1. Select one of the radio buttons to customize **Usernames** or **Passwords**.

All fields and data displayed in the Configuration area now apply to the selected item. For each item the drop-down list lets you configure the structure of the usernames/passwords that will be created and used in the attack.

2. From the **Element Type** drop-down list, select an element type that you want to be included in the attack, and in the field below it, fill in values for the element as described in “List types.”
3. Click **Add** to add the new definition to the list in the Configuration pane.
4. Repeat for additional elements as necessary.
5. Arrange the element types (using **Up** and **Down**) so that each username or password that is tried will be built of each type, in the order listed.

##### *List types:*

Element	Description	Sample Values
Dictionary	A "value-per-line" file used when generating usernames/passwords for the attack.  Default files are provided for both Usernames and Passwords, but you can use the <b>Browse</b> button to browse to any other suitable file. The data folder also includes a much larger password dictionary file (passwords_long.txt) that can be used instead of the default file.	pathname to "value-per-line" dictionary file
Numeric	A range for numbers that will be included in the usernames/passwords generated	0 to 99999999
Constant	A string that you want included in all usernames/passwords that are generated	any string

Element	Description	Sample Values
Character Range	A range of characters and string length, for characters that will be included in the usernames/passwords that are generated	[space to tilde] ~~ and a string length

**Note:** The **space to tilde** range includes a-z, A-Z, 0-9, and ASCII input symbols. If you enter " ~~" (without the quotes) as a Character Range, Authentication Tester automatically inserts all included characters in the regular expression listing.

The valid value of the **length** field depends on the range. For example, if the range is 0-9 and the length is 10, the range is valid [0000000000, 0000000001, ...9999999999]; but if the range is a-z, Authentication Tester will not accept 10 as a valid length because the number of combinations would take an unreasonable amount of time and resources.

**Note:** To use a dash as a character, rather than to denote a range, precede it with a backslash: \-

*Examples of Credential Generation configuration.:*

Credential Generation	Resulting Credentials
Dictionary: C:\web tests\data\users.txt Constant value: _ Numbers range: 0 to 9	user_0 user_1 user_2 ... user_9
Constant value: passwd Character Range: a-zA-Z0-9; length: 3	passwdaaa passwdaab ... passwd999
Constant value: iamgod Character Range: ~~; length: 1 Numbers range: 1900 to 3000	iamgod 1900 iamgod!1900 ... iamgod~3000

*Deleting an element:*

#### Procedure

Select the unwanted element in the lower pane, by clicking on it, and then click **Remove**.

## Connection Test

This PowerTool enables you to ping web sites without using the Ping protocol, which is blocked by many firewalls.

- You can run Connection Test from AppScan, by clicking **Tools > PowerTools > Connection Test**
- You can run Connection Test on its own, from the Windows Start menu, by clicking **All Programs > IBM Security AppScan Standard > PowerTools > Connection Test**

## Using Connection Test

#### Procedure

1. In the **Web Site** text box, type or paste the URL.
2. Click **Ping Site**.

The Ping Results list shows whether Connection Test was able to successfully reach the URL and if so, how many milliseconds it needed to reach the URL.

#### What to do next

- To clear the Ping Results list, click **Clear**.

#### HTTP Method:

The **HTTP Method** drop-down menu offers two HTTP Request Methods.

- **HEAD:** Select to get only HTTP headers. This is the faster method.
- **GET:** Select to get entire page. Some sites do not allow HTTP requests using the Head method; in this case, try the Get method.

#### **Port:**

In the Port text box, type the listening port of the Web server.

The default for most non-secure URLs is **80**. You can change the port for specific URLs if required.

To use SSL (Secure Sockets Layer), check **Secure**. The default port changes to **443**.

#### **No. of Intervals:**

In the **No. of Intervals** text box, type the number of times Connection Test should ping the given URL.

- If you only want to test that you can reach the URL, use a small value (such as 3 to 5).
- If you need to monitor server crashes, the value should be much larger (such as 100 to 1000). Connection Test works in the background while you carry on with other tasks, so you can make the number of intervals as large as you wish.

You can stop Connection Test at any time before the next ping by clicking **Stop**.

#### **Interval Wait:**

In the **Interval Wait** text box, type the number of milliseconds for Connection Test to wait between pings.

- If you want to test that you can reach the URL, the value could be 0 (zero).
- If you need to monitor server crashes, the value should be larger (such as 2000, which is two seconds).

#### **Show Server Header:**

Check the **Show Server Header** checkbox to see the type of Web server being pinged. The information is displayed after the final interval is reached.

Server header examples:

- www.ibm.com - Microsoft-IIS/6.0
- www.cnn.com - Apache
- www.sky-news.co.uk - Microsoft-IIS/5.0
- www.sourceforge.org - Apache/1.3.31 (Unix) PHP/4.3.11 mod\_ssl/2.8.19 OpenSSL/0.9.7a

## **EncodeDecode**

This PowerTool encodes and decodes strings you put into it, to and from the format of your choice.

- You can run EncodeDecode from AppScan, by clicking **Tools > PowerTools > EncodeDecode**
- You can run EncodeDecode on its own, from the Windows Start menu, by clicking **All Programs > IBM Security AppScan Standard > PowerTools > EncodeDecode**

## **Using EncodeDecode**

### **Procedure**

1. In the Input text box, type or paste the text.
2. From the Method drop-down menu, select an encoding/decoding method. (See “Methods” on page 245)

If you choose 3DES for the method, type or paste a suitable encryption key in the Encryption Key text box.

3. Click Encode or Decode.

The encoded or decoded text appears in the Output box.

**Layered Encoding/Decoding:** You can send output back as input and encode or decode again, by clicking Send Back.

This is useful if you want to:

- Ensure encoding is 1-to-1 with decoding
- Perform multiple encoding on a string
- Decode a string that has been multiply encrypted

**Methods:**

Method	Capability	Translate between
URL	encode + decode	text <-> URL encoding
Base64	encode + decode	text <-> Base64 encoding
Overlong UTF-8	encode + decode	text <-> Overlong UTF-8 (2 bytes)
UU	encode + decode	text <-> UU encoding
HTML	encode + decode	text <-> HTML entities
MD5	encode	text <-> digital signature
SHA1	encode	text <-> digital signature
SHA256	encode	text <-> digital signature
SHA384	encode	text <-> digital signature
SHA512	encode	text <-> digital signature
3DES (192bit key)	encode + decode	readable text <-> encrypted text (using a given key)

## Expression Test

Writing precise regular expressions can be a tedious trial-and-error process. You can use this PowerTool to help accelerate the process.

- You can run Expression Test from AppScan, by clicking **Tools > PowerTools > Expression Test**
- You can run Expression Test on its own, from the Windows Start menu, by clicking **All Programs > IBM Security AppScan Standard > PowerTools > Expression Test**

## Using Expression Test

### Procedure

1. In the **Text** field, type or paste text that you want the regexp to find.
2. In the **Regular Expression** field, type the regular expression that you believe will find the text.
3. Click **Test**.

## Results

Text in the Text box that matches the pattern is highlighted in red.

### Text replacements:

Replacing text that matches a specific pattern.

### About this task

If you are using regular expressions to replace text that matches a specific pattern, you can use Expression Test to test that the replacement will be done correctly.

### Procedure

1. In the **Regular Expression** field, type the regular expression.
2. In the **Replace** box, type the text that should replace the text that matches the pattern of the regular expression.
3. In the **Text** box, type or paste example text.
4. Click **Test** and check that the predicted text is matched by the regular expression.
5. Click **Replace**.

## Results

The text that matched the regular expression pattern (highlighted in red) is replaced with the text in the **Replace** box.

### Pattern groups:

Applying a regular expression to a group of characters or metacharacters.

Often you need to apply a regular expression to a group of characters or metacharacters.

To group a set, put the characters in parentheses ( ).

The group is given a number in the Match Groups box and the matching text is displayed next to it.

If a match occurs, you can see which text matches which group in the Match Groups box.

For example, you have an HTML code with <UL>first in the text.

You type (<.L>) ([a-z ]\*) as the regular expression.

In the Text field, <UL>first is highlighted in red.

In **Match Groups**, the groups are broken down:

- Group 1 <UL>
- Group 2 first

### Metacharacters:

General information about metacharacters.

Any single character (letter, digit, or symbol) in a regular expression is matched to itself, literally; unless it is a metacharacter. A metacharacter is one or more characters that have a unique meaning and are not used literally in the regular expression match.

For example, the circumflex character (^) is a metacharacter that means "search at the beginning".

If you want to find the character, rather than the metacharacter pattern, put a backslash (\) before it.

For example, to find the circumflex as a text character, the regular expression must be: \^

Reg. Exp.	Description	Example
\	Find the next character as a character; do not use its metacharacter pattern.	\. finds a period (.) in the text . finds the first character (any character)
^	Find at the beginning of a string	^1 finds "1. Click Save." but not: "in the 210th line"
.	Find any character (except newline).	Finds a, A, 1, <, ., =, and so on; whatever the first character is.
( )	Find a pattern group.	(word) finds "In this <b>word</b> " ^(Word) finds " <b>Words</b> in this line"
[]	Find a pattern range.	[a-z] finds letters, but not numbers
*	Find the previous pattern zero or more times.	.* finds all characters <(.*> finds all HTML tags
+	Finds the previous pattern one or more times.	(<.1>)+ finds <UL><OL>
?	Finds the previous pattern zero or one time.	(<.1>)? finds <UL>
(?i)	Find the next characters with a case-insensitive search.	(?i)word finds <b>word</b> and <b>Word</b>

## HTTP Request Editor

This PowerTool enables you to send a fully-controlled HTTP request to your site, to test how your site responds to different kinds of HTTP request.

- You can run HTTP Request Editor from AppScan, by clicking **Tools > PowerTools > HTTP Request Editor**
- You can run HTTP Request Editor on its own, from the Windows Start menu, by clicking **All Programs > IBM Security AppScan Standard > PowerTools > HTTP Request Editor**

### Using HTTP Request Editor

#### Procedure

1. In the **Request** tab, fill in the relevant fields.
2. Click **Send**.
3. Open the **Response** tab to see how the host dealt with the request.

#### Request tab

**View options:** The View options in the Request tab enable you to create an HTTP request in different ways:

- **Raw:** Create the request by typing or pasting text.
- **Parsed:** Create the request by filling in the form.

To update the raw HTTP request with details entered in the Parsed view, on the Raw tab, click **Update from Parsed**.

The Request that you set up by filling out the parsed form is displayed in raw form.

#### Request settings:

Setting	Description
Host	Type in the IP address or hostname of your website.
Port	Type in the TCP port on which your web server is listening. Default=80
Method	Select a method from the drop-down menu, or enter a value for the method request parameter. The default drop-down menu includes:  GET: Retrieve all information identified by the Request-URI.  POST: Request that origin server accept an entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line. The actual function performed by this method is determined by the server and is usually dependent on the Request-URI.  HEAD: Retrieve information identified by the Request-URI, but the server must not return a message-body in response. Meta-information in the HTTP headers received in response to a HEAD request are the same as those received in response to a GET request. Use to obtain information without transferring the entire body; often used to test hypertext links for validity, accessibility, and modifications.
Secure	Check to send request using HTTPS.
URL	Enter or paste a URL.  You can view the list of links extracted from the last response in the Response Tab.
HTTP Version	Enter the version number of the HTTP protocol that the request should use. Default=1.1
Auto Content Length	When checked, HTTP Request Editor will automatically calculate and add the Content-Length header and its correct value.

**Message details:** Create or modify parameters, headers, cookies, and their values, to include in the request.

#### Parameters:

##### *Adding parameters:*

##### **Procedure**

1. Click **Add**.  
The Add Parameter dialog box opens.
2. Enter a name for the parameter and then enter a value.
3. Select the type:
  - **Body:** Send the parameter in the body of the request.
  - **Query:** Send the parameter in the query part of the request.
4. Click **OK**.  
The parameter is added to the request.

In the Parsed view, the Parameters table shows the name, value, and location that you entered. In the Raw view, the parameter appears differently, depending on the location that you chose:

- **Body:** a name=value pair will appear in the body of the request.
  - **Query:** a name=value pair will appear in the query part of the request (after the ? symbol)
- In both cases, the name=value pairs will be concatenated with & if more than one pair exists.

*Editing parameters:*

#### **Procedure**

Do one of the following:

- In the Parsed view: select the parameter and click **Edit**.
- In the Raw view: edit the text.

*Removing parameters:*

#### **About this task**

You can remove parameters from the request without deleting them from HTTP Request Editor.

#### **Procedure**

In the Parsed view, uncheck the parameter name.

*Deleting parameters:*

#### **Procedure**

Do one of the following:

- In the Parsed view, select the parameter and click **Remove**.
- In the Raw view, delete the text.

*Headers:*

*Adding headers:*

#### **Procedure**

1. Click **Add**.

The Add Header dialog box opens.

2. Enter a name and a value for the new HTTP header.

3. Click **OK**.

The header is added to the request.

- In Parsed view, the Headers table shows the name/value of the new HTTP header.
- In Raw view, the header appears as: Header name: Header value

*Editing headers:*

#### **Procedure**

Do one of the following:

- In Parsed view, select the header name and click **Edit**.
- In Raw view, locate the header you want to edit and modify its text.

*Removing headers:*

#### **About this task**

You can remove headers from the request without deleting them from HTTP Request Editor.

## **Procedure**

In Parsed view, clear the checkbox next to the header name.

The HTTP Request Editor generates a request with several default headers (Accept, Host, User Agent). You can remove or modify these headers just like any other header.

*Deleting headers:*

## **Procedure**

Do one of the following:

- In Parsed view, select the header name and click **Remove**.
- In Raw view, locate the HTTP header you want to delete and remove the text from the request.

*Cookies:*

*Adding cookies:*

## **Procedure**

1. Click **Add**.

The Add Cookie dialog box opens.

2. Enter a name for the cookie and then enter a value.

3. Click **OK**.

The cookie is added to the request.

- In Parsed view, the Cookies table shows the name and value that you entered.
- In Raw view, the cookie appears in the header: `Cookie: name=value`

*Editing cookies:*

## **Procedure**

Do one of the following:

- In Parsed view: select the cookie and click **Edit**.
- In Raw view: locate the cookie header you want to change and edit the text.

*Removing cookies:*

## **About this task**

You can remove cookies from the request without deleting them from HTTP Request Editor.

## **Procedure**

In Parsed view, clear the checkbox of the cookie name.

*Deleting cookies:*

## **Procedure**

Do one of the following:

- In Parsed view: select the cookie and click **Remove**.
- In Raw view: locate the cookie header you want to delete and remove the text.

## **Response Tab**

After you click **Send** in the Request tab, you will get a response from the web server.

There are three ways to view a response:

- **Raw:** Response is presented in raw text

- **Parsed:** List of links extracted from the HTML in the response, and SSL information
- **Browser:** Response is shown in a web browser

**Note:**

Only the actual response that was sent by the web server is presented in the browser. Redirections, images, and CSS are not requested automatically.

In raw view, you have the option to search for text in the response. In the text box at the bottom of the window, enter a regular expression (for example: "[a-z0-9]@" to find an email) and then click **Search**.

Check **Case Insensitive** to make the pattern match both upper- and lower-case letters.

## Token Analyzer

This PowerTool provides various tests for your web application session tokens, to help you evaluate how well secured your application is against session theft. Based on mathematical algorithms, these tests can reveal weaknesses in your session tokens.

- You can run Token Analyzer from the AppScan Tools menu, by selecting **Tools > PowerTools > Token Analyzer**
- You can run Token Analyzer from the Windows Start menu, by selecting **All Programs > IBM Security AppScan Standard > PowerTools > Token Analyzer**

## Understanding session tokens and the problem they pose

- “What are session tokens?”
- “How do session tokens cause security issues?” on page 252
- “How can Token Analyzer help?” on page 252

**What are session tokens?:** HTTP is a “stateless” protocol, which means that Web application servers respond to client requests without linking multiple requests to any one client. However, the security and/or functioning of many web applications depends on the system being able to distinguish between users, and to recognize users and their permission levels. This is solved by the use of session tokens.

The transfer of session tokens between your site and the client provides “state” to HTTP. They enable the site to respond appropriately to each user during a session, keeping track of the user’s activity (requests and responses) within the application.

There are three types of session token:

- Cookies
- URLs
- Post data

### Cookies

Cookies are simple files originating from your site, and stored on customers’ computers by their browsers. They usually contain a name and a value that are used to identify the client as a specific user with certain permissions on your site. (They may also contain additional properties that we will not discuss here.)

A cookie is connected with its source domain in such a way that the information stored in it can be accessed *only* by the source domain and no other. A third-party server can neither read nor change the content of cookies on your user’s computer that originate from your domain.

## URLs

A URL can contain a parameter with the name and value of a session token.

After the customer logs in, the application server assigns a session token to the customer and replaces variables in the linked URLs with random token IDs. The server keeps track of the URL tokens and pairs them with the customer's session token. When the customer's browser sends a request for a URL, it includes the URL session token.

For example, a link to `http://www.bank.com/account.jsp?account=10&type=today` will be changed to `http://www.bank.com/account.jsp?token=1234`

## Post Data

Post data information is information sent in the body of an HTTP request. A customer's browser can send session token name-value pairs embedded into a request.

**How do session tokens cause security issues?:** If a malicious user succeeds in predicting valid values of cookies or URL parameters, s/he can enter your web application under the guise of a legitimate customer, "stealing" a session. Session theft allows the malicious user to access information and permissions that are given to legitimate customers during valid sessions. Session tokens are a security issue if their values are predictable.

Alternatively, a web application cookie or URL contains user information, such as a valid username or password. This is especially true with URL session tokens that contain readable session token parameters. If a malicious user is able to collect these tokens, either through system invasion or network interception, this user would have enough information to steal legitimate customer sessions.

**How can Token Analyzer help?:** An essential step in securing your web application is ensuring that session token values are distributed randomly (that they are as "unpredictable" as possible) and that they do not contain readable strings that a malicious user could easily identify as session information (such as "user", "pass", "token", "ID").

Your web application contains algorithms that generate the session token values. You can check the efficiency of these algorithms (without knowing the code or the underlying algorithm), using Token Analyzer. Token Analyzer results can alert you to session tokens that may constitute a security risk, so you can take the necessary steps to fix the issue.

Token Analyzer sends a predetermined number of requests to a URL on your site that returns a specific cookie, and analyzes the contents of the cookie. You can then analyze all or a segment of the cookie, as required. The results of the analysis are presented in a series of charts.

## Workflow

### About this task

The use of Token Analyzer to analyze your Web application session tokens must be interactive to ensure that the collection of data and the results that you receive are relevant for your application.

### Procedure

1. **Start a new session:** Browse to the web application's URL in the embedded browser, and browse to a page that sends the session token you want to analyze.  
Token Analyzer lists the requests sent, and session tokens sent/received per request.
2. **Select a session token:** Select the required session token from the list and define the number of samples to collect.

Token Analyzer sends the selected request repeatedly, until the specified number of samples is received, and displays the results as a table in the upper pane.

3. **Select which part of the results to analyze:** Select all or a subset of the results and initiate an analysis.  
Token Analyzer analyzes the selection and displays the results in the tabs in the lower half of the window.
4. **Review the results:** Review the resulting graphs to evaluate the predictability of your site's session tokens.  
As a result of your review you may make additional selections from the results on which to run a Token Analyzer analysis.

## Starting a new session

### About this task

Browse to a page of the web application that sends one or more session tokens, and take a sample of the tokens.

### Procedure

1. On the Token Analyzer main screen, click **New**.  
The New Session window opens.
2. In the **Address** field above the left pane, type in the URL of the application and click **Go**.  
The embedded browser opens the page in the left pane.
3. If necessary, browse to the page with the session token you wish to analyze.  
Each URL you browse to is listed in the **Request View** pane on the upper right. For the currently selected URL, details of cookies sent, received and of post data (included as part of the URL) appear in the lower **Details** pane.
4. **Request View** pane, select the URL that includes the session token you want to analyze.  
The URL is highlighted in the upper pane, and relevant session token information appears in the lower pane.
5. In the **Number of Samples** field, define the number of sample session tokens for Token Analyzer to collect. For greater accuracy, increase the number (max. 1,000); for a faster test, decrease it (min. 10).
6. If your web application requires HTTP or NTLM authentication, click **Advanced**, to open the "Advanced settings" dialog box and configure Token Analyzer to supply this authentication during its sample collecting. (See "Advanced settings" for details.)
7. Click **OK**.  
The New Session window closes and a "Collecting Session Data" message appears. When the process is complete the data is displayed as a table in the upper pane of the main screen.

### What to do next

See: "Analyzing session data" on page 254

See also: "Advanced settings"

#### Advanced settings:

If your web application requires HTTP or NTLM authentication, click **Advanced**, to open the Advanced Settings dialog box.

Tab	Item	Description
General	Enable	Click to enable HTML/NTLM Authentication, then type in Username, Password and (optionally) Domain.

Tab	Item	Description
	Timeout	Enter a time limit (seconds) for Token Analyzer to wait for a response from the server
Proxy	Enable	Click to enable a Proxy, then enter Address (and port), and Username, Password and (optionally) Domain.

## Analyzing session data

The session data collected from the selected request is displayed in the **Session Tokens** table in the upper pane of the main window, by name and type (cookie or URL).

The screenshot shows a table titled "Session Tokens" with a dropdown menu set to "ASP.NET\_SessionId (Cookie)". The table has 23 columns labeled 0 through 22. The first column contains the letter "T" above each row, indicating the allowed character set for that position. The rows represent different samples of the cookie, with each sample having 45 columns corresponding to the 0-44 range. The data consists of various letters and numbers, such as "t s f b 1 3 n o e k 1 4 g s n y y z i q m w v". The table includes standard scroll bars and navigation buttons at the bottom.

For each token (selected from the drop-down list) the table shows each sample collected as a row in the table, with each character in the cookie as a column. For example:

- If the token contains 45 characters, there will be 45 columns (from 0-44)
- If you configured Token Analyzer to take 100 samples, there will be 100 rows in the table (from 0-99)

At the top of each column appears a color-coded character set symbol. This indicates the characters that are allowed for that particular position in the token. (See "Character set classification" on page 255 for details.)

**Viewing a different token:**

## Procedure

Select a session token from the **Session Tokens** drop-down list.

Session Tokens:			ASP.NET_SessionId (Cookie)												
			Limit (Cookie)												
			Interest (Cookie)												
			CardType (Cookie)												
			UserInfo (Cookie)												
			URL SessionID (URL)												
0	t	s	f	4	g	s	n	y	y	z	i	q	m	w	v
1	5	c	n	5	e	y	5	5	x	4	v	i	z	h	5
2	3	v	5	k	p	u	5	5	y	y	h	i	d	t	2
3	a	g	a	z	5	d	3	j	l	o	1	o	q	i	4
4	o	f	a	t	g	s	n	j	k	o	v	c	f	k	5
5	u	0	l	d	z	p	f	r	p	s	4	p	y	t	y
6	c	i	g	u	p	r	3	g	a	r	w	f	e	g	5
7	m	j	4	0	2	5	m	c	w	s	k	o	5	2	2

The data of the selected token is displayed in the table.

### Analyzing the data of a particular token:

#### Procedure

1. Select a session token from the **Session Tokens** drop-down list.
2. Do one of the following:
  - To select the entire session token, click **Select All**.
  - To select consecutive columns, click-and-drag the mouse across the columns you want.
  - To select non-consecutive columns, click while holding down the **Ctrl** key.
3. Click **Analyze**.

The data is analyzed and the results summarized in the Analysis Result tabs in the lower half of the main screen.

#### What to do next

See: "Analysis Results" on page 256

See also: "Character set classification" "Selecting subsets" on page 256

#### Character set classification:

The first row in the session data table shows the permitted character set for that position in the session token. Obviously, evaluating this correctly is essential to being able to analyze the predictability of that position in the token.

The table below summarizes these classifications used by Token Analyzer:

Class	Label	Description
Number	N	0123456789
Alphabetic	A	abcdefghijklmnopqrstuvwxyz (case insensitive)
Text	T	abcdefghijklmnopqrstuvwxyz 0123456789 (case insensitive)

Class	Label	Description
Base64	B	ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz 0123456789+/_
Hexadecimal	H	0123456789abcdef (case insensitive)
Delimiter	D	Includes only characters that are not in the previous sets
Other	O	Includes characters from both Base64 and Delimiter sets

For example:

- If the first character of the first collected value of a cookie is “w”, it might be Alphabetic, Text, Base64, Hexadecimal, or Other.
- If the first character in the fifteenth collected value is “2”, the character is either Text or Base64.
- If “+” or “/” do not appear in any of the collected samples, the character is classified as Text. However, if the 104th sample has “&” as the first character, the character should be classified as Other.

This example shows how even in the first and most basic of the value analysis, having more samples can mean the difference between true or faulty evaluations.

### Selecting subsets:

#### About this task

Session tokens are often comprised of subsets. Each subset may be created by a different token-creator algorithm. In this case, it would not be relevant to analyze the session token as a whole; you want the tests to show whether each algorithm is producing random values. Therefore, Token Analyzer allows you select a subset of columns before running the cruncher tests.

You could select subsets at random, to see if the test results are interesting. You could select subsets of similar character classifications. Or you could examine the session token of the selected session and attempt to view it as a malicious user might; where does one subset seem to end and the next begin?

For example, delimiters are already marked for you in the character classification. You, or a malicious user, could choose the characters between delimiters as an educated guess that each subset will be separated by symbols that are different from the valid values of the subsets.

#### Procedure

1. Select a session token from the **Session Tokens** list.
2. Do one of the following:
  - To select consecutive columns, drag the mouse across the column headers you want.
  - To select non-consecutive columns, use the Ctrl or Shift keys.
  - To select the entire session token, click **Select All**.

### Analysis Results

The result of the analysis are displayed in five tabs:

- “Size Analysis” on page 257
- “Singles Count” on page 257
- “Column Chi-Square” on page 258
- “Pairs Count” on page 259
- “Correlation” on page 261

The sixth tab is for running “String Search” on page 262

### Size Analysis:

Token Analyzer provides statistics on the size of the session token: shorter values are easier to guess. Secure session tokens are long. The desired length of a token depends on the character set classification. For example, a value made of Number-type characters should be at least twelve-digits long.

If a session token, or a discernible subset, is too short, the entire session token can be more easily predicted.

This test also provides the standard deviation and variance values, which describe the spread of session token lengths around the average length (0.00 indicates that all selected session tokens are the same length).

*Table 4. Example of secure size*

Number of tokens checked	1000
Max session token length	24 Characters
Min session token length	24 Characters
Average session token length	24.00 Characters
Variance	0.00
Standard Deviation	0.00

This token above is strong, because it must always be a secure length of 24 characters.

*Table 5. Example of weak size*

Number of tokens checked	300
Max session token length	20 Characters
Min session token length	3 Characters
Average session token length	8.87 Characters
Variance	19.31
Standard Deviation	4.39

Although the maximum size of the above sample session token is 20 characters, which is a secure length in most cases; the average size of less than nine characters, and the minimum size of only three, may indicate that this token is weak.

### Singles Count:

This test counts the appearances of each character. In general:

#### Secure tokens

use most or all of the characters in the permitted character set, and no character is used more often than the others.

#### Weak tokens

have a non-random character distribution. For example, it might use only a few of the permitted characters, or some characters appear so much more often than others that they can be guessed, making the entire token more predictable.

The graph shows each character value on the X axis, and the number of times that it appeared on the Y axis. A dotted blue line indicates the Mean or “expected number of appearances” for each character.

The Singles Count graph may indicate that a session token is weak if:

- One (or more) characters are significantly above the Mean line (meaning they appear “frequently”)
- Several characters that are above the line (even slightly) share common characteristics (for example, they are close to one another in the character set)
- One or more valid characters do not appear at all

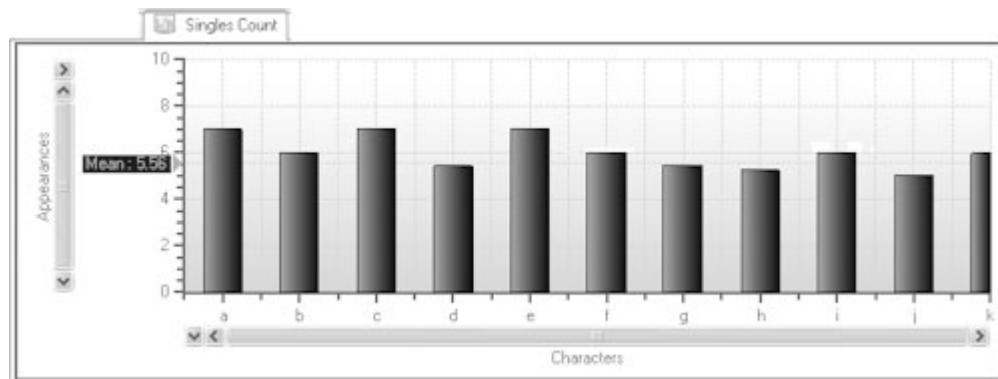


Figure 1. Secure number of appearances of each character

In the above example of a secure token, the values are all near the mean value.

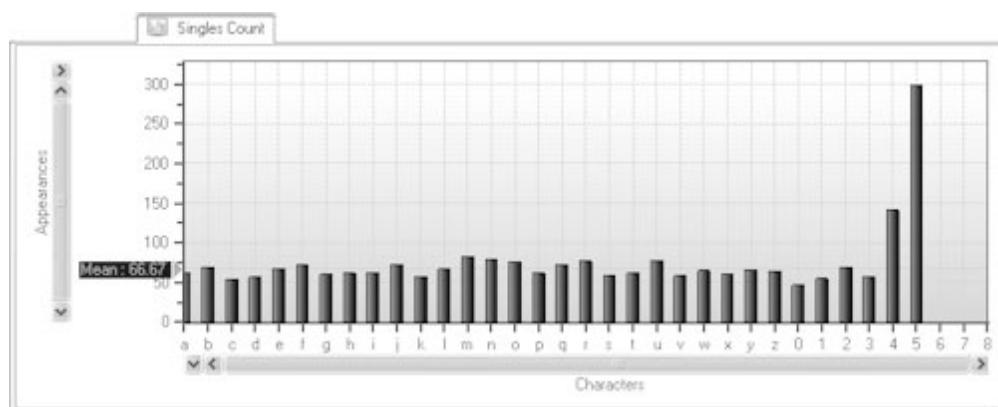


Figure 2. Weak session token discovered in Singles Count

In this example the characters “4” and “5” appear much more often than would be expected in a random token, and may be predictable.

Notice also that characters above 6 do not appear at all. This could indicate either that the session token does not use a random selection of the full character set, or that the initial analysis of which character set is used is faulty.

#### Column Chi-Square:

This test checks the selected columns using the Chi-Square method, which finds whether each column, over the number of samples taken, is random or not. The chart shows each selected column of the token (X axis) against its chi-square score (Y axis). The higher the score, the further from random the column is.

Low (secure) Chi-square scores are indicated by bright blue columns. Scores of 600 and above are very insecure, and indicated by bright red columns. In between this are various shades of purple. In general:

**Secure token:**

All columns have low chi-square scores.

**Weak token:**

Some or all columns have much higher chi-square scores than others.

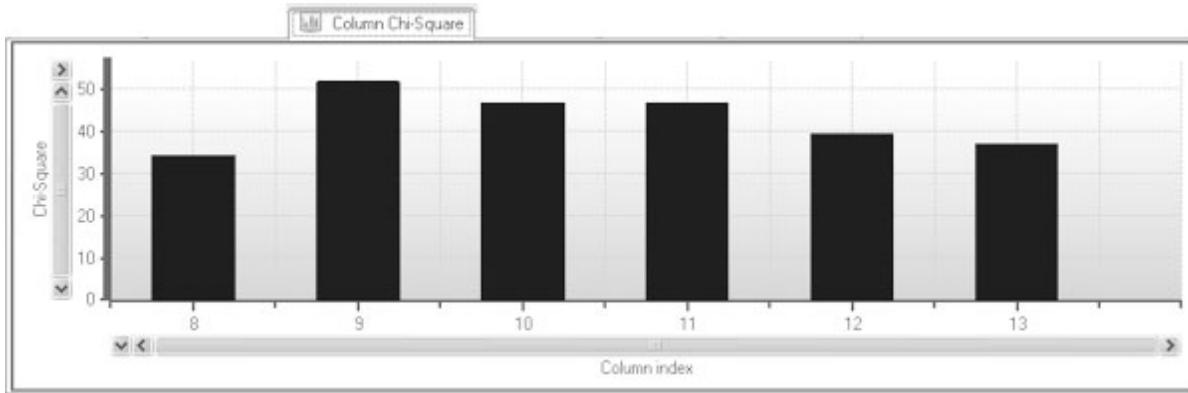


Figure 3. Secure Chi-Square

In a secure session token, the Chi-Square results are all similar low values (blue-shading), indicating secure random distribution for each character in the token.

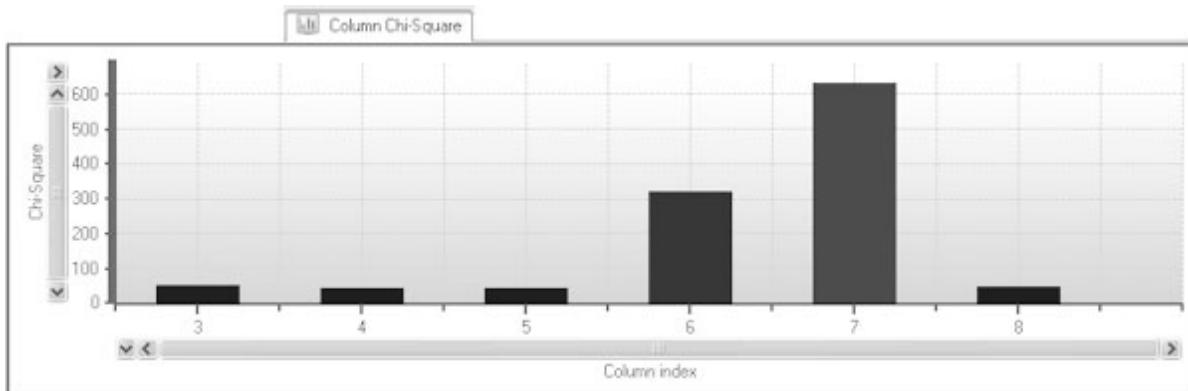


Figure 4. Chi-Square discovers weakness in specific positions

In this sample of the 4th column is purple, indicating a weak position in the token, and the 5th column is red, indicating that its chi-square value is above 600, and is probably easy to predict.

Note that the Chi-Square test results are dependent upon the character set:

- Columns classified as Delimiters are always given a score of 0
- Columns of different classifications will have different score ranges. For example, the scores of a Base64 column will be higher than the scores of an Alphabetic column, even if the characters appear the same number of times.

**Pairs Count:**

This test counts the appearances of overlapping character pairs in each session token.

A secure session token will show a random distribution in the Pairs Count: using all available characters in random pairs, with no pair appearing more often than others.

A weak session token may have discernible patterns, such as a specific character always appearing after another character, pairs missing from the results, or pairs that appear more often than others.

The Pairs Count test provides different views for its results: **Graph** and **Table**.

The Pairs Count graph has the first character on the X axis, the second character on the Y axis (you might have to scroll upward to see the entire character set), and a point on the graph represents one or more occurrences of an x-y pair.

The color is green if the count of this pair is low in relation to the other pairs, red if the pair appears much more often than would be expected in a random distribution, with two shades of orange in between. The tooltip of any meeting point lists the first character, the second character, and count of the pair. (For example ('35',12) indicates that there are 12 occurrences of the digit 3 followed by the digit 5.)

The Pairs Count table lists the first character, the second character, number of appearances that this pair was counted, and the expected number of appearances in a random distribution.

If the Singles Count results in one or two characters appearing much more than any other, try the Pairs Count. The repeated characters could be a sign of doubling.

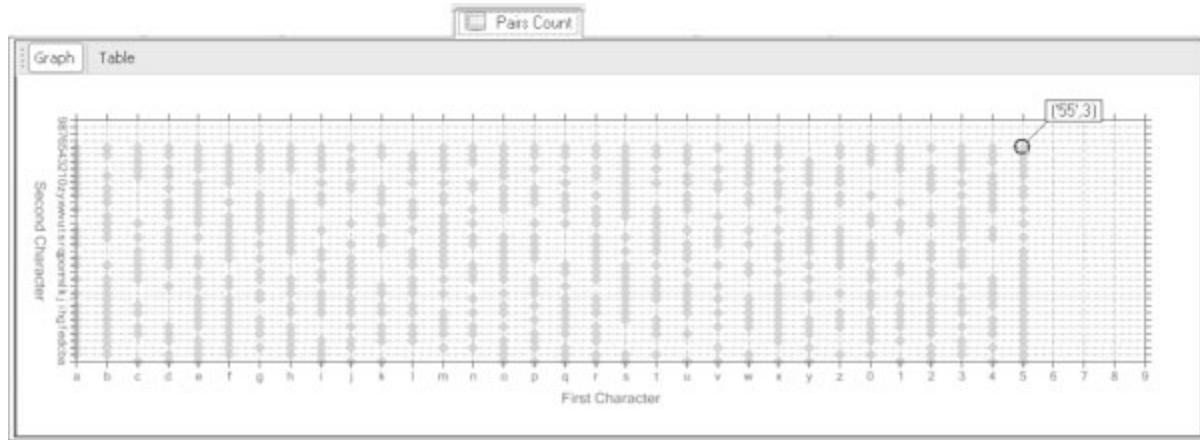


Figure 5. Pairs Count: Strong Session Token

In the sample above all pairs are colored green, which shows that the pairs counts are not above the expected number of appearances. Also notice that virtually every character is used, and that there is no discernible pattern.

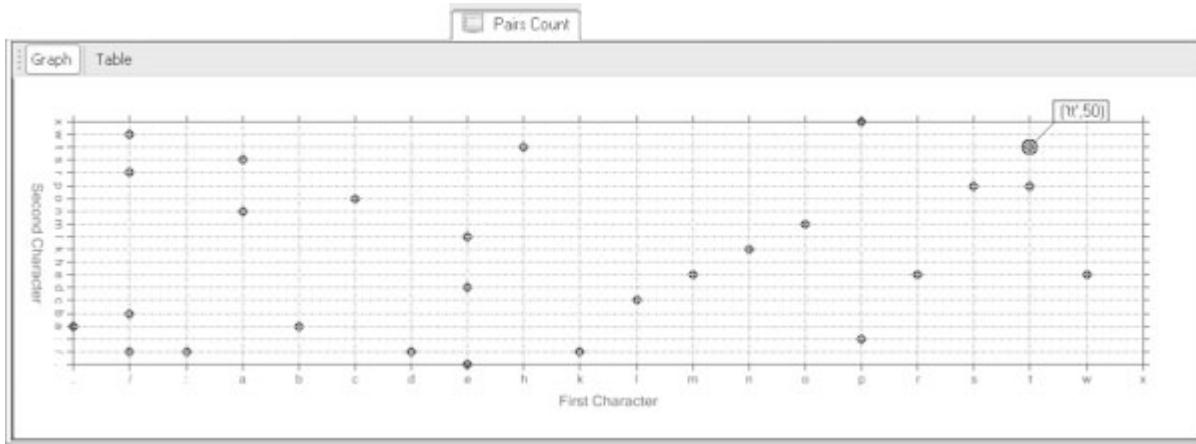


Figure 6. Pairs Count: Weak Session Token

In this sample, the red points on the graph show that the characters used may be predictable, because they appear much more often than would be expected in a random distribution.

			Pairs Count	
y	f	1	0.192901234...	***
y	s	2	0.192901234...	****
5	3	1	0.192901234...	**
3	i	1	0.192901234...	*
r	u	2	0.192901234...	****
u	p	2	0.192901234...	****
p	m	1	0.192901234...	*
m	r	2	0.192901234...	****
r	s	1	0.192901234...	*
u	x	2	0.192901234...	****
x	a	1	0.192901234...	*
a	t	1	0.192901234...	*
t	l	1	0.192901234...	*
l	w	1	0.192901234...	*
1	4	1	0.192901234...	*
4	0	1	0.192901234...	*
0	p	1	0.192901234...	*
p	4	1	0.192901234...	*

Figure 7. Pairs Count: Table View

Table view shows the results in five columns: first character, second character, number of occurrences, expected number of occurrences (in a random distribution), and a representation of the number of occurrences as a row of asterisks.

#### Correlation:

The Correlation chart is a scatter chart that can show certain non-random trends in session tokens from your application. (Note that the Correlation chart cannot usually cope with the data from a complete token, but with a subset of its characters.)

Let's say we took 50 samples of a particular token. Token Analyzer gives each sample an index number from 0 to 49, and these values are plotted on the X-axis. (The first token is 0, the second 1, and so on.) The X-axis therefore represents the progress of time.

Based on the constituent values and character set of the token (or selected subset), Token Analyzer uses an algorithm to assign a value to each token, and this value is plotted on the Y-axis.

#### Secure token:

Shows no correlation between its calculated numeric value and time (index value). The points on the graph will be scattered.

### **Weak token:**

May reveal a pattern. For example, if your token-creator algorithm provides the value of a session token subset using an incremental function, the points of the Correlation test will show an upward slope.

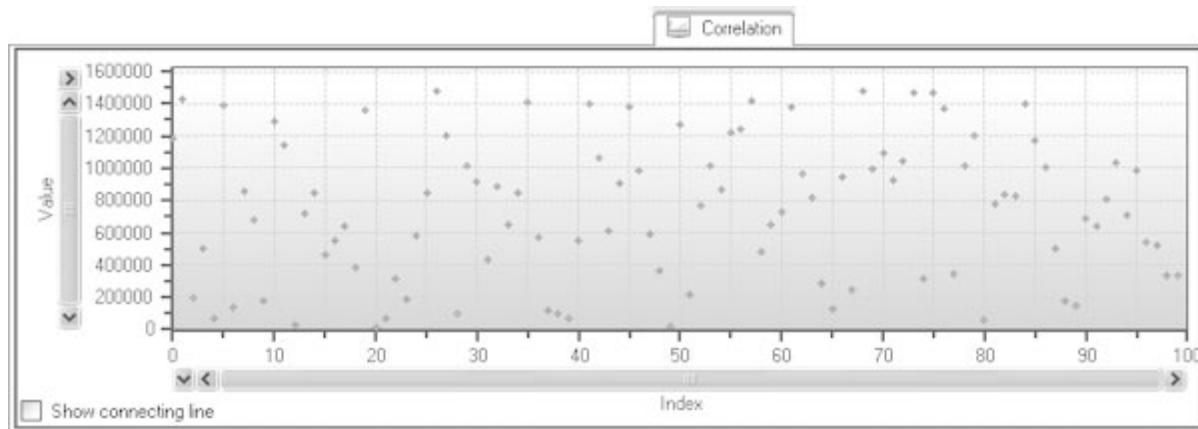


Figure 8. Secure Correlation

In a correlation analysis of a secure session token, random scattering of index-to-value correlation indicates secure unpredictability.

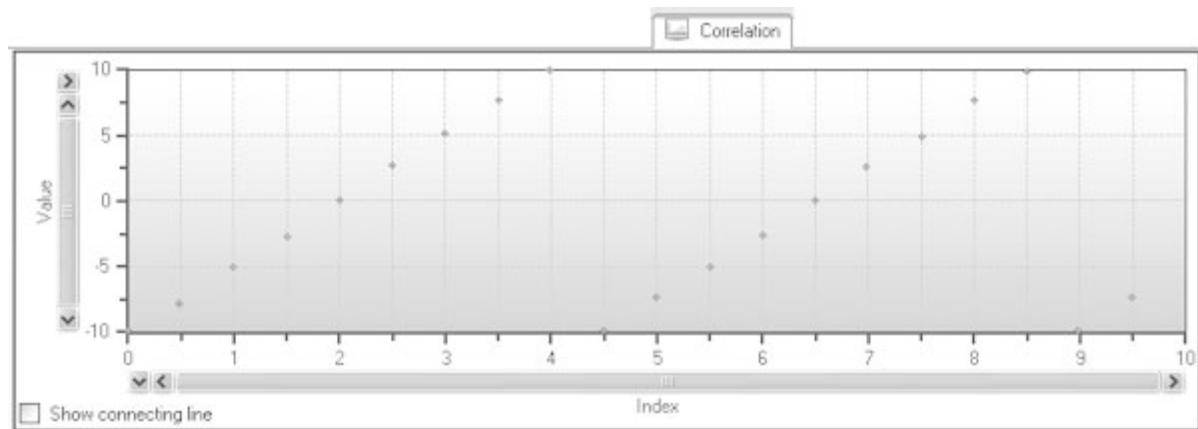


Figure 9. Correlation: Weak Token Values

Discernible patterns may reveal weak ("predictable") session tokens.

### **String Search:**

#### **About this task**

Some servers create unique session tokens by implanting user information and credentials in the session token. This makes for easier-to-guess session tokens, also making it possible for a malicious user to obtain confidential user information. Often servers attempt to overcome this by basing a transformation algorithm on the user data. If the transformation is not complex enough and can be guessed or reverse-engineered, a malicious user is still able to hijack session tokens and obtain confidential information that is supposedly hidden by the transformation.

- A secure session token will result with no dictionary or user input strings.
- A weak session token will result in strings that appear under easy-to-guess transformations.

The String Search test runs automatic searches for commonly found strings and for strings in the GET and POST requests. If the results show that some value-string was found to appear, other than common strings such as **user** or **passwd**, it is because Token Analyzer found name-value pairs in the GET or POST data and performed the transformation tests on this data.

If you suspect that your session tokens are displaying specific strings, you can run a search for any string you choose.

### Procedure

1. Enter the string in the **Manual Search** text box and then click **Go**.

If the string is found, it is listed in the left-hand panel, with the number of appearances. The right-hand panel lists the rows of samples that contain the string.

2. To see where in the session token this string is found, click **View in Grid**.

The **Session Data Grid** window opens. This window highlights the string in the samples where it is found, from within the selected subset.

### Results

The String Search test also searches for strings based on reverse transformations of different combinations of Unicode, MD5, Ascii, SHA1, Hex, and Base64; in either the user input or the session data, or both.

For example, a result of the String Search test might display the following:

**The following transformations were done on the input string:**

1. The string was transformed to bytes, assuming Ascii encoding.
2. The bytes were transformed back to string assuming Hex encoding.

### What to do next

To see the list of transformation combinations that is used, click **String Transformations**.

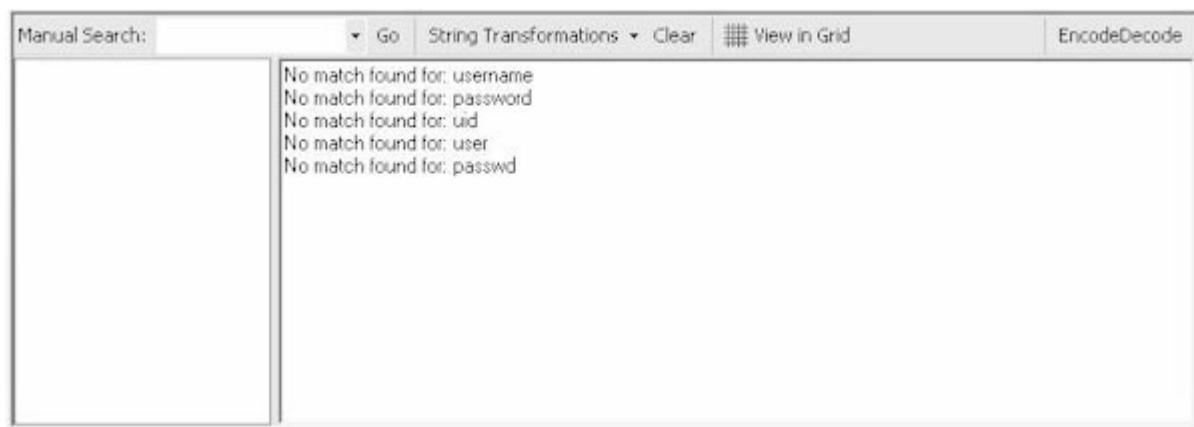


Figure 10. Secure String Search

A secure session token does not contain any of the commonly used strings that malicious users exploit.

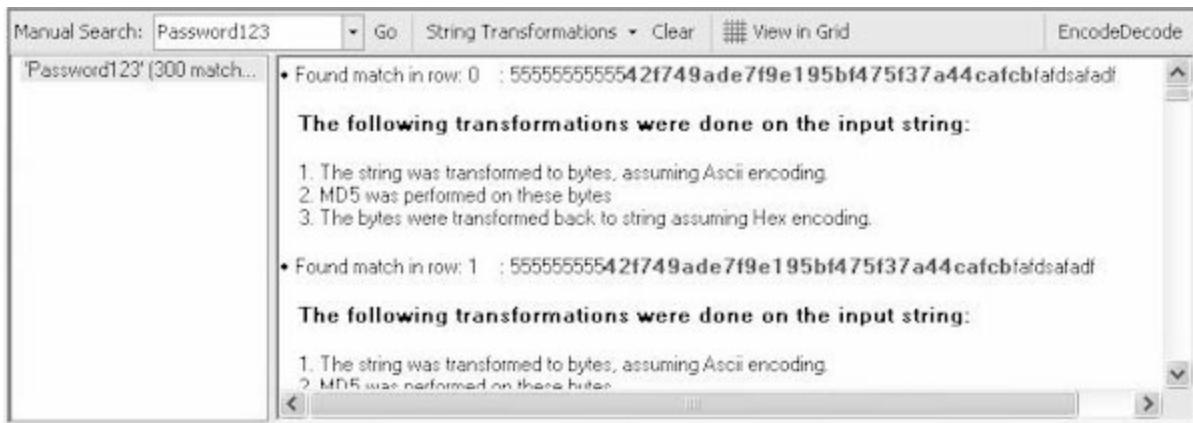


Figure 11. Strings Found in Weak Session Token

The session token shown in this sample shows that Token Analyzer was able to transform the user input (string to bytes using Ascii, MD5, and then bytes to string using Hex) from an apparently random set of characters into the original string that contained the user's password.

**Note:** You can access the IBM Security EncodeDecode PowerTool from this test, to run more transformation samples, in various formats: click **EncodeDecode**.

## Working with results

After reviewing the results, you may decide to:

- “Updating the sample”
- “Taking a larger sample”
- “Analyzing a subset of the results”

### Updating the sample:

#### About this task

If the sample size proves to be too small to produce significant results, you can take a larger sample of the same URL without reopening the New Session window.

#### Procedure

##### Click **Update**.

A new set of session data samples is taken, and the previous ones discarded.

### Taking a larger sample:

#### About this task

If the sample size proves to be too small to produce significant results, you can take a larger sample of the same URL without reopening the New Session window.

#### Procedure

1. In the **Update Samples** area, type in the number of samples to be taken
2. Click **Update**.

A new set of session data samples is taken, and the previous ones discarded.

### Analyzing a subset of the results:

## About this task

Sometimes results may not show clear “predictability”, but may nonetheless indicate paths for further enquiry.

To give just one or countless possible examples: Let’s say that in the Column Chi-Square chart, positions 8, 16 and 24 are within the blue “safe” limit, but slightly higher than the other columns. This may indicate some problem in the way these particular positions are generated, that could be utilized by a malicious user. To test this you could select just these positions, and analyze them.

## Procedure

1. Select the columns you want:
  - To select consecutive columns, click-and-drag the mouse across the columns you want.
  - To select non-consecutive columns, click the required columns while holding down the **Ctrl** key.
2. Click **Analyze**.  
A new set of session data samples is taken, and the previous ones discarded.

---

## Generic Service Client

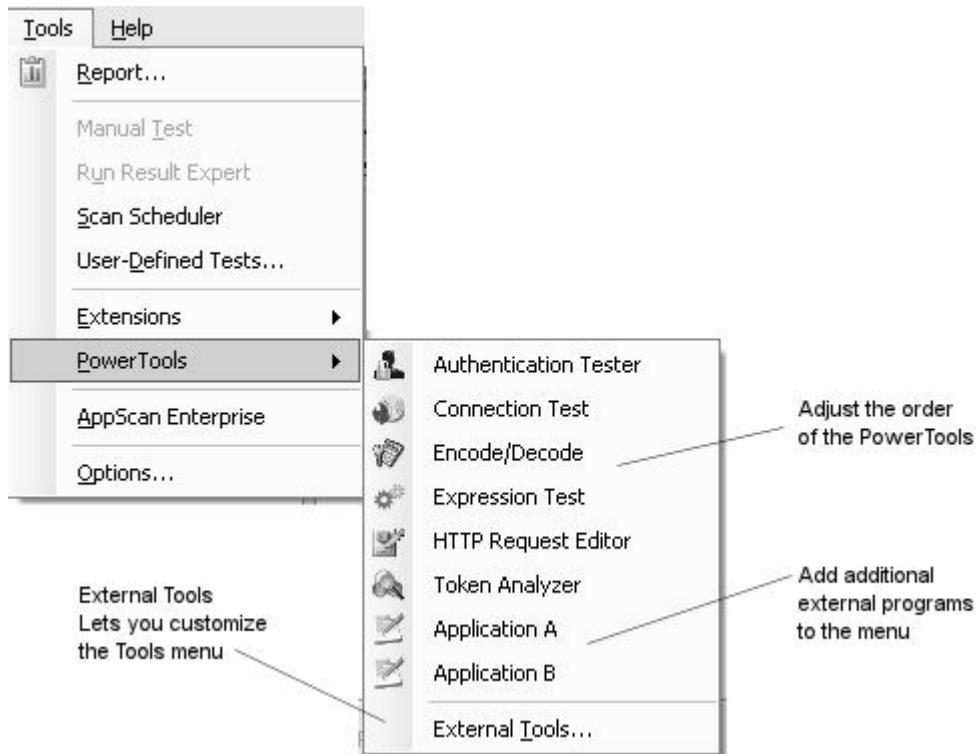
Generic Service Client (GSC) provides a simple interface that displays the services available, and lets you input parameters and view the results. Use it to explore your SOAP web service manually, so that AppScan can use your input to create appropriate tests.

See

## Customizing the Tools menu

You can customize the Tools menu to:

- “Adjust the order of the PowerTools”
- “Add programs to the Tools menu”



### Adjust the order of the PowerTools

#### Procedure

1. Click **Tools > PowerTools > External Tools**  
The External Tools dialog box opens.
2. Select a menu item from the list, and click **Move Up** or **Move Down** as required.

### Add programs to the Tools menu

#### About this task

You can add a link to an external program that you frequently use a particular program while running AppScan, you can add a link to the Tools menu so that you can open it from within AppScan.

#### Procedure

1. Click **Tools > PowerTools > External Tools**  
The External Tools dialog box opens.
2. Click **Add**  
The Create a New External Tool dialog box opens.
3. In the **Title** field, type in the name you want to appear on the **Tools** menu.
4. Click the **Browse** button (...), locate the EXE file for the program, and click **Open**.
5. Click **OK**.

- The title and path of the .exe file are added to the External Tools list.
6. To adjust its position in the menu, select the new item and click **Move Up** or **Move Down** as required.
- 

## Extensions

With AppScan it is easy to write and integrate extensions. Extensions are add-ons that add functionality to AppScan. They can be anything from a small email notification utility, to a full blown vulnerability exploit tool. Extensions enable users to customize AppScan to their own needs. They are written using AppScan's SDK and extensions framework.

Further information on how extensions can be used with AppScan, and where to download/host extensions, is available at: [http://www.ibm.com/developerworks/rational/downloads/08/appscan\\_ext\\_framework/](http://www.ibm.com/developerworks/rational/downloads/08/appscan_ext_framework/)

The SDK online help file **AppScanSDK.chm** can be found in the main AppScan documentation folder.

AppScan comes with the Pyscan extension, and some others, built-in. It is easy to integrate and manage additional extensions using the Extension Manager.

- “Extension Manager”
- “Pyscan” on page 268
- “Explore Optimization module” on page 269

Extensions are also used to enable integration with the following defect-tracking systems:

- “Rational ClearQuest” on page 279
- “HP Quality Center” on page 279

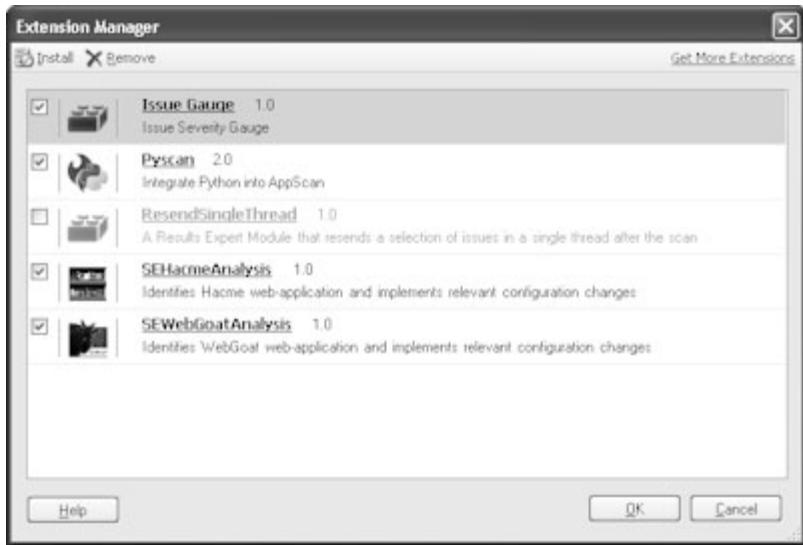
## Extension Manager

Extension Manager lets you install/uninstall, activate/deactivate extensions for use with AppScan

Extension Manager (**Tools > Extensions > Extension Manager**) lists all extensions currently installed.

- You can download and install extensions that are available from IBM Security, as well as install your own.
- You can activate and deactivate extensions that are installed.

If an extension is installed but not activated the entry appears grayed out.



Option	Explanation
Install	Click to install extensions that you have already saved to your computer.
Remove	Click to uninstall one of the listed extensions.
Get More Extensions	Click to browse to the AppScan eXtensions Framework site, to select and download extensions.
check box next to installed extension	<p>Select/deselect the check box to activate/deactivate the extension. Note that this change only takes effect the next time you open AppScan.</p> <p>When a new extension is installed it is automatically activated (and this check box selected), unless there is a reason AppScan was unable to do this (such as an incompatible version).</p>

## Pyscan

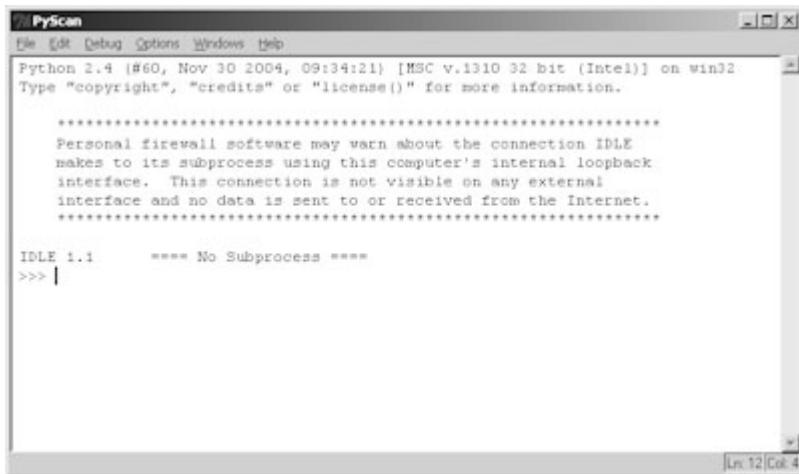
### About this task

Pyscan is a Python extension that enables you to control AppScan using the Python interface. Pyscan is installed and activated during the standard AppScan installation.

### Procedure

On the **Tools** toolbar, click **Extensions > Pyscan**.

The Pyscan interface opens.



## Explore Optimization module

This extension helps improve scan efficiency by eliminating unnecessary URLs from the Explore results.

The extension can particularly improve performance in cases where URL rewriting is used by the site, inflating the Explore stage of the scan with duplicate URLs that appear different due only to built-in parameters.

The Explore Optimization module identifies URLs that are rewritten to include parameters. For example, if URLs such as those below occur a large number of times, AppScan would create a folder for each one where in fact these are actually parameters that have been rewritten into the URL for convenience.

```
http://mysite.com/2010/10/01/  
http://mysite.com/2010/10/02/  
http://mysite.com/2010/11/01/  
http://mysite.com/2010/11/02/  
http://mysite.com/2010/12/01/  
http://mysite.com/2010/12/02/
```

A case like this could result in thousands of unnecessary URLs, or a never-ending Explore stage.

When it encounters a large number of URLs that seem related, the module analyzes them. It then creates a custom parameter (in this case it would be / [year] / [month] / [day] ), clears the Explore results, and runs a new Explore stage, which may have far fewer URLs to be tested in the Test stage. This can greatly improve performance.

**Note:** An additional functionality of this module, that can be useful in maximizing scan efficiency, is **Run navigational parameter detection** (see “Configuring Explore Optimization” on page 271 for details).

### Running the module automatically

It is suggested to run the module during scans whenever the number of links discovered in the Explore stage (including unvisited links) reaches the configured threshold. This threshold ("Minimum links to start module"), is generally at least 1,000.

**Note:** If you start a scan from the command line interface (CLI), Explore Optimization does *not* run automatically even if configured.

**Note:** If your scan configuration includes Manual Explore data or a multi-step operation, Explore Optimization does *not* run automatically even if configured, though you can run it manually (Tools > Extensions > Explore Optimization Module > Run).

### To change the configuration:

1. Click **Tools > Extensions > Explore Optimization: Configure**.  
Explore Optimization Configuration dialog box opens.
2. Select the **Run Explore Optimizer automatically during scans** check box.
3. Click **OK**.

### Running the module manually

You can run the module manually on any Explore results, even if the number of URLs discovered is *less* than the configured threshold ("Minimum links to start module"). However the module is most useful when the number of URLs is at least several hundred.

### To run Explore Optimization manually:

- Click **Tools > Extensions > Explore Optimization: Run**.

The module begins to Explore, starting at the configured Starting URL. When the Explore stage is complete it analyzes the results. If optimizations are discovered, it will give you the option of saving the first set of results before clearing them and re-Exploring.

*See also:*

"Redundancy tuning" on page 69

"Redundancy tuning defaults" on page 71

## Working with Explore Optimization

This section describes running a scan with Explore Optimization active.

### About this task

If your site rewrites parameters into its URLs, or if initial attempts to explore the site, with Explore Optimization disabled, result in very large numbers of URLs or unending scans, the Explore Optimization module can help to reduce the scan to manageable proportions. In addition, activating the **Run navigational parameter detection** option may help.

### Procedure

1. Configure the Starting URL and any other necessary settings as for a regular scan.
2. Open **Tools > Extensions > Explore Optimization: Configure**, and select the **Run Explore Optimizer automatically during scans** check box.

**Note:** If your scan configuration includes Manual Explore data or a multi-step operation, Explore Optimization does *not* run automatically even if configured, though you can run it manually (**Tools > Extensions > Explore Optimization Module > Run**).

3. In the Scan configuration area, optionally change the **Run navigational parameter detection** setting to **True**.
4. Click **OK**.
5. Start a scan (**Scan > Full Scan**).

If the number of URLs found in the Explore stage (including unexplored URLs) reaches the threshold defined in **Minimum links to start module** (default: 1000), the Explore stage is paused and the Explore Optimization module starts a *phase*, consisting of two (main) stages:

#### Identify navigational parameters (if configured)

The module looks for navigational parameters, using the defined Names and Values (configured in **Tools > Extensions > Explore Optimization: Configure**). If it succeeds in identifying navigational parameters, it then:

- a. Defines them in the list of parameters (Scan Configuration > Parameters and Cookies > main tab)
- b. Sets their Redundancy Tuning to the strictest level
- c. Decreases the *default* Redundancy Tuning (for non-navigational parameters) to a lower level (see “Redundancy tuning defaults” on page 71)

#### **Identify URL rewriting**

The module looks for parameters written into the URL. If it finds such parameters, it defines them in the list of custom parameters (Scan Configuration > Parameters and Cookies > Advanced: Custom Parameters tab).

6. At the end of this *phase* of Explore Optimization:
  - If any configuration changes have been made, the existing Explore data is cleared and a new Explore stage is run. (If you started the module manually, you are given the option of saving the current data before the new Explore stage is run.)
  - If no configuration changes have been made, a new *phase* of Explore Optimization is run, using a higher threshold (not user-configurable), in an attempt to gather enough data to identify parameters and reduce the Explore data to a reasonable size.
7. After the module has successfully run (with one or more phases, and one or more Re-Explore stages), the scan resumes and finishes.
8. When the scan is complete, review the results, looking for the following indications that it has been successful:
  - The status bar, at the bottom left of the screen, indicates how many tests were created and how many sent. All tests should have been sent.
  - The application tree should be complete, indicating that all the important parts of the site were visited.
  - Review the navigational parameters that were added to see that all important parameters were fully tracked.
  - Review the custom parameters that were added to see that they correctly represent the way the site writes its parameters into its URLs.

#### **Configuring Explore Optimization**

This dialog box is used to activate and configure the Explore Optimization module.

Apart from activating the module, in most cases you do not need to change any other configurations. Some should not be changed at all without guidance from Support, as indicated below.

This dialog box is opened from **Tools > Extensions > Explore Optimization: Configure**.

**Note:** If you make changes to these settings and then create a new scan, all settings will revert to their defaults, except the check box setting.

Name	Description
Checkbox	

Name	Description
Run Explore Optimizer automatically during scans	<p>When selected the module runs automatically when you run either Explore Only or a Full Scan, whenever the "Minimum links to start" limit (below) is reached.</p> <p><b>Important:</b> This setting applies to <i>all</i> scans. Other settings in this dialog box apply to the current scan only.</p> <p>Default: Cleared</p> <p><b>Note:</b> If you start a scan from the command line interface (CLI), Explore Optimization does <i>not</i> run automatically even if configured.</p> <p><b>Note:</b> If your scan configuration includes Manual Explore data or a multi-step operation, Explore Optimization does <i>not</i> run automatically even if configured, though you can run it manually (Tools &gt; Extensions &gt; Explore Optimization Module &gt; Run).</p>
<i>Scan Configuration</i>	
Anchor rewrite rules to the right	<p>Consider URLs such as these:</p> <pre>http://...php/1/index http://...php/2/index http://...php/3/index</pre> <p>When the module creates the custom parameter:</p> <p>If this is set to False, the parameter created will be:</p> <pre>php/([^.]+)</pre> <p>If this is set to True, the parameter created will be:</p> <pre>php/([^.]+)/index</pre> <p>Default: False</p>
Confidence margin percentage	Change this only if instructed by Support.
Delimiters used in URL rewriting	Add any custom delimiters that your application uses.
Maximum depth for start of line pairs	Change this only if instructed by Support.
Maximum phase time (mins)	Maximum time for a phase of Explore Optimization to run.
Merge rewrite rules to the left	Change this only if instructed by Support.
Minimum links to start module	<p>Minimum number of links in Explore stage data (including unvisited URLs) needed for the Explore Optimization module to start automatically, if the Run Explore Optimizer automatically during scans check box is selected.</p> <p><b>Note:</b> Even if this setting is changed, the notice suggesting that you activate the module appears when 1,000 links are reached in the Explore stage.</p> <p>Default: 1,000</p>
Name/Value pair delimiters	Specify any custom delimiters used by your application.
Navigational parameter names	Partial matches for navigational parameter names, separated by single spaces.
Navigational parameter values	Partial matches for navigational parameter value patterns, separated by single spaces.
Remove unused delimiters	<p>Change this only if instructed by Support.</p> <p>Default: True</p>

Name	Description
Run navigational parameter detection	<p>When True, the module attempts to identify navigational parameters by name or value. It sets the Redundancy Tuning configuration for these parameters to the strictest level, <i>and reduces the default setting that is applied to all other parameters</i>. This enables AppScan to test navigational parameters <i>more</i> thoroughly than before, while safely treating all non-navigational parameters <i>less</i> thoroughly.</p> <p>Setting this value to True can greatly improve both accuracy and performance, however you should carefully review the results to make sure that scan coverage was not affected.</p> <p>Default: False</p>
Switch complexity limit	<p>If a specific folder contains more subfolders than this limit, AppScan will assume that the subfolders are dynamic parameter values and do not require individual scanning.</p> <p><b>Note:</b> The value entered here is also used as a threshold value for running Explore Optimization manually. If less links have been discovered than the amount entered here, the module will not run.</p> <p>Default: 20</p>

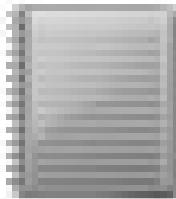
## Logs

Logs can help you troubleshooting.

### Scan Log

This log lists the actions performed by AppScan during the current scan.

When a scan runs the Scan Log is updated live, so you can see what AppScan is doing at any moment.



- On the toolbar, click the **Scan Log** icon

Time	Event
9/11/2006 4:37:37 PM	Scan Configuration modified
9/11/2006 4:37:59 PM	Scan Configuration modified
9/11/2006 4:38:08 PM	Session Identifier detected; name = ASP.NET_SessionId; ...
9/11/2006 4:40:30 PM	Scan Configuration modified
9/11/2006 4:40:31 PM	Scan Configuration modified
9/11/2006 4:40:32 PM	Scan Configuration modified
9/11/2006 4:44:52 PM	Scan Configuration modified
9/11/2006 4:45:25 PM	Starting Explore
9/11/2006 4:45:26 PM	Crawling
9/11/2006 4:45:31 PM	Performing login
9/11/2006 4:45:31 PM	Visited URL: http://bern/
9/11/2006 4:45:31 PM	Visited URL: http://bern/bank/default.aspx
9/11/2006 4:45:31 PM	Skipping URL (due to extension): http://bern/bank/image...
9/11/2006 4:45:32 PM	Skipping URL (due to extension): http://bern/bank/image...

The scan log is saved as part of the scan. When a saved scan is loaded, the existing scan log loads too, and data is added to it as scanning continues.

**Note:** The Scan Log window does not show previous data saved in the scan log. This data can be found only in the actual log file. To view it, open the scan file (.SCAN) using a program that opens compressed files, look for ScanLog.log, and open it using a text viewer.

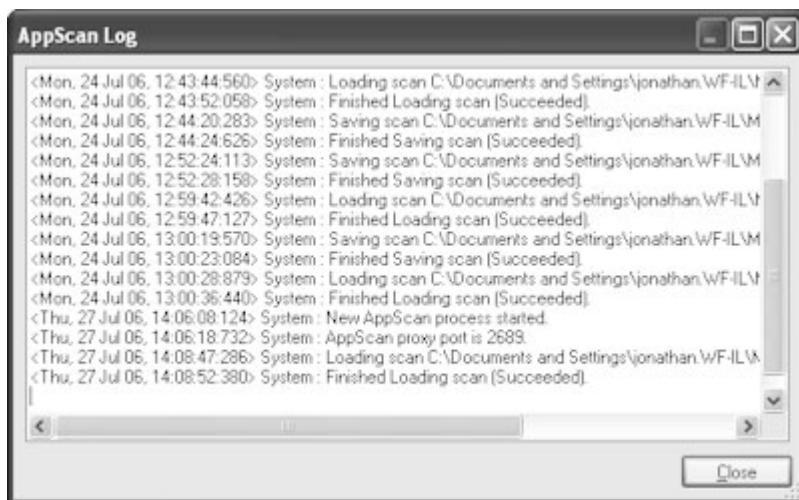
You can customize the scan log, defining exactly which items are included in it, and control their color in the Scan Log window (see “Scan Options tab” on page 217).

Where relevant, explanations of the messages, and suggested user responses are described in “Scan Log messages” on page 305

## AppScan Log

This log lists errors, connections and AppScan system messages (general application events and warnings).

- To view the AppScan Log within AppScan, click **Help > AppScan Log**.



- To see where this log is saved to, click **Tools > Options > General tab > Log files folder**.

In Windows 7 the default location is:

...\\AppData\\Roaming\\IBM\\AppScan Standard\\Logs

Where relevant, explanations of the messages, and suggested user responses are described in “AppScan Log messages” on page 314

## Update Log

This log lists all updates to the program since installation.

- To view the Update log click **Help > Update Log**.



- To see where this log is saved to, click **Tools > Options > General tab > Log files folder**.  
In Windows 7 the default location is:  
...\\AppData\\Roaming\\IBM\\AppScan Standard\\Logs

*See also:*

"Updates" on page 12

## Traffic Log

This log lists the requests and responses between AppScan and the site during scanning, and can be useful when troubleshooting.

The Traffic Log lists all requests responses from the application during the scan. By default AppScan does not save a traffic log, but you can enable it if needed by Support to help give you assistance.

Turning on the log may affect performance, so it should only be enabled if you need it.

- To enable the Traffic Log, click **Tools > Options**, and select **Enable request/response logging**.
- To see where this log is saved to, click **Tools > Options > General tab > Log files folder**.  
In Windows 7 the default location is:  
...\\AppData\\Roaming\\IBM\\AppScan Standard\\Logs

---

## Searching Results

You can filter the Result List in any of the views, for specific data.

### Procedure

1. On the toolbar click **Find**, or press **Ctrl + F**.  
The **Find** bar appears at the bottom of the screen.
2. On the **Application Tree**, select a node.
  - If you select the **My Application** node, Find searches through all the results.
  - If you select a node within the tree, Find searches through the selected node and its sub-nodes.
3. In the **Look for** text box of the **Find** bar, enter a string or partial string.
4. In the **Show** list, click a data type.
5. Click **Find Now** or press **Enter**.

Results appear in the **Result List**, overwriting the list that was previously displayed, and the counters update to show the new number of items in each category. Tabs without results are grayed out.

If you enter another **Find** string and click **Find Now** again, the new search is done on the selected node in the **Application Tree**, not on the displayed results of the previous search.

**Note:** To revert to the full, unfiltered Result List, click **Clear**.

---

## Chapter 12. Integrations

AppScan offers integrations with the following applications:

---

### AppScan Enterprise

This section describes ways AppScan Standard and Enterprise editions can interact.

AppScan Enterprise edition provides centralized scanning and user-access controls, remediation capabilities, executive dashboards, compliance reporting and seamless integration with AppScan Standard. If your organization uses AppScan Enterprise, you can:

- Import AppScan Enterprise user permissions to use *in addition* to those allowed by the local AppScan Standard license.
- Export AppScan Standard scan results to work with in AppScan Enterprise.

### Importing AppScan Enterprise license permissions

Configure AppScan Standard to scan sites allowed by your AppScan Enterprise license.

#### About this task

If your organization has an AppScan Enterprise license that allows scanning additional sites to those allowed by your local AppScan Standard license, you can import these permissions to use on your local machine *in addition* to your existing license. You will then be able to scan any URL that is allowed by one or both of the licenses.

**Note:** This option is available only when a full AppScan Standard license (not a demo license) is loaded.

#### Procedure

1. Click **Help > License**.  
The License dialog-box opens.
2. Click **Add AppScan Enterprise License**.  
The AppScan Enterprise dialog-box opens.
3. Select the **AppScan Enterprise** check box.  
The fields become active.
4. Select the check box, and complete the User Name, Password, Domain and URL for the AppScan Enterprise server.
5. (Optional:) Verify that you are connected to the network, and click the **Test these settings** button.  
AppScan establishes contact with the AppScan Enterprise Server and the organization's license is displayed in the lower pane.

**Note:** In the event that the local client fails to make contact with the Enterprise server, you are notified that scanning is restricted to IPs allowed by the local license.

6. Click **OK**.  
The AppScan Enterprise permissions are loaded to the AppScan License, in addition to its own permissions.

### Publishing to AppScan Enterprise

You can publish AppScan Standard results to AppScan Enterprise.

## About this task

Scan results can be exported to AppScan Enterprise, to work with them there.

### Procedure

1. Open the scan whose results you want to export.
2. Click **File > Export > Publish to AppScan Enterprise**.  
The Publish Results dialog-box opens.
3. Assign a Job Name, and optionally select an AppScan Enterprise Folder, and Application.

**Note:** Folder selection is supported from AppScan Enterprise version 8.7, and Application selection from AppScan Enterprise version 9.0. If you do not select a folder the default AppScan Enterprise folder is used.

**Note:** The Select Application dialog box includes a **Create a new application on server** option if your permissions allow this.

4. If you have not yet configured your machine for connecting to AppScan Enterprise, click **Connection Settings** to open the lower part of the dialog box, and fill in your settings, then verify that you are connected to the network, and click **Test these settings**.  
AppScan establishes contact with the AppScan Enterprise Server and displays the retrieved license in the lower pane.
5. Click **Publish**.  
When the process is complete a green success message appears in the dialog box. The job can now be opened and worked with in AppScan Enterprise.

## Creating a job in AppScan Enterprise

You can create an AppScan Enterprise job using your AppScan Standard configuration.

### About this task

Scan configurations can be exported to AppScan Enterprise as new jobs to work with them there.

**Note:** Security AppScan Enterprise Version 9.0 or later is required.

### Procedure

1. Open the scan whose configuration you want to use.
2. Click **File > Export > Create Job in AppScan Enterprise**.  
The Create Job dialog-box opens.
3. Assign a Job Name, and select the AppScan Enterprise Folder, Application, and Test Policy.

**Note:** The Select Application dialog box includes a **Create a new application on server** option if your permissions allow this.

4. If you have not yet configured your machine for connecting to AppScan Enterprise, click **Connection Settings** to open the lower part of the dialog box, and fill in your settings, then verify that you are connected to the network, and click **Test these settings**.  
AppScan establishes contact with the AppScan Enterprise Server and displays the retrieved license in the lower pane.
5. Click **Create**.  
When the process is complete a green success message appears in the dialog box. The results can now be opened and worked with in AppScan Enterprise.

---

## Rational ClearQuest

 Support for logging defects found by AppScan directly to Rational ClearQuest® is deprecated as of this version.

### About this task

Using extensions supplied with AppScan, you can create defects within your Rational ClearQuest defect-tracking system from issues in the scan results. The extension is installed by default. Supported versions are:

- Rational ClearQuest 7.1.1, 7.1.2, 8.0

**To add a defect to your tracking system:**

### Procedure

1. You must have the ClearQuest client installed on your machine.
2. In the Result List, right-click on the issue from which you want to create the defect, and select **Add Defect to Rational ClearQuest**.

**Note:** If you do not have the right-click option, the extension has been deactivated. (See “Extension Manager” on page 267.)

3. (First defect only, each time you open AppScan:) The Login dialog box opens to log in to the defect-tracking system. Verify the details, if necessary click **Authenticate** to update Domain and Project options - then click **Log in**.  
AppScan logs in to your defect-tracking system, and the Defect Details dialog box opens, with default values filled in some of its fields.
4. Fill in the Details fields you need (fields in red are mandatory).
5. Review and if necessary edit the Description text.
6. Review and if necessary edit or remove the attachments, which include the advisory and fix recommendation for the issue. You can also *add* attachments.
7. You can optionally switch to a different user using the Switch User link.
8. Click **Add Defect**.

The defect is added to your defect-tracking system.

*See also:*

“Extension Manager” on page 267

“HP Quality Center”

---

## HP Quality Center

 Support for logging defects found by AppScan directly to HP Quality Center is deprecated as of this version.

### About this task

Using extensions supplied with AppScan, you can create defects within your HP Quality Center defect-tracking system from issues in the scan results. These extensions are installed by default. Supported versions are:

- HP Quality Center 9.2, 10

**To add a defect to your tracking system:**

## Procedure

1. (First use only:) Log in to your Quality Center account. Any required ActiveX components are downloaded automatically.
2. In the Result List, right-click on the issue from which you want to create the defect, and select **Add Defect to HP Quality Center**.

**Note:** If you do not have the right-click option, the extension has been deactivated. (See “Extension Manager” on page 267.)

3. (First defect only, each time you open AppScan:) The Login dialog box opens to log in to the defect-tracking system. Verify the details, if necessary click **Authenticate** to update Domain and Project options - then click **Log in**.  
AppScan logs in to your defect-tracking system, and the Defect Details dialog box opens, with default values filled in some of its fields.
4. Fill in the Details fields you need (fields in red are mandatory).
5. Review and if necessary edit the Description text.
6. Review and if necessary edit or remove the attachments, which include the advisory and fix recommendation for the issue. You can also *add* attachments.
7. You can optionally switch to a different user using the Switch User link.
8. Click **Add Defect**.

The defect is added to your defect-tracking system.

*See also:*

“Extension Manager” on page 267

“Rational ClearQuest” on page 279

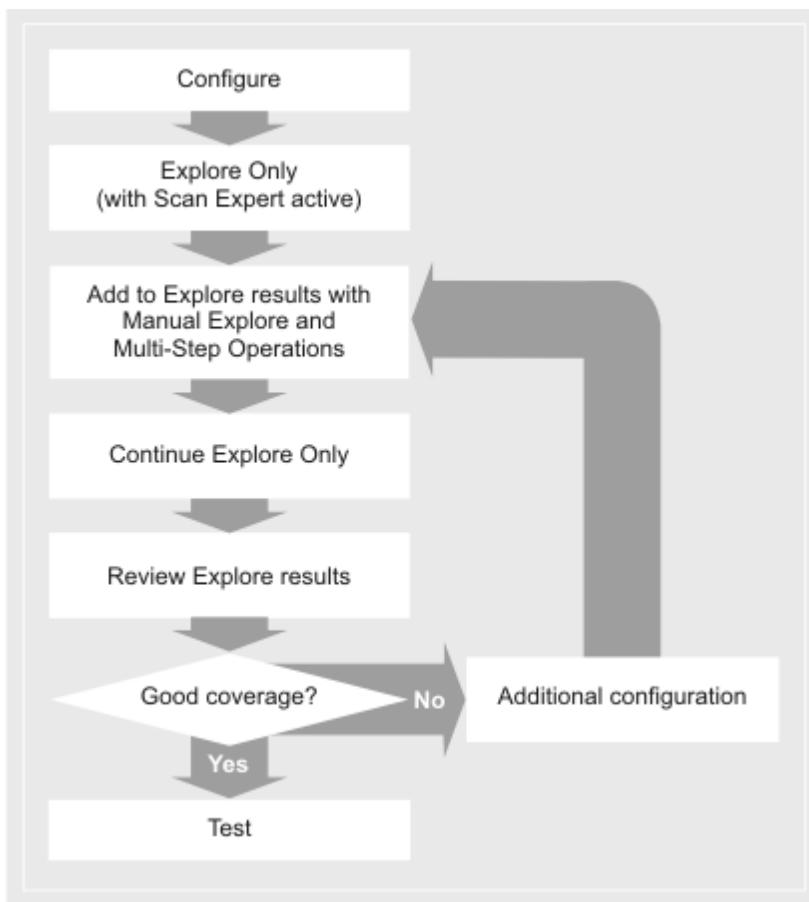
# Chapter 13. Best practices

This section contains some best practices and use cases for advanced users.

## Workflow for advanced users

This workflow can help users with experience in the field of web security achieve a more thorough scan.

The success of the Test stage, and therefore of the scan itself, depends on the coverage achieved during the Explore stage. If the Explore stage misses important parts of the application logic, the Test stage will not be able to reveal important vulnerabilities that might exist. Following this workflow can help you improve Explore stage coverage.



Task	Description
1. Initial Configuration	<p>Using the wizard or the Scan Configuration dialog box:</p> <ol style="list-style-type: none"> <li>1. Define the starting URL</li> <li>2. Record the login procedure</li> <li>3. Validate the in-session pattern and if needed choose a new pattern</li> <li>4. If your site has an account lockout feature, either disable the feature, or configure AppScan not to test login pages - otherwise AppScan will get locked out of the site during the Test stage and be unable to proceed.</li> </ol> <p>For a fuller description of this step, see "Initial Configuration" on page 283</p>
2. Explore Only	<p>Run an initial Automatic Explore:</p>  <ol style="list-style-type: none"> <li>1. On the toolbar, click  &gt; <b>Explore Only</b>, and let the new Explore stage complete. AppScan will explore the site but not test it yet. At the start of the Explore stage Scan Expert will run, and may suggest some changes to the configuration. The default setting lets Scan Expert make only those changes that can be applied automatically. <b>Note:</b> If your site uses URL rewriting, run Explore Optimization (<b>Tools &gt; Extensions &gt; Explore Optimization Module: Run</b>), and if recommended by the extension, run the Automatic Explore stage again (<b>Scan &gt; Re-Explore</b>).</li> <li>2. If the Explore terminates early due to AppScan being out-of-session, re-record and reconfigure the login procedure, paying special attention to In-Session Detection and Session ID Tracking.</li> </ol> <p>For a fuller description of this step, see "Initial Automatic Explore" on page 284</p>
3. Improve site coverage manually using the browser	<p>Add URLs which Automatic Explore missed:</p> <ol style="list-style-type: none"> <li>1. <b>Manual Explore:</b> Use Manual Explore to add individual pages, such as those that require specific input. <b>Note:</b> In rare cases where the built-in browser is unable to browse the application, you can configure AppScan to use a different browser.</li> <li>2. <b>Multi-Step Operations:</b> If parts of the site can be reached only by clicking links <i>in a specific order</i>, record one or more Multi-Step Operations.</li> </ol> <p>For a fuller description of this step, see "Improve site coverage manually" on page 284</p>
4. Continue Explore Only	<p>With the new data you have provided through the Manual Explore, Automatic Explore will probably be able to explore the application more thoroughly.</p>  <p><b>Note:</b> Click  &gt; <b>Continue Automatic Explore</b> (or <b>Scan &gt; Explore Only</b>), to preserve the initial Explore results and the Manual Explore data. Do not click <b>Re-Scan &gt; Re-Explore</b>, as this will delete the existing data.</p>
5. Evaluate Explore results	<p>Review the results so far, to see if the application logic has been well covered by the exploring done so far. <b>Note:</b> If you make any configuration changes, you should run Automatic Explore again (<b>Scan &gt; Re-Explore</b>).</p> <p>For a fuller description of this step, see "Evaluate Explore results" on page 286</p>
6. (If needed) Additional configuration	<p>There are some additional configuration options that you should consider if application coverage so far is not sufficient.</p> <p>For a fuller description of this step, see "Additional configuration" on page 288</p>
7. Test stage	<p>Click <b>Test Only</b> to proceed with the Test stage, completing the scan.</p>

# Initial Configuration

## About this task

Perform these basic configuration steps before attempting to start exploring the site, whether manually or automatically.

### Procedure

1. **Define and verify the starting URL for the scan.**
  - a. Click **File > New** and create a new web application scan using the wizard (or configure using the configuration dialog box, **Scan Configuration > URL and Servers view**).
  - b. Type in the Starting URL for the scan.
  - c. If your application is case-sensitive, make sure that the Case Sensitive Path check box is selected.  
In general, applications that run on Linux based operating systems tend to be case-sensitive, while those that run on Microsoft Windows are not case-sensitive. Java based applications are an exception, and tend to be case-sensitive on any operating system.
  - d. Click the **View in Browser** icon, next to the URL field, and verify that the expected page appears in the AppScan browser.
2. **Record the login procedure.** This enables AppScan to log in to the application both to start the scan and whenever it gets logged out during the scan.
  - a. In Step 2 of the wizard, (or in **Scan Configuration > Login Management view**), click  Record to start recording your actions. The browser opens at the starting URL you defined previously.
  - b. Perform all the steps a user needs to do in order to log in to the application.
  - c. Look for some indication on the page that confirms you are logged in, such as "Welcome [Username]", or a "Log Out" link, *that would only be seen by a logged-in user*.
  - d. Close the browser and look for the green key icon  to confirm that an in-session pattern has been identified.

If the icon is red , an in-session pattern has *not* been detected, and you must define it manually (see “[Changing the in-session detection pattern](#)” on page 54).

**Note:** Although, in general, the first URL whose response includes the in-session pattern should be the “In-Session URL”, and this is the URL that is selected automatically, sometimes you can improve performance by selecting a later URL (see “[Optimizing In-Session Detection](#)” on page 54).

3. **Validate the in-session pattern.** The in-session pattern is a regular expression that matches a pattern or string on the page that appears to the user after successful login, such as "Welcome [Username]", or a "Log Out" link. Even though the icon is green, you should verify this pattern.
  - a. In Step 2 of the wizard, select **I want to configure In-session detection options**, and then click **Next** (or go to **Scan Configuration > Login Management > Details view**).  
The Login sequence is shown.
  - b. Double-click on the page marked "In Session" to open it in the browser.
  - c. In the browser, click on the Request/Response tab to see the source code, and verify that the selected pattern does indeed indicate in-session status.

**Note:** If the page content is JavaScript or CSS, then in all cases it is not suitable as the in-session page, and you should choose another page.

If the key icon is green, but the selected pattern is *not* an in-session pattern, refer to “[Request-based login troubleshooting](#)” on page 300.

4. **Lockout configuration.** During the Test stage, AppScan makes many invalid login attempts. If your site has an account lockout feature, that locks users out when invalid passwords are entered a certain number of times, AppScan will get locked out and be unable to complete the scan.
  - Disable account lockout, or (if this is not practical)
  - Configure AppScan not to test login and logout pages (**Scan Configuration > Test Options**, deselect **Send tests on login and logout pages**).

#### Related concepts:

“Workflow for advanced users” on page 281

This workflow can help users with experience in the field of web security achieve a more thorough scan.

## Initial Automatic Explore

After your basic configuration, you can run an initial Automatic Explore to see how well AppScan covers the site at this stage.

### About this task

A full scan consists of both Explore and a Test stages, but for the moment run only an Explore stage.

Site coverage based on your initial configuration may be incomplete, but seeing which parts of the site are discovered and which are not can help you improve the configuration.

### Procedure

1. Click **Scan > Explore Only**.  
Scan Expert is configured to run automatically before the Explore stage.
  2. If Scan Expert suggests configuration changes, follow its recommendations.
- Note:** Some changes can be implemented automatically, others require input from you.
3. Let AppScan explore the site until it stops. During the scan you can see the Application tree and data fill in as the site is crawled.
  4. Verify that the Explore stage finished successfully and did not terminate early due to AppScan being out-of-session.
- Note:** If it ended early because AppScan was out-of-session, re-record and reconfigure the login procedure, paying special attention to In-session detection and session ID tracking.
5. (Sites that use URL rewriting:) If your site uses URL rewriting, the Explore Optimization module can improve scan efficiency by eliminating unnecessary URLs from the Explore results at this stage.
    - a. Click **Tools > Extensions > Explore Optimization: Run**
    - b. If recommended by the module at the end of the process, click **Scan > Re-Explore**.

**Attention:** Do not perform this step unless you are sure your site uses URL rewriting.

#### Related concepts:

“Workflow for advanced users” on page 281

This workflow can help users with experience in the field of web security achieve a more thorough scan.

## Improve site coverage manually

You can add URLs that the initial Automatic Explore stage missed, both individual URLs (such as those accessed by forms that require specific input) and ordered sequences of URLs (such as shopping carts).

## About this task

There are various reasons why AppScan may be unable to crawl certain parts of the site automatically without first getting input from you.

- If the application has forms that require specific input, you can navigate to the page using the Manual Explore feature and fill in the required data. This will be recorded in the automatic form filler for use during scanning.
- If your application uses JavaScript, Java applets, or Flash - and has links that appear only in certain states, following a specific sequence of states - such links may be missed by Automatic Explore. Using Manual Explore to access them will ensure that AppScan tests them and any additional links they lead to.
- If your application uses Java applets that reveal links, these will not be tested by AppScan unless you explore them manually.
- If certain parts of the site can only be reached by clicking links in a *specific order* (such as a shopping cart), you should record a Multi-Step Operation (**Scan Configuration > Multi-Step Operations view**).

## Procedure

1. **Manual Explore.** Use this feature to add URLs that were not discovered during the automatic Explore, and which can be accessed without a specific context.

- a. On the toolbar, click **Manual Explore**.

The AppScan built-in browser opens.

**Note:** If the application does not appear in the browser as expected, it may not be optimized for the AppScan built-in browser, in which case you can configure AppScan to use a different browser. See "Changing the default browser" on page 297.

- b. Browse the application, clicking the links you want tested, completing data, and trying to cover as much functionality as possible.
- c. When finished, close the window (by clicking the X button on the titlebar).

The manual Explore window opens.

- d. Save the data for use with future scans by clicking **Export**, and saving.
- e. To add the data to the current scan, select **Add All**, then click **OK**.
- f. After the new pages load, on the toolbar click **Scan > Continue Explore Only**.  
AppScan will now explore new links revealed by your Manual Explore.
- g. After the Explore is finished, review the Application data to verify that the Explore achieved the coverage you expected.

2. **Multi-Step Operations.** Use this feature to record sequences of URLs that must be accessed in a specific order.

- a. Open **Scan Configuration > Multi-Step Operations view**.
- b. Click the red **Start Recording** button, and choose **Login and then record**.
- c. Log in to the application and perform the process you want to record (such as adding an item to the shopping cart and checking out).

**Note:** You can click **Pause** if you need to click links that you do not want included in the recording, and then click **Pause** again to resume recording.

- d. Close the browser window.
- e. If some of the parameters in the sequence require a unique value, refer to tech note "Using Variables in Multi-Step operations":

<http://www.ibm.com/support/docview.wss?uid=swg21327313>

**Related concepts:**

"Workflow for advanced users" on page 281

This workflow can help users with experience in the field of web security achieve a more thorough scan.

"Using AppScan" on page 115

Manual exploring enables you to explore specific parts of your application, filling in fields and forms as you go. This can be a way of ensuring that particular areas of the site are covered, and that AppScan has the information needed to complete forms correctly.

#### Related reference:

"Multi-Step Operations view" on page 79

Multi-Step Operations view of the Configuration dialog box is for testing parts of the site that can only be reached by clicking links in a specific order.

## Evaluate Explore results

Before proceeding to the Test stage, review the Explore results, because if important areas of the site were missed during the Explore stage, they will not be tested during the Test stage.

### About this task

The results of the Explore stage are shown in the three Data view panes. The following are some tips for evaluating whether the Explore stage worked well and achieved good coverage of the application.

**Note:** If you make any configuration changes during this stage, you should re-Explore the application before starting the Test stage.

### Procedure

1. **Scan Log.** Use this to check that AppScan wasn't out-of-session too often.
  - a. Click **View > Scan Log**,
  - b. Scroll down the log entries to check that AppScan wasn't out-of-session too often.

If AppScan was out-of-session more than a couple of times in five minutes, it may help to re-record and reconfigure the recorded login, paying special attention to the in-session detection configuration.
2. **Application Tree.** This is a graphical representation of all the areas of the site that were discovered and explored. Use it to see how well the site was covered.
  - a. Does the application tree accurately show the hierarchical structure and main pages of your application?
  - b. Are there Login URLs in the tree? (If there are not, the login was never sent.)
  - c. Does the total number of visited URLs (bottom left-hand corner) match your understanding of the actual site size?
  - d. Were a reasonable number of tests created, for sending during the Test stage (should be at least five times the number of URLs)?
3. **Requests sent.** Review and validate the requests sent during the Explore stage.
  - a. In the Data pane, select **Requests** view, to display all requests sent.
  - b. Verify that the login URLs appear in this list, particularly the in-session request and the login request containing the user credentials.
  - c. Look over some of the requests that appear after the login request in the Login procedure. Verify that the response does not contain errors. To do this, type the word "error" into the Detail pane search field, and then select URLs one by one in the upper panel. If a particular response contains the word "error", the search field color will change from red ("not found"), to green ("found"), and the word "error" will be highlighted in the response body.
  - d. If these requests contain error strings, this indicates that the user was out-of-session, and therefore that the Login procedure was not correctly recorded. Record it again.
4. **Application Data view.** This is the default view during the Test stage, and it offers various views by clicking the filters along the top of the pane.

- a. Click F2, or the data icon on the right hand side of the toolbar to open this view.
  - b. At the top of the Data pane, select a filter to view information.
  - c. Click on an item in the data pane to view its details in the Detail pane.
5. **Custom error pages.** 4xx responses are automatically identified as error pages. If your site returns 2xx responses with custom error pages, you must configure AppScan to recognize them. This information is essential in deciding whether tests succeed. Not configuring custom error pages will lead to inaccurate results, both false positive and false negative. Therefore, if you found pages in the previous step containing the word "error" in their response but were not classified as error pages, configure them now.
- a. In the Detail pane, click **Show in Browser** to verify that it is indeed an error page.
  - b. Click **Set as Error Page**.
- Note:** You can also define error pages in **Scan Configuration > Error Pages**, by clicking the plus (+) icon, and defining a string, regular expression, URL or page.
6. **Filtered URLs.** Review the list of requests that were *not* sent, to verify that it does not contain requests that *should* have been sent.
- a. In the Data pane, select **Filtered URLs** view, and verify that the filtered URLs should indeed be filtered, and that they are categorized correctly.
  - b. If URLs were incorrectly filtered because of their domain, ("Untested web Server"), add the domain to the scan (**Configuration > URLs and Servers > Additional Servers and Domains > +**).
  - c. If URLs were incorrectly filtered because the Path Limit was reached, consider making one of the following configuration changes:
    - Increase Redundant Path Limit (**Configuration > Explore Options > Redundant Path Limit**)
    - Adjust default Redundancy Tuning (**Parameters and Cookies > Redundancy Tuning Defaults**)
    - Adjust Redundancy Tuning for individual parameters
7. **Parameter-based navigation.** If all or part of the site sends a single URL, while different parameters control content and structure, refer to "Sites that use parameter-based navigation" on page 288.
8. **Parameters.** In the Data pane, review the parameters discovered during the Explore stage.
- a. In the Data pane, select **Parameters** view, to see all parameters discovered during the Explore stage.
  - b. If necessary, update the definitions (**Configuration > Parameters and Cookies**).
9. **Failed Requests.** These are requests whose response status was 4xx ("Error"). Review this list to check whether legitimate requests unexpectedly got error responses.
- a. In the Data pane, select **Failed Requests** view.
  - b. **404 Not Found:** Click **Show in Browser** to verify that the URL doesn't exist.
  - c. **Timeout or Connection Failed:** See if the scan needs a higher timeout (**Configuration > Communication and Proxy > Timeout**), the site's server or environment needs improving, or whether the connection problems are due to multi-thread scanning (**Configuration > Communication and Proxy > Number of Threads**), and can be solved by reducing the number of simultaneous threads sent.
  - d. **401 or 407 Authentication Required:** This means there are areas of the application that require HTTP authentication (set these in **Configuration > Platform Authentication**).
  - e. **Other 4xx statuses:** Check whether the site returned an error because the user was not logged in. If necessary, record the login procedure again (**Configuration > Login Management**).
10. If, after reviewing the initial Explore results, you feel that coverage is insufficient, refer to the next section for possible configuration changes. See "Additional configuration" on page 288.

#### Related concepts:

"Workflow for advanced users" on page 281

This workflow can help users with experience in the field of web security achieve a more thorough scan.

“Sites that use parameter-based navigation”

Sites in which all pages are reached using a single URL, need a specific scan configuration.

#### Related tasks:

“Additional configuration”

Here are some configuration changes to consider if site coverage in the initial Explore stage was insufficient.

## Additional configuration

Here are some configuration changes to consider if site coverage in the initial Explore stage was insufficient.

### About this task

If site coverage in the initial Explore stage was insufficient, or took more than about 30 minutes, consider making some or all of the following possible configuration changes to parameter and cookie definitions.

### Procedure

1. Click **Configuration > Parameters and Cookies** view.
2. **Tracking.** If your site tracks cookies (such as session identifiers that are updated each time a user logs in), you should make sure they are correctly defined.
  - a. Verify that your site's cookies and parameters have been identified and are listed in the main Parameters and Cookies tab, and set as "Tracked".
  - b. If necessary, define additional parameters and cookies by clicking the  icon. For more details see “Parameter definition” on page 66.
3. **Redundancy Tuning.** Careful redundancy tuning can significantly reduce scan time without any reduction in scan coverage and accuracy. Review the redundancy tuning to ensure that unnecessary duplicate requests are not sent, but necessary ones are. For more details, see “Redundancy tuning” on page 69.
  - a. If the cookie or parameter you want to configure is not listed in click the  icon and define it.
  - b. Verify that settings are appropriate for each cookie or parameter.
4. **Session ID verification.** Accurate session ID definition is important to enable AppScan to create legitimate requests during the Explore stage.
  - a. Verify that the tracking setting for all session IDs are correctly defined as Login Value.
  - b. Verify that all parameters set as Login Value are session IDs.
5. If you made any changes to the Parameters and Cookies definitions, re-Explore the application. It may also be necessary to re-record the manual Explore and multi-step operations.

“Parameters and Cookies view” on page 65  
Parameters and Cookies view of the Configuration dialog box.  
“Parameter definition” on page 66  
“Session IDs” on page 69  
“Redundancy tuning” on page 69  
Careful redundancy tuning can significantly reduce scan time.  
“Workflow for advanced users” on page 281  
This workflow can help users with experience in the field of web security achieve a more thorough scan.

---

## Sites that use parameter-based navigation

Sites in which all pages are reached using a single URL, need a specific scan configuration.

A site with "parameter-based" navigation is a site where only one URL (controller) is sent, while different parameters in the URL result in different content and structure being returned. (Sites of this kind are sometimes referred to by our support team as "megascript" sites.)

- In some sites the URL does actually remain the same for all "pages", as in the following example:  
`http://site.com/content.aspx?PageName=page1`  
`http://site.com/content.aspx?PageName=page2`
- In other sites, all links are directed through a single, proxy page, using a GET parameter. For example:  
`http://site.com/default.aspx/redirect=page1`  
redirects to:  
`http://site.com/page1.aspx`  
This poses the same problem for AppScan as the previous case, since the same URL is sent each time.
- In the case of ASP.NET 2.0 postback links, each link generates a POST request to the page it is located on.  
Once again, the problem posed to AppScan is the same.

In all cases the requests for all "pages" are sent to the same URL. This requires special treatment from AppScan, as the default configuration will result in an incomplete scan.

#### To scan a site with parameter-based navigation

1. When creating the scan, select the Parameter-Based Navigation template instead of the Regular Scan template.
2. Verify that the navigational parameter(s) of your site are correctly defined: Go to **Scan > Scan Configuration > Parameters and Cookies**, and check that the regular expression defining the last parameter in the list includes the navigational parameter(s) of your site. If necessary edit the regular expression. (See "Parameters and Cookies view" on page 65.)
3. (Optional and Advanced:) Configure the Content-Based Results tab of the Scan Configuration dialog box, so that AppScan can present a meaningful application tree. (See "Content-Based Results view" on page 82.)
4. Make any other configuration changes you need, and continue as for a regular scan.
5. (Optional:) When viewing scan results in the application tree, select Content-based view instead of the default URL-based view. (See "Application Tree" on page 20.)

*See also:*

"The Paramater-Based Navigation template" on page 112

"The challenge of parameter-based navigation sites"

## The challenge of parameter-based navigation sites

An explanation of the configuration changes needed to scan a site in which the navigation is parameter-based.

By default, AppScan has a Redundant Path Limit of 5 (the maximum number of time a request can be sent to the same URL, see "Explore Options view" on page 62). In regular sites this prevents repeating tests unnecessarily. However, when site navigation is parameter-based, this low limit will effectively prevent AppScan from scanning the site thoroughly, and a scan run using the Regular Scan template will discover and test hardly any of the site.

Increasing the Redundant Path Limit, or disabling it altogether, does not alone solve the problem. In fact it may throw AppScan into an infinite loop, or at least create a scan with so many tests that AppScan runs out of memory. There are two reasons for this:

1. During the Explore phase, when hashing a request, AppScan will include all the parameters and cookies it finds in the request. With the redundant path limit gone, all combinations of these values will be taken into consideration.

Let's say for example that each page from a section of the site contains hundreds of links to a script that retrieves information from the database about an item available for sale. These links contain a parameter named `item_id` which is of no importance in generating new pages, and is used only to retrieve information about the item. AppScan will end up requesting thousand of instances of this item info page unless `item_id` can be excluded from the hash.

2. In the Test stage the problem becomes more severe. Let's say that a request contains two parameters `par1` and `par2`, and AppScan encounters four links that include these parameters:

```
http://site.com/content.aspx?par1=a&par2=c  
http://site.com/content.aspx?par1=a&par2=d  
http://site.com/content.aspx?par1=b&par2=c  
http://site.com/content.aspx?par1=b&par2=d
```

If there are 400 tests applicable to each parameter, AppScan will send 1600 tests in total (800 on `par1` when `par2=c` and `par2=d` and 800 on `par2` when `par1=a` and `par1=b`)! Therefore, in addition to excluding these parameters from the Explore hash, we must tell AppScan to test each parameter only once: 400 tests on `par1` and 400 tests on `par2`.

The principles derived from the above for scanning a site with parameter-based navigation are therefore:

1. **Explore stage:** Ignore the values of all parameters except navigational parameters.
2. **Test stage:** Do not create new tests when the value of a parameter changes, except for navigational parameters.

*See also:*

"Sites that use parameter-based navigation" on page 288

"The Parameter-Based Navigation template" on page 112

---

## Live production environments

When the site being scanned is live, there are several possibilities that should be taken into account.

### Database may get filled with artificial information sent during scanning

You can reduce the impact of this by taking the following precautions:

- Disable Automatic Form Fill (Scan Configuration > Automatic Form Fill > first check box).  
This will ensure that AppScan does not fill forms automatically, submitting data that might flood a database, bulletin board or online forum system, or send unwanted email to an administrator or moderator account. However, you should be aware that doing this will limit AppScan Standard's ability to reach areas of the site that are accessed by submitting forms. In this mode of operation, AppScan will only scan areas of the site that can be accessed by following links (with or without parameters).
- Create a test account for AppScan to use.  
Using a test account makes it easier to track database changes (for example, to make sure that services are not actually ordered), and helps site administrators clean up site after scanning.  
Consider the following suggestions when creating the account:
  - Limit database access to test records only, so that modified records can be restored.
  - Ensure that new records created by the test account will be deleted.
  - Ensure that purchase orders (or other transactions) from the test account will be ignored.
  - If transactions have an impact (such as when dealing with stocks), allow the account access to test records only.

- If the site has forums, allow the test account access to test forums only, so that real customers will not see the tests created during the Test phase.
- If the site has different privileges for different accounts, set up more than one test account, with different privileges. This will ensure a more comprehensive scan of the site.
- Do not create a test account with administrator-level access.

### Risk of email flooding

When testing pages that use email notification, AppScan generates many requests and may overload the site's email server.

One or more of the following suggestions can help dealing with this:

- Temporarily change the email addresses on the pages being tested, so that email is sent to an invalid email address.
- If practical, configure AppScan to exclude those pages from the production scan.
- Scan only one web server at a time, and prevent it from connecting to the SMTP server during the scan.
- If you decide to leave Automatic Form Fill enabled, configure it to insert a unique value in the email field, so recipients can easily identify email generated by AppScan.

### Scanning through a proxy

If possible, avoid scanning through a proxy. While this is supported, the proxy sometimes obscures results.

### Risk of overloading the application

The application might not be able to handle the large amount of HTTP requests sent by AppScan in a short time, resulting in reduced service to actual users.

To reduce the risk of this happening, or to troubleshoot the problem when it does:

- Reduce the number of threads AppScan sends at the same time (Scan Configuration > Communication and Proxy > Number of Threads).

### Risk of scan getting locked out of the application

Some applications are configured to lock users out after a certain amount of incorrect login attempts. If this happens during the scan, obviously AppScan will be unable to complete the scan.

To avoid this:

- Disable **Send tests on login and logout pages** (Scan Configuration > Test Options).

### Risk of causing application failure

To avoid the risk of AppScan causing your live application to fail, you may want to deactivate invasive tests in the test policy. This will ensure that Denial of Service, Buffer Overflow, or other tests that might cause the application or web server to fail, are not sent.

**Important:** web applications often contain vulnerabilities that can only be discovered by invasive tests. It is not recommended that you omit invasive tests altogether. Instead, test your application for these kinds of vulnerabilities in coordination with your website owner or administrator, perhaps scheduling scans during off-peak hours when the application is likely to be idle.

To disable invasive tests in the current test policy:

1. Open Configuration > Test Policy.
  2. Click on the **Invasive** column, to group all invasive tests together.
  3. Scroll down the invasive tests (tests for which the Invasive value is "Yes"), and deselect any that are currently selected, to exclude them from the scan.
- 

## Flash content

This section contains suggestions for scanning Adobe Flash content.

AppScan explores and tests Adobe ActionScript 1.0, 2.0 and 3.0, and Adobe Flex frameworks 2.0 and 3.0. Flash parsing and execution are activated from Explore Options view of the Scan Configuration dialog box.

### System requirements

In order for AppScan to be able to execute Adobe Flash content during scanning, you must have a supported version of the Adobe Flash Player for Internet Explorer installed. Versions 9.0.124.0 or later are supported.

**Note:** The Flash Player comes in the form of an ActiveX plugin that is browser-specific. AppScan requires the Adobe Flash Player for Internet Explorer.

- If your version is too old, you can download the latest Flash Player from <http://get.adobe.com/flashplayer/>
- If you have Adobe Flash Player Version 10.1 or higher, you may get a message that it requires configuration to work with AppScan. See "Flash Player configuration" on page 7

**Note:** In both cases ("Supported Flash Player not installed", or "Flash Player not configured") an alert will appear in the Configuration dialog box *and Flash Execution will not run* during scans.

### Limitations to Flash scanning

The following limitations to Flash scanning should be noted:

- Since AppScan scans Flash content through a proxy, localhost URLs are not scanned.
- Issues found are specific to the version of the Adobe Flash Browser for Internet Explorer that is installed on your machine. It is possible that:
  - Your player is vulnerable to the issue reported, but a player for a different browser, or a later version of the player, is not
  - There are issues that your Internet Explorer player is *not* vulnerable to, and therefore that AppScan does not report, which *do* affect players for other browsers, or earlier versions of this player

### Incomplete URL coverage

The following are suggestions for when you have run a scan and looked at the results and it seems that AppScan has not identified URLs from your Flash content.

#### Why does AppScan identify some URLs from the Flash content but miss others?

There are a few possibilities:

- Verify that the Flash movie version is supported. (Unsupported versions are listed in Application Data view under "Filtered URLs.")
- Play the movie in the Internet Explorer browser, on the machine that ran the scan, to verify that it plays correctly.

- Check that JavaScript Execution (enabled by default) has not been disabled. (**Scan Configuration > Explore Options > Execute JavaScript to discover URLs and dynamic content**)

### The Flash content coverage seems to be incomplete

There are a few possibilities:

1. Check that the form filler information is complete (**Scan Configuration > Automatic Form Fill**).
2. Try increasing the **Scan Configuration > Explore Options > Flash > Click limit**.
3. Try increasing the **Scan Configuration > Advanced > Flash: Max time between samples** setting above its default value of 160 ms.
4. Try increasing the **Scan Configuration > Advanced > Flash: Coverage** setting from 1 to 2.
5. Try playing the movie in Internet Explorer to verify that it plays as expected.
6. If the movie does not play, set the Flash Browser **Debug Level** to Trace 3, rescan, and send the browser log (...\\Program Files\\IBM\\AppScan Security\\Logs\\AppScanFlashBrowser.log ) to your Support provider.

### Vulnerabilities not discovered

The following suggestions are for when AppScan has discovered the URLs and added them to the application tree, but has not discovered vulnerabilities in them.

#### Why doesn't AppScan discover vulnerabilities in the Flash URLs it discovered?

Possible reasons are:

- There may simply be no suspicious parameters in the movie. Look in **Application Data > Script Parameters** to see the Flash parameters that were discovered.
- Check that all Flash tests (ActionScript 2 and 3) are enabled. (Open **Scan Configuration > Test Policy**, search for "ActionScript", and verify that all tests are selected).
- The movie may simply not be vulnerable.

#### What else can I do?

If you still suspect that Flash vulnerabilities are being missed, enable Extended Support Mode, repeat the scan, and send to your Support provider. See "Extended Support Mode" on page 296.



---

# Chapter 14. Troubleshooting

This section provides instructions for do-it-yourself troubleshooting.

---

## License troubleshooting

### About this task

In order to use AppScan for scanning your site you must have a valid license. If you are sure your license is valid, but AppScan does not accept it, try the following steps.

### Procedure

1. Verify that the server on which AppScan is used has a network connection with the license server.

**Note:** From Version 7.8 onwards, AppScan Standard Edition licenses are installed onto the IBM Rational License Server, which may not be the same as the server AppScan is run on. In order for AppScan to function, the server on which it is installed must have a network connection with the license server. Each time the user opens AppScan a licence is checked out, and when AppScan is closed the license is checked back in.

2. Open the license with a text editor (such as MS Notepad).

**Important:** Do not alter the license file!

3. Verify that the license has not expired.
4. Verify that the site you want to scan is within the license restrictions (both the IP and the hostname).
5. Verify that the MAC address and Disk Serial Number on the license are the same as on the machine.

### What to do next

If this procedure does not solve the problem, contact AppScan Support.

---

## Troubleshooting features

AppScan provides various information that you can use to more efficiently locate and solve problems.

- **AppScan Log** - Displays information about AppScan functionality, including how it connects to your proxy and the port that the proxy listens to after each restart.
- **Traffic Log** - When enabled, AppScan writes to this log, all traffic sent to and received from your website.

**Note:** Enabling the traffic log will slow the application and increase disk space consumption.

- **Dump File** - In the event of it crashing, AppScan creates a memory dump file and notifies you of its path and filename. This file contains all available information about the use case that caused the crash, and should be sent to AppScan Support for further investigation.
- **Support Mode** - Enables you to reproduce the problem and then to pack data files, to send to technical support.
- **Packed Files** - AppScan creates an archive of data files for you to send to technical support. These files include: scan file (\*.scan), AppScanDbg.log, AppScanSys.log, and AppScanTraffic.log.
- **Report False Positive** - Enables you to send IBM Rational AppScan Support feedback about specific tests, especially those with positive test results that you suspect should not be positive in your application. AppScan zips information from selected variants and attaches the zip file to a new e-mail message, opened in the default e-mail client.

## Reporting false positive results

If you feel that a particular AppScan result is wrong (false positive) you can email the request that AppScan sent, and your web application's response, to IBM Security AppScan Support. See "Report false positive test results" on page 187.

## Troubleshooting the Report False Positive feature

The following tips are provided to help you if you experience any difficulty using the Report False Positive feature to send the AppScan support team feedback with attached variant information.

Problem	Cause	Solution
Recipient cannot read attachment	E-mail was sent encrypted.  No other recipient will have the private key needed to read the e-mail.	To turn off encryption: From <b>Tools</b> menu > <b>Options</b> command > <b>General</b> tab > Report False Positive option, clear <b>Encrypt Attachments</b> check box.

## Extended Support Mode

Extended Support Mode logs all AppScan activity, for packing and sending to your support provider to help troubleshooting a problematic procedure.

### About this task

If you need help troubleshooting a problematic procedure, your support provider may ask you to run AppScan in Extended Support Mode, to log all activity and pack the data into a single file.

**Important:** Do not turn on Extended Support Mode unless directed to do so by authorized support personnel. This mode affects AppScan performance.

### Procedure

1. Activate Extended Support Mode: **Help > Support > Extended Support Mode**  
A message appears to say that Extended Support Mode is enabled, and will log all your actions.
2. Click **OK**.  
An indicator in the Status bar shows that AppScan is running in Extended Support Mode:  
 **Extended Support Mode**
3. Reproduce the problematic procedure.
4. When you have finished, disable Extended Support Mode: **Help > Support > Extended Support Mode**  
OR  
If AppScan stopped working as a result of the procedure, reopen AppScan, and when asked if you want to keep Extended Support Mode enabled, select **Disable**.
5. Select the options you require:

Option	Description
<b>Encrypt</b>	All support data that is saved, including the scan file if included, will be encrypted. <b>Important:</b> Do not select encryption unless you are sending the package to AppScan Support, as only they can open the encrypted file.
<b>Include a scan file</b>	You can include the current scan, or another relevant saved scan, in the support file.

6. Click:

Option	Description
Save	Saves the existing support data as a .SUPPORT file (encrypting and including a scan file if so configured). The support data is cleared from the AppScan interface, and AppScan reverts to its regular mode.  The default path is:  ...\\My Documents\\AppScan\\Support\\
Don't Save	The support data is cleared from the AppScan interface, and AppScan reverts to its regular mode.
Cancel	The support data is <i>not</i> cleared from the AppScan interface, and AppScan remains in Extended Support Mode.

## Changing the default browser

You can configure AppScan to use a browser other than its built-in browser.

### About this task

By default, AppScan opens its built-in browser when you record the login procedure, or a multi-step operation, or when you click the **Show in Browser** button.

If the built-in browser is unable to access parts of the site, or if your site is not optimized for it, you can configure AppScan to use a different browser that you have installed on your machine instead.

### Procedure

1. Click **Tools > Options**, and select **Use External Browser**.
2. Select a browser from the drop-down list. Options are:
  - Internet Explorer
  - Mozilla Firefox
  - Google Chrome

**Note:** You can only choose a browser that is already installed on your machine.

3. If either of the following two configuration options has been changed, the defined header values will be used, and not the header values of the newly selected browser. If necessary, you can restore the default values
  - **Configuration > Advanced Configuration > Communication: Accept-Language request header value**
  - **Configuration > Explore Options > User-Agent header value**

The ways in which these settings may have been changed *by the user* are:

- You opened the browser to record the login procedure or a multi-step operation, or clicked "Show in Browser"
- You manually changed them
- You loaded a saved scan

In order to set these two settings to those of the newly defined browser, simply delete the values, and (if relevant) record the login or multi-step operation again. From now on AppScan will automatically use the header value of the defined browser.

**Note:** The following limitations apply to the newly defined browser:

- If the selected browser is configured to use a proxy configuration file, this selection will be ignored and the built-in browser will open.
- If you record a multi-step operation using the "Login and then Record" option, the built-in browser will open.
- If another instance of AppScan is open, and you try to open an external browser, the built-in browser will open.
- In **Login Management > Details tab**, if you select a URL and click either **Show in browser** or **Select**, the built-in browser will open.
- If you select Firefox, AppScan automatically creates a Firefox profile named "AppScan" (if one does not already exist). If Firefox was open when this was done, you must close it for the change to take effect.

#### Related concepts:

"Workflow for advanced users" on page 281

This workflow can help users with experience in the field of web security achieve a more thorough scan.

#### Related tasks:

"Improve site coverage manually" on page 284

You can add URLs that the initial Automatic Explore stage missed, both individual URLs (such as those accessed by forms that require specific input) and ordered sequences of URLs (such as shopping carts).

#### Related reference:

"Advanced Configuration view" on page 98

The **Advanced** tab of the **Scan Configuration** dialog box is used to change advanced registry settings that affect specific scans (Scan > Scan Configuration > Advanced tab), and it should only be used by experienced AppScan users, or when instructed to do so by the support team to troubleshoot a problem.

"Explore Options view" on page 62

Explore Options view of the Configuration dialog box.

## Login troubleshooting

Tips for troubleshooting session detection problems in Scan Configuration > Login Management view.



When you close the browser after recording the login procedure, the green key icon confirms that AppScan has detected an in-session pattern that can be used during scanning to verify its in-session status. If one of the other icons appears instead, AppScan may not have enough information to log in to the site during scanning.

Scan Configuration > Login Management > Details records the login sequence in two ways: as actions and as requests. As long as *one* of these two methods succeeds, AppScan will be able to log in to the site. The table below can help you troubleshoot in cases where both methods have failed.

Actions		Requests	
ID	Type	Value	Element
1	Click		Sign In
2	Set	jsmith	uid
3	Click		passw
4	Set	*****	passw
5	Click		btnSubmit

The following table summarizes messages and possible user troubleshooting actions.

Icon	Message	Possible user actions
	<b>Using action-based login</b> Action-based login: Succeeded Request-based login: Succeeded	No action necessary. Action-based login will be used, and request-based login will be available as a fallback method.
	<b>Using action-based login</b> Action-based login: Succeeded Request-based login: Failed	No action necessary. Action-based login will be used.  To troubleshoot the request-based sequence, see "Request-based login troubleshooting" on page 300
	<b>Using request-based login</b> Action-based login: Failed Request-based login: Succeeded	No action necessary. Although action-based login is the preferred method, since request-based login succeeded, that version will be used.  To troubleshoot the action-based sequence, see "Action-based login troubleshooting" on page 300 <b>Note:</b> If one of the login pages is very slow, it may be more practical to use request-based login as many logins are typically required during a scan.
	<b>Login not yet recorded</b>	Either click  and record a login or, if login is not required, in the Login/Logout tab > Login method, select <b>None</b> to disable session detection.
	<b>Login not yet validated</b>	If changes have been made to one of the sequences, you must click the <b>Validate</b> button to validate the new login sequence.
	<b>Session request same as login request</b>	Generally, the login sequence should end after AppScan is logged in to the application. However, in rare cases, the in-session request also contains the login request (with username and password). In such cases, whenever AppScan replays the in-session request (to verify that it is logged in) it will actually log itself in, and therefore be unable to detect when it is logged out.  The solution is to record the login sequence again, and at the end of the recording, when logged in to the application, to click another link on the page. Then stop recording. The log in sequence will now have an extra step. As long as this new request does not include the credentials, AppScan will be able to use the sequence to verify when it is logged out. The key icon will change to green.
	<b>Session page redirects</b>	If the page selected as the first in-session page redirects to another page, it is likely that the in-session pattern selected by AppScan is incorrect. <ul style="list-style-type: none"> <li>Verify that the current <b>In-Session Detection Pattern</b> does indicate in-session status</li> <li>If you are unsure, try adding the redirect page as an extra step in the request-based login sequence</li> </ul>

Icon	Message	Possible user actions
	<b>Session page not identified</b>	In the Request tab, open the final page of the login sequence, look for a pattern (either in the Browser tab or the Request/Response tab) that is <i>unique</i> to logged-in users (such as a "log out" link), and select that as the in-session pattern. <ul style="list-style-type: none"> <li>• To troubleshoot the action-based sequence, see "Action-based login troubleshooting"</li> <li>• To troubleshoot the request-based sequence, see "Request-based login troubleshooting"</li> </ul>
	<b>Session detection disabled</b>	No action necessary.  Session detection can be enabled by selecting one of the three Login methods: Recorded, Prompt, or Automatic.

## Action-based login troubleshooting

Use these steps to troubleshoot action-based login.

### Procedure

1. In **Login Management > Details > Actions**, verify that the username (uid) and password (passw) are correct (double-click on the password value to see the value entered), and correct it manually if needed
2. Click  to play the sequence in the player. This may help you see where the sequence fails.  
  
**Note:** If one of the login pages is very slow, it may be more practical to use request-based login as many logins are typically required during a scan.
3. Click  and record the sequence again, but this time:
  - a. Type the username and password in a text editor and copy-and-paste them into the web page
  - b. Instead of clicking the Submit button on your site, click **Enter** on your keyboard
 If none of these steps solve the problem, try using the request-based sequence.

## Request-based login troubleshooting

If the In-Session Detection Pattern shown in the Details tab does *not* correctly identify in-session status, you can select a different pattern using the Requests sequence.

### Procedure

1. In **Login Management > Details > Requests**, select the URL marked In-Session (highlighted in green), then click the **Select** button at the bottom of the dialog box, to choose a different pattern.  
  
The browser opens and you can select a new pattern either in the browser or the response body tab. Then close the browser and click **Validate**.
2. If you cannot identify an in-session pattern on the final page, do the following:
  - a. Select the request *above* the request you just looked at
  - b. Double-click on it and check that it does not contain the login credentials
  - c. If it does not, click **Select**, and try identify a different pattern
3. If you do not find an in-session pattern, repeat the previous step for the next request up. You can repeat this as necessary, until you reach a request that contains the login credentials.

4. If you are unable to identify an in-session pattern in any of these pages, and there are one or more URLs listed *after* the In-Session page, use the same procedure to look for an in-session pattern on *that* page.
5. If there are *no* extra URLs, try recording the login sequence again, but click one extra link after you are logged in, preferably a personalized setting, and look for an in-session pattern on that page.
6. If this fails, try selecting an out-of-session pattern:
  - a. Select the URL that was originally marked as the In-Session request
  - b. Open a browser (outside AppScan) and send this request on its own (not preceded by the rest of the login sequence)
  - c. Compare the two responses, and try to identify an expression in the body of the response from Step B that does not exist in the in-session page (such as "You are not logged in")

**Note:** If the request redirects a different page, you cannot use the response you see in the browser, but need to use the response to the actual request, which can be done using a sniffer

- d. At the bottom of the Details tab click the In-Session drop-down button and select **Out-of-session**, and then paste the pattern you identified into the **Detection Pattern** field

## What to do next

If this procedure does not solve the problem, refer to the following online technote:

<http://www.ibm.com/support/docview.wss?rs=3378&uid=swg21283302#Overview%20of%20In-Session%20Detection>

## Long or never-ending Explore stage

For some types of site the Explore stage may take a long time or never end.

The cause of this problem may be that the site rewrites parameters into the URL, creating tens - or even hundreds - of "dynamic" URLs for what are essentially the same page.

### Example:

```
http://...php/1/index
http://...php/2/index
...
http://...php/100/index
```

Although these URLs will appear in the application tree as 100 separate nodes, and tests will be created for each of them, there is probably no need to test them all separately.

The Explore Optimization module can identify the pattern and create a custom parameter that will enable AppScan to treat all these URLs as one.

See "Explore Optimization module" on page 269

## Flash movie troubleshooting

This section offers suggestions for troubleshooting the Flash scanning feature.

Problem	Action
After installing a new version of AppScan, I get the following persistent error message each time I open it:  FlashBrowser has encountered a problem and needs to close.	<ol style="list-style-type: none"><li>1. Close AppScan</li><li>2. Delete all files named <b>user.config</b> in folders of type (Windows 7): C:\Users\&lt;USER_NAME&gt;\AppData\Local\\IBM_Corporation\FlashBrowser.exe_Url_&lt;CODE&gt;</li><li>3. Restart AppScan</li></ol>
AppScan didn't detect any URLs in the movie.	<ol style="list-style-type: none"><li>1. Verify that your system meets the system requirements.</li><li>2. Verify that "Execute Flash files to discover potential vulnerabilities" (<b>Scan Configuration &gt; Explore Options</b>) is selected.</li></ol>
AppScan identified URLs in some Flash movies but not in others.	<ol style="list-style-type: none"><li>1. Make sure the Flash movie version is supported. (Unsupported movies are listed in "Application Data" view under "Filtered URLs".)</li><li>2. Play the problematic movie in Internet Explorer, on the machine that ran the scan. Verify that the Internet Explorer can play it properly, and the IE status bar displays 'Done' at the end of the movie.</li><li>3. Verify that JavaScript Execution is enabled in <b>Scan Configuration &gt; Explore Options</b>.</li></ol>
Scan coverage is incomplete.	<ol style="list-style-type: none"><li>1. Make sure that <b>Scan Configuration &gt; Automatic Form Fill</b> information is complete.</li><li>2. In Advanced configuration, increase Flash: Time between samples.</li><li>3. In Advanced configuration, set 'Flash: Coverage' to 2.</li><li>4. Set <b>Advanced Options &gt; ShowDebugFlashExecution</b> to True, restart AppScan, and verify that the movie plays properly in the "Player container" that opens.</li></ol>
AppScan doesn't find any vulnerabilities in the Flash movies.	<ol style="list-style-type: none"><li>1. Make sure that in the Test Policy, all ActionScript 2 and 3 Tests are enabled.</li><li>2. Check in <b>Application Data view &gt; Script parameters</b> that no Flash parameter is missing.</li><li>3. This particular movie may simply not be vulnerable.</li></ol>
Why do I notice a change in standalone Flash movie performance after exploring Flash movies with AppScan?	AppScan makes a reversible configuration change to Flash Player versions 10.1 and later. For details, and instructions for undoing the change, see "Restore Adobe Flash Player settings" on page 304.
Where can I find the Flash Log?	See "Flash Log messages" on page 319.
If I contact Support, what information should I send?	<ol style="list-style-type: none"><li>1. Activate Extended Support Mode.</li><li>2. Rescan the movie.</li><li>3. Send the AppScanFlashBrowser.log file from the AppScan log directory.</li></ol>

## Some Flash movies are not scanned

Flash execution is enabled, but AppScan fails to load certain Flash movies during the scan.

### Cause

There is a difference in the initialization sequence that Adobe Flash Player uses when executing a Flash movie versus that of a embedded Flash SWF file in a Flash movie.

#### Behavior 1: Flash movie

In the case of a Flash movie, the Adobe Flash Player executes the following actions:

1. Initializes the Flash Stage object
2. Calls the constructor of the Flash movie itself (Sprite object or Movie Clip object)

#### Behavior 2: Embedded SWF file

In the case of an SWF file embedded in a Flash movie, the Adobe Flash Player executes the following actions:

1. Calls the constructor of the Flash movie itself (Sprite object or Movie Clip object)
2. Initializes the Flash Stage object

It follows that any embedded Flash movie that refers to the Stage object in its constructor will run into a “null pointer exception”, because the Stage is not initialized at that point.

In order to crawl SWF files during a scan, AppScan Standard loads those files in its own Flash container. While this should not influence the way the Flash files behave, due to the inconsistency mentioned above, when the movie is loaded into the AppScan Standard container, the Adobe Flash Player will default to Behavior 2 (instead of the expected Behavior 1). If the movie includes any reference in its constructor to the Stage object, AppScan Standard will find a null pointer and will be unable to load the movie.

### Workaround

Given the current Adobe Flash Player functionality, the only workaround for this issue is to make a small change to the SWF files of the site being scanned. This change will not affect the functionality of the Flash movie, and will not pose any security risk for the SWF file in question.

#### Code example:

Typical structure of the problematic SWF file:

```
package {  
  
    import flash.display.*;  
    import flash.events.*;  
  
    public class TestSample extends MovieClip {  
  
        public function TestSample(){  
  
            // Begin initialization tasks  
            // There may be one or more references to the Stage object here  
  
            // For example: stage.addEventListener(MouseEvent.CLICK,MouseClicked);  
  
            // End of initialization tasks  
        }  
    }  
}
```

```

    // other functions – no change required
}
}

}

```

The solution is to copy the initialization tasks:

```

package {

import flash.display.*;
import flash.events.*;

public class TestSample extends MovieClip {

    public function TestSample(){
        this.addEventListener(Event.ADDED_TO_STAGE, solutionToFlashProblem);
    }

    private function solutionToFlashProblem(e:Event):void
    {

        // Begin initialization tasks
        // There may be one or more references to the Stage object here
        // For example: stage.addEventListener(MouseEvent.CLICK,MouseClicked);

        // End of initialization tasks

    }
    // other functions – no change required
}
}

```

All that has been done is to copy the content of the constructor into a callback function that will be invoked when the current class is added to the Stage object. When that happens, the Stage object will be initialized and the Flash Player will therefore follow Behavior 1.

## **Restore Adobe Flash Player settings**

### **About this task**

After scanning with Flash Execution active, Flash Player 10.1 (and later) functionality is slightly impaired and behaves like Flash 10.0 in one respect. It is a feature of version 10.1 (unlike version 10.0) that the Flash player stops running when its window is not the active window. This feature is disabled when you scan.

When you have finished scanning you can undo the change that was made. However, be aware that next time you scan with Flash Execution active the setting will be changed back again.

### **Procedure**

1. Open the Flash Player folder. By default this is usually:  
C:\WINDOWS\system32\Macromed\Flash
2. Locate the file mms.cfg and open it with a text editor (such as Microsoft Notepad).
3. Locate the line:  
FullFramerateWhenInvisible=True
4. Delete this line (or set its value to False).
5. Save the file.
6. Restart your browser for the change to take effect.

## Scan Log messages

The following sections include explanations of Scan Log messages (View > Scan Log).

All Scan Log messages are documented in this section. When relevant, explanations and suggested user actions are included.

---

### CRWAD0201I Scan created

**Explanation:** A new scan has been created. Any previous open scan data has been discarded.

**User response:** n/a

---

### CRWAD0202I Starting Explore

**Explanation:** Starting the Explore stage of the scan.

**User response:** n/a

---

### CRWAD0203I Crawling

**Explanation:** Crawling the application as part of the Explore stage.

**User response:** n/a

---

### CRWAD0204I Analyzing Explore results

**Explanation:** Analyzing Explore results to create tests.

**User response:** n/a

---

### CRWAD0205I Starting Test

**Explanation:** Starting the Test stage of the scan.

**User response:** n/a

---

### CRWAD0206I Sending Starting URLs

**Explanation:** Sending the Starting URL, and any multi-step operations that were recorded.

**User response:** n/a

---

### CRWAD0214I Starting Manual Explore

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0215I Multiphase scanning: Starting phase <#>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0216I Scan stopped; scan duration: <time>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0217I Explore and Test data cleared

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0218I Test data cleared

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0219I Time limit reached. Stopping scan...

**Explanation:** The time limit set for Scan Expert's Explore stage was reached, so it was stopped and Scan Expert proceeded to the evaluation stage. (If AppScan is configured to apply changes automatically and then proceed to the main scan, it will do so. "Stopping scan" refers only to Scan Expert's Explore stage.)

**User response:** To change the time limit go to Scan Configuration > Scan Expert.

---

### CRWAD0220I Starting Flash Execution

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0221I Supported Flash Player (Version 9.0.124.0 or later, for Internet Explorer) not installed. Flash Execution will not run.

**Explanation:** The scan is configured to include Flash file execution, but a supported version of the Adobe Flash Player is not installed on this machine, therefore Flash files will not be executed.

**User response:** If you want AppScan to execute Flash files during scanning, you must install a supported version.

You can download the latest Adobe Flash Player from <http://get.adobe.com/flashplayer/>

---

### CRWAD0222I Flash Player not configured. Flash Execution will not run.

**Explanation:** The scan is configured to include Flash file execution, but the Adobe Flash Player is not configured to work with AppScan, therefore Flash files will not be executed during the scan.

**User response:** If you want AppScan to execute Flash

## CRWAD0301I • CRWAD0403I

files during scanning, an administrator must configure the Flash Browser. See “Flash Player configuration” on page 7.

---

### CRWAD0301I Visited URL: <URL>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0302I Link <URL> extracted from dynamic content in URL <URL>

**Explanation:** A URL has been extracted for testing from JavaScript or Flash content.

**User response:** n/a

---

### CRWAD0303I Skipping URL (due to extension): <URL>

**Explanation:** The scan is configured to exclude files with this extension, so this URL is being skipped.

**User response:** This can be changed in Configuration | Exclude Paths and Files.

---

### CRWAD0304I Skipping URL (due to exclusion): <URL>

**Explanation:** The scan is configured to exclude this URL.

**User response:** This can be changed in Configuration | Exclude Paths and Files.

---

### CRWAD0305I Skipping URL (host not included in scan): <URL>

**Explanation:** The scan is not configured to include this host.

**User response:** This can be changed in Configuration | URL and Servers.

---

### CRWAD0306I Skipping URL (path limit exceeded): <URL>

**Explanation:** n/a

**User response:** Path limit can be changed in Configuration | Explore Options.

---

### CRWAD0307I Skipping URL (depth limit exceeded): <URL>

**Explanation:** n/a

**User response:** Depth limit can be changed in Configuration | Explore Options.

---

### CRWAD0308I Skipping URL (link limit exceeded): <URL>

**Explanation:** n/a

**User response:** Link limit can be changed in Configuration | Explore Options.

---

### CRWAD0309I Created test <ID> <Issue Type Name> for URL: <URL> <param>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0310I Visited page: [URL]

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0311I Visited page: <URL>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0312I No decoding service was found for the domain: <Host>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0313I A WebSphere Portal decoding service URL <URL> was located.

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0401I Login request detected: <URL>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0402I Logout request detected: <URL>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD0403I Session Identifier detected; name = <name>; value = <value>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0404I Session Identifier value refreshed; name = <name>; value = <value>**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0405I Session expired**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0406I Performing login**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0407I In-session pattern not detected**

**Explanation:** n/a

**User response:** The in-session pattern is defined in Configuration | Login Management | Details.

---

**CRWAD0408I Operation timed out**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0409I Communication error**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0410I Unnecessary request removed: [URL]**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0411I JavaScript execution in recorded login enabled**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0412I Parameter [name] should be tracked in login, but is predefined as untracked.**

**Explanation:** The specified parameter occurs in the login sequence, and AppScan wanted to add it to the list of parameters and cookies (**Scan Configuration > Parameters and Cookies**) and set it to "Tracked" (to track it during the scan). However it is already listed, and is configured as "Do not track", so AppScan has not changed its configuration.

**User response:** You may want to manually change the status of this parameter to "Tracked".

---

**CRWAD0413I Cookie [name] should be tracked in login, but is predefined as untracked.**

**Explanation:** The specified cookie occurs in the login sequence, and AppScan wanted to add it to the list of parameters and cookies (**Scan Configuration > Parameters and Cookies**) and set it to "Tracked" (to track it during the scan). However it is already listed, and is configured as "Do not track", so AppScan has not changed its configuration.

**User response:** You may want to manually change the status of this cookie to "Tracked".

---

**CRWAD0414I New tracked parameter [name] added:**

**Explanation:** The specified parameter has been added to the list of parameters and cookies (**Scan Configuration > Parameters and Cookies**), and defined as "Tracked", so it will be tracked during scanning.

**User response:** n/a

---

**CRWAD0415I New tracked cookie [name] added:**

**Explanation:** The specified cookie has been added to the list of parameters and cookies (**Scan Configuration > Parameters and Cookies**), and defined as "Tracked", so it will be tracked during scanning.

**User response:** n/a

---

**CRWAD0416I In-Session Detection format set to: [pattern format]**

**Explanation:** The format for the pattern which AppScan searches for to verify that it is in-session (see **Scan Configuration > Login Management > Details**), has been changed.

**User response:** n/a

---

**CRWAD0417I In-Session Detection pattern set to: [pattern]**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0418I In-Session URL set to: [URL]**

**Explanation:** The URL in the login sequence which AppScan uses to verify that it is logged-in, has been set to this URL (**Scan Configuration > Login Management > Details**).

**User response:** n/a

## CRWAD0419I • CRWAD0601I

---

**CRWAD0419I** Login username set to: '<name>' = '<value>'

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0420I** Login password set to: 'name' = 'value'

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0421I** Logout page detection pattern set to: 'pattern'

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0501I** Test <ID> (<name>) is POSITIVE on: <URL> <param>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0502I** Test <ID> (<name>) is negative on: <URL> <param>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0503I** Test request <URL> failed due to communication error: <Error description>

**Explanation:** Possible reasons for the five possible error messages:

- Unable to connect
  - The remote host is actively refusing the connection
  - The remote host is down
  - The network is unavailable
- Connection timed out
  - No response was received from the server within the time limit
- Connection closed (remotely)
  - The application tried to set KeepAlive on a connection that had already timed out
  - The connection was reset by the remote peer
- Connection closed (locally)
  - The connection was aborted by the underlying socket provider
  - The overlapped operation was aborted due to the closure of the socket
  - There are too many open sockets in the underlying socket provider
- Unknown

- All other causes

**User response:** n/a

---

**CRWAD0504I** URL discovered in test response: <URL>; Adding to unvisited URLs

**Explanation:** The new URL will be explored in the next phase of the scan, if there is one. If the phase limit has been reached, it will appear in the results as unvisited.

**User response:** n/a

---

**CRWAD0505I** Test <ID> (<name>) on <URL> was filtered by Adaptive test.

**Explanation:** n/a

**User response:** Adaptive test filtering can be changed in Configuration | Test Options.

---

**CRWAD0506I** Sending test <ID> (<name>) on: <URL> <param>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0507I** Page analysis started on [URL]

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0508I** Page analysis completed on [URL]

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0509I** Page analysis failed on [URL] with error: (error)

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0510I** Malware test <URL> result: Not Vulnerable (<short description>)

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0601I** Cannot connect to host: <server name>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0602I** Connection established with host:  
<server name>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0603I** Stopping scan because required host(s)  
stopped responding

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0604I** Stopping scan due to out-of-session  
detection

**Explanation:** AppScan has detected that it is  
out-of-session, based on the pattern defined in  
Configuration | Login Management | Details, and was  
unable to log in. The scan is therefore being stopped.

**User response:** n/a

---

**CRWAD0605I** AppScan has detected that it is  
out-of-session.

**Explanation:** AppScan has detected that it is  
out-of-session, based on the pattern defined in  
Configuration | Login Management | Details.

**User response:** n/a

---

**CRWAD0606I** Cannot connect to AppScan Enterprise.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0607I** SSL configuration mismatch on host:  
{1}, defaulting to SSL protocol: {0}.

**Explanation:** When AppScan is unable to connect to  
the host using the most secure of the SSL protocols  
selected by the user, it attempts to use the other  
selected protocols, starting with the next most secure  
option. This message indicates that a successful match  
was found. This issue can arise when the operating  
system doesn't support an SSL protocol that is  
supported by AppScan and the host.

**User response:** Installing service packs or using a  
different operating system might resolve this mismatch  
issue.

---

**CRWAD0701I** Starting Manual Test

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0702I** Security issue added from Manual  
Test

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0801I** Scan saved: <full path>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0802I** Security issue <Issue Type Name>  
deleted by user

**Explanation:** This issue was deleted by the user  
selecting it, right-clicking on it, and choosing Delete.

**User response:** n/a

---

**CRWAD0803I** Retesting security issue <Issue Type  
Name>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0804I** Re-test not supported for Test <ID>  
[<name>]

**Explanation:** The Re-test feature supports only  
black-box tests.

**User response:** In order to re-send this test you must  
run the appropriate module (such as JSA or Test for  
Malware) again.

---

**CRWAD0805I** Loading scan from file [full path].  
Scan created in [version], build [build  
number]

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0806I** Security issue <Issue Type Name> set  
as vulnerable by user

**Explanation:** n/a

**User response:** n/a

---

**CRWAD0807I** Security issue <Issue Type Name> set  
as not vulnerable by user

**Explanation:** n/a

**User response:** n/a

## CRWAD1001I • CRWAD1017I

---

### CRWAD1001I Starting Scan Expert

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1002I Scan Expert finished

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1003I Starting Scan Expert Evaluation

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1004I Scan Expert Evaluation finished

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1005I Automatically applying Scan Expert test results

**Explanation:** Automatically updating the scan configuration based on Scan Expert test results.

**User response:** n/a

---

### CRWAD1006I Finished applying Scan Expert test results

**Explanation:** Finished automatically updating the scan configuration based on Scan Expert test results.

**User response:** n/a

---

### CRWAD1007I Scan Expert Analysis evaluated: <name>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1008I Scan Expert Analysis failed to evaluate: <name>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1009I Applying Scan Expert Recommendation: <name>

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1010I Scan Expert: Explore Optimization: Pausing scan to analyze...

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1011I Scan Expert: Explore Optimization: Automatically started

**Explanation:** The Explore Optimization module is configured to start automatically when you run a scan

**User response:** This setting can be changed in Tools > Explore Optimization: Configure.

---

### CRWAD1012I Scan Expert: Explore Optimization: Manually started

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1013I Scan Expert: Explore Optimization: Maximum repeats per iteration reached. Resuming scan to gather more URLs

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1014I Scan Expert: Explore Optimization: Activating High Performance Mode for this phase

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1015I Scan Expert: Explore Optimization: Phase complete. For next phase setting URL limit to {0}

**Explanation:** n/a

**User response:** n/a

---

### CRWAD1016I Scan Expert: Explore Optimization: Restarting scan...

**Explanation:** Following configuration changes made by the Explore Optimization module, it was necessary to delete the existing Explore data and start the Explore stage again.

**User response:** n/a

---

### CRWAD1017I Scan Expert: Explore Optimization: Resuming scan...

**Explanation:** No configuration changes were made that required clearing the existing data before continuing. The scan is being resumed from where it left off.

**User response:** n/a

---

**CRWAD1018I Scan Expert: Explore Optimization: Not enough URLs (should be above {0})**

**Explanation:** The Switch Complexity setting defines how many URLs must contain the same rewritten segment before the Explore Optimization module will attempt to define that segment as a parameter and rerun the Explore stage. This threshold was not reached.

**User response:** You may want to reduce the Switch Complexity setting in Tools | Extentions | Explore Optimization: Configure.

---

**CRWAD1019I Scan Expert: Explore Optimization: Aborting...**

**Explanation:** Explore Optimization was aborted by the user.

**User response:** n/a

---

**CRWAD1020I Scan Expert: Explore Optimization: Step 1, identifying navigational parameters...**

**Explanation:** The Explore Optimization module is attempting to identify navigational parameters. If it does so it will set their Redundancy Tuning to the strictest setting.

**User response:** n/a

---

**CRWAD1021I Scan Expert: Explore Optimization: Analyzing {0} unique URLs**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1022I Scan Expert: Explore Optimization: Time allowed for analysis exceeded**

**Explanation:** The time allowed for Explore Optimization to run was exceeded.

**User response:** You can change this setting in Tools | Extensions | Explore Optimization: Configuration.

---

**CRWAD1023I Scan Expert: Explore Optimization: Analysis complete**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1024I Scan Expert: Explore Optimization: Analysis failed**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1025I Scan Expert: Explore Optimization: Applying configuration changes...**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1026I Scan Expert: Explore Optimization: No optimizations found**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1027I Scan Expert: Explore Optimization: Redundancy Tuning Default settings changed**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1028I Scan Expert: Explore Optimization: Found rewrite rule: {0}**

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1029I Scan Expert: Explore Optimization: Found navigational parameter '{0}'**

**Explanation:** The Explore Optimization module found a navigational parameter. It will add it to the list of parameters and cookies, and increase its Redundancy Tuning to the strictest setting.

**User response:** n/a

---

**CRWAD1030I Scan Expert: Explore Optimization: Step 2, identifying URL rewriting...**

**Explanation:** The Explore Optimization module is attempting to identify URL rewriting. If it does so, it will create appropriate custom parameters (Scan Configuration | Parameters and Cookies | Advanced tab) and run a new Explore stage.

**User response:** n/a

---

**CRWAD1101I <Extension log messages>**

**Explanation:** This message was written to the log by an eXtension.

**User response:** eXtensions are managed in Tools | Extensions | Extension Manager.

## CRWAD1201I • CRWAD1401I

---

### CRWAD1201I Starting Explore of Multi-Step Operation Sequences

Explanation: n/a

User response: Multi-Step Operations are managed in Configuration | Multi-Step Operations.

---

### CRWAD1202I Finished Explore of Multi-Step Operation Sequences

Explanation: n/a

User response: n/a

---

### CRWAD1203I Exploring Multi-Step Operation Sequence: <name>

Explanation: n/a

User response: n/a

---

### CRWAD1204I Starting Test of Multi-Step Operation Sequences

Explanation: n/a

User response: n/a

---

### CRWAD1205I Finished Test of Multi-Step Operation Sequences

Explanation: n/a

User response: n/a

---

### CRWAD1206I Testing Multi-Step Operation Sequence: <name>

Explanation: n/a

User response: n/a

---

### CRWAD1207I Optimization found for sequence <sequence name>: Should play requests #<index> to #<index> before request #<index>

Explanation: AppScan has determined that the specified steps in the sequence can be omitted without affecting the outcome of the sequence.

User response: If AppScan is wrong to ignore these steps, you can disable the Allow Play Optimization check box in Configuration | Multi-Step Operations.

---

### CRWAD1208I Optimization found for sequence <name>: No sequence needed before request #<index>

Explanation: AppScan has determined that all the steps before the specified step can be omitted without affecting the outcome.

**User response:** If AppScan is wrong to ignore these steps, you can disable the Allow Play Optimization check box in Configuration | Multi-Step Operations.

---

### CRWAD1209I Can't optimize multi-step [multi-step operation name] up to request [n]

**Explanation:** Multi-step optimization is active (Scan Configuration > Multi-Step Operations > Allow play optimization), therefore AppScan attempts to remove redundant requests from the end of the sequence, to reduce scan time. This message indicates that in this attempt all requests after number [n] were removed. The optimization attempt was unsuccessful.

**User response:** You can review and edit the sequence yourself in Scan Configuration > Multi-Step Operations.

---

### CRWAD1301I Starting Result Expert automatically

Explanation: n/a

User response: You can control whether Result Expert runs automatically at the end of a scan in Tools | Options.

---

### CRWAD1302I Finished automatic Result Expert

Explanation: n/a

User response: n/a

---

### CRWAD1303I Starting Result Expert module : <name>

Explanation: n/a

User response: n/a

---

### CRWAD1304I Result Expert module finished: <name>

Explanation: n/a

User response: n/a

---

### CRWAD1305I Result Expert module failed: <name>

Explanation: n/a

User response: n/a

---

### CRWAD1401I <#> web service request(s) were explored

Explanation: n/a

User response: n/a

---

**CRWAD1601I** Glass box detected the parameter [parameter name] in [URL]

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1602I** Glass box added the new URL [URL]

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1603I** Cannot connect to glass box server: [server name]

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1604I** Glass box test <ID> (<name>) is POSITIVE on: <URL> <param>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1605I** Glass box test <ID> (<name>) is negative on: <URL> <param>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1606I** Glass box test request <URL> failed due to communication error: <Error description>

**Explanation:** Possible reasons for the five possible error messages:

- Unable to connect
  - The remote host is actively refusing the connection
  - The remote host is down
  - The network is unavailable
- Connection timed out
  - No response was received from the server within the time limit
- Connection closed (remotely)
  - The application tried to set KeepAlive on a connection that had already timed out
  - The connection was reset by the remote peer
- Connection closed (locally)
  - The connection was aborted by the underlying socket provider
  - The overlapped operation was aborted due to the closure of the socket
  - There are too many open sockets in the underlying socket provider
- Unknown

- All other causes

**User response:** n/a

---

**CRWAD1607I** Sending glass box test <ID> (<name>) on: <URL> <param>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1608I** Fetch glass box sink results

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1609I** Fetch glass box unreferenced parameter results

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1610I** Glass box: Started unreferenced parameter monitoring.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1611I** Glass box: Stopped unreferenced parameter monitoring.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1612I** Glass box: Started sink monitoring.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1613I** Glass box: Stopped sink monitoring.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD1614I** Detected glass box agent host operating system, or operating system version, or web server.

**Explanation:** An informational message indicating that AppScan detected this information about the glass box agent host.

**User response:** n/a

---

**CRWAD1615I** Action-based player started.**Explanation:** n/a**User response:** n/a

---

**CRWAD1616I** Action-based player finished  
[SUCCESS]**Explanation:** n/a**User response:** n/a

---

**CRWAD1617I** Action-based player finished [FAILED]

---

## AppScan Log messages

The following sections include explanations of AppScan Log messages. (Help > AppScan Log)

Currently only the more important AppScan Log messages are documented in this section. When relevant, explanations and suggested user actions are included.

---

**CRWAD3000E** Bad AppScan registry entry <key name>.**Explanation:** n/a**User response:** n/a

---

**CRWAD3002I** Advanced Configuration option set.  
ID: <ID> = value <value>.**Explanation:** n/a**User response:** n/a

---

**CRWAD3003E** Advanced Configuration option not set due to error. ID: <#>, value <value>.

**Explanation:** AppScan was unable to apply an Advanced Configuration change made by the user due to incorrect format (ID and value specified).

**User response:** Locate the Advanced Configuration setting by its ID, and correct the value.

---

**CRWAD3010I** AppScan version: <#> (Build number: <#>, Security rules version: <#>)**Explanation:** n/a**User response:** n/a

---

**CRWAD3100I** New AppScan process started.**Explanation:** n/a**User response:** n/a

---

**CRWAD3101I** AppScan exited.**Explanation:** n/a**User response:** n/a

---

**Explanation:** n/a**User response:** n/a

---

**CRWAD1618I** Action-based player failed to log in, rolling back to request-based login.

**Explanation:** AppScan failed to log in using the action-based log in recording, and from now on will use the request-based version.

**User response:** Try reviewing or re-recording the action-based login sequence to resolve the issue.

---

**CRWAD3102I** AppScan proxy port is <port number>.**Explanation:** n/a**User response:** n/a

---

**CRWAD3103I** AppScan Flash proxy port is <port number>.**Explanation:** n/a**User response:** n/a

---

**CRWAD3104I** Starting Explore stage on host(s): <host(s)>.**Explanation:** n/a**User response:** n/a

---

**CRWAD3105I** Resuming Explore stage.**Explanation:** n/a**User response:** n/a

---

**CRWAD3106I** Explore stage stopped by user.**Explanation:** n/a**User response:** n/a

---

**CRWAD3107I** Explore stage completed.**Explanation:** n/a**User response:** n/a

---

**CRWAD3108I** Clearing Explore and Test stage data.**Explanation:** n/a

**User response:** n/a

---

**CRWAD3109I** Creating a new scan.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3200I** Starting Test stage on host(s):  
<host(s)>.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3201I** Resuming Test stage.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3202I** Test stage stopped by user.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3203I** Test stage completed.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3204I** Clearing Test stage data.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3205I** Importing Explore data from: <path>.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3300W** Import Explore Data failed, from file:  
<file path>.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3407E** License verification failed! RCL error  
message (if any): <message>

**Explanation:** Possible causes: 1) License expired, 2) No floating license available, 3) Unable to connect to license server for more than 3 days. The RCL (Rational Common Licensing) message may contain more details of the cause of the problem.

**Note:** In some cases the RCL message may not be available.

**User response:** n/a

---

**CRWAD3409W** No license for host: <host>.

**Explanation:** n/a

**User response:** Update your license (Help > License) to include the specified host, or remove the host from your list of additional hosts (Scan Configuration > URL and Servers > Additional Servers and Domains).

---

**CRWAD3410W** License server unavailable. Running  
in Disconnected Mode allowed for up to  
3 days. RCL message: <message>

**Explanation:** When AppScan is unable to connect to the license server to verify the validity of a floating license, it can continue to run in Disconnected Mode for up to 3 days. After that time, if it is still unable to connect to verify the license, scanning will be disabled for all sites except demo.testfire.com. The RCL (Rational Common Licensing) message may contain more details of the cause of the problem.

**User response:** n/a

---

**CRWAD3411I** License successfully checked out.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3500E** An AppScan internal error occurred.  
A dump file was created in: <file path>. Contact Support.

**Explanation:** A critical internal error has occurred, which caused AppScan to terminate. The memory dump file may help to determine the cause of the problem.

**User response:** Contact Support and provide the dump file any other relevant information.

---

**CRWAD3501E** An AppScan internal error occurred.  
Attempting to save the scan.

**Explanation:** A critical internal error occurred. AppScan attempted to save the scan before closing. If the save was successful, the path of the saved scan will appear later in the log.

**User response:** n/a

---

**CRWAD3502E** Auto-Recover save succeeded.  
Recovered scan saved at: <file path>

**Explanation:** AppScan succeeded in saving a scan after a critical error. The scan is located in the path indicated.

**User response:** Try to load the saved scan file and continue the scan.

## CRWAD3503E • CRWAD3813E

---

**CRWAD3503E** Attempt to save the scan before terminating failed.

**Explanation:** AppScan was unable to save an Auto-Recover scan file before terminating.

**User response:** Contact Support.

---

**CRWAD3600E** Rules file is corrupted.

**Explanation:** The security rules file, which defines AppScan tests, is corrupted.

**User response:** Contact Support.

---

**CRWAD3601E** User-Defined rules file is corrupted.

**Explanation:** The security rules file, which defines user-defined tests, is corrupted.

**User response:** Contact Support.

---

**CRWAD3602W** Some security issues in the loaded scan were deleted due to missing rules.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3707I** Server <name> is not responding.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3708I** Server <name> is responding again.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3709I** Proxy <name> is not responding.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3710I** Proxy <name> is responding again.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3800E** Fatal error: scan file <file path> is corrupt.

**Explanation:** Unable to load scan file.

**User response:** Contact Support.

---

**CRWAD3801E** AppScan was unable to load the session. The session file may be corrupt or invalid.

**Explanation:** The scan file was successfully opened, but AppScan failed to load a session from it.

**User response:** Contact Support.

---

**CRWAD3806I** Saving scan to file: <full path>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3807I** Finished Saving scan to file: <full path>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3808I** Loading scan from file: <full path>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3809I** Finished Loading scan from file: <full path>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3810I** Loading scan with version <original version>, updating to current version <current version>.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3811E** Loading scan with version {0}, later than current version {1}!

**Explanation:** The loaded scan was created in a version (specified), that is later than the current version (specified). The scan may fail to load.

**User response:** n/a

---

**CRWAD3813E** Failed to save scan <file path> due to insufficient disk space.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3814E** Unable to save scan <file path>, since the scan file is a read only file.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3816E** AppScan was unable to load the scan due to insufficient disk space. Please free <#> MB on drive <drive letter> and try again.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD3817W** Rebuilding SDK result database from engine database.

**Explanation:** The SDK result database was not found. AppScan is rebuilding it from the engine database.

**User response:** n/a

---

**CRWAD3818E** Loading scan with DB version {n}, this version is unsupported.

**Explanation:** The scan was saved using an unsupported AppScan database version, and cannot be loaded in the current AppScan version.

**User response:** n/a

---

**CRWAD4105E** Critical thread (<thread name>, tid: <thread id>) died unnaturally.

**Explanation:** A critical internal error occurred.

**User response:** Contact Support.

---

**CRWAD4106E** Non-critical thread (<thread name>, tid: <thread id>) died unnaturally.

**Explanation:** A critical internal error occurred.

**User response:** Contact Support.

---

**CRWAD4300I** Starting AppScan Extended Support mode.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD4301I** AppScan Extended Support mode is already <On/Off>.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD4302I** AppScan Extended Support mode stopped.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD4303I** Failed to pack Extended Support information into file: <full path>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD4500E** AppScan stopped due to insufficient disk space. [ Free: <#>M , Required: <#>M ]

**Explanation:** n/a

**User response:** n/a

---

**CRWAD4501E** AppScan stopped because memory usage reached the predefined limit. [ MemUsage: <#>K, MaxMemUsage: <#>K ]

**Explanation:** n/a

**User response:** Try increasing the memory limit (Tools > Advanced Options), or contact Support.

---

**CRWAD4502E** AppScan stopped because the system is running low on virtual memory. [ VM: <#>K, AvailableVM: <#>K ]

**Explanation:** n/a

**User response:** Try closing other applications to free memory.

---

**CRWAD4503E** Low system memory - AppScan may not continue to run properly. [ MemUsage: <#>K , TotalPhysicalMem: <#>K ]

**Explanation:** n/a

**User response:** Try closing other applications to free memory.

---

**CRWAD5002E** Importing XML Configuration failed. Parse error (if available): %s

**Explanation:** Failed to send the configuration to the engine. Unable to proceed with scan.

**User response:** Contact Support.

## CRWAD5003I • CRWAD5407E

---

**CRWAD5003I** Applying updated scan configuration to engine.

**Explanation:** n/a

**User response:** n/a

---

**CRWAD5004E** Parsing XML configuration failed.

**Explanation:** Failed to send the configuration to the engine. Unable to proceed with scan.

**User response:** Contact Support.

---

**CRWAD5100E** AppScan Severe Error: %s

**Explanation:** n/a

**User response:** n/a

---

**CRWAD5101E** Software exception occurred:  
<message> Call Stack: <stack>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD5400I** Glass box server <URL> versions:  
Agent version: <#> ; GBootStrap  
version: <#> ; Agent rules version: <#>

**Explanation:** n/a

**User response:** n/a

---

**CRWAD5401E** Cannot connect to glass box server {0}  
due to: Either server on which agent is  
installed, or agent application URL, is  
unreachable.

**Explanation:** AppScan failed to connect with the glass box server for one of the listed reasons. The agent URL may be incorrect, or GBootStrap may not be running, or the server is down.△

**User response:** Verify that the server is running and the agent URL you entered is correct. Using your web browser to open the agent URL may help you verify that GBootStrap is running correctly. If this does not help you identify the problem, contact Support.

---

**CRWAD5402E** Cannot connect to glass box server {0}  
due to: Credentials for connecting to  
glass box agent are either not supplied  
or incorrect.

**Explanation:** The credentials given for accessing the glass box agent are not given or incorrect.△

**User response:** Supply the correct username and password, as defined during the glass box agent installation.△

---

**CRWAD5403E** Cannot connect to glass box server {0}  
due to: No glass box agent is installed  
on the glass box agent URL specified.

**Explanation:** The URL leads to a site that is not the glass box GBootStrap root.

**User response:** Verify the URL. By default it is of the form:

`http://</>/</><server_name>:<port_number>/GBootStrap/`

---

**CRWAD5404E** Cannot connect to glass box server {0}  
due to: The glass box agent  
instrumentation is not running.

**Explanation:** The glass box agent is not instrumented on the specified server.

**User response:**

1. Stop the server and restart it, using the desktop shortcut (if one was created when you installed the agent). △
2. Reinstall the glass box agent manually (see “Installing the glass box agent” on page 136).

---

**CRWAD5405E** Cannot connect to glass box server {0}  
due to: Client version {1} is older than  
either glass box agent file version {2} or  
glass box agent application version {3}.

**Explanation:** The version of AppScan on the local machine is not up to date.△

**User response:** Install the latest version of AppScan.△

---

**CRWAD5406E** Cannot connect to glass box server {0}  
due to: Both glass box agent file version  
{1} and glass box agent application  
version {2} are older than glass box  
client version {3}

**Explanation:** The glass box agent version is not up to date.△

**User response:** Install the latest version of the glass box agent on the server machine.△

---

**CRWAD5407E** Cannot connect to glass box server {0}  
due to: Glass box agent application  
version {1} older than glass box client  
version {2}

**Explanation:** You may have installed the glass box agent manually and omitted to deploy GBootStrap.war.

**User response:** Follow exactly the instructions for manually installing the glass box agent on your specific server. See “Installing the glass box agent” on page 136.

**CRWAD5408E** Cannot connect to glass box server {0} due to: Glass box agent file version {1} older than glass box client version {2}

**Explanation:** Glass box agent JAR file version is older than AppScan version△.

**User response:** You may have simply omitted to restart the server after installing the agent. Try restarting the server. If this does not help, reinstall the glass box agent.

**CRWAD5409E** Cannot connect to glass box server {0} due to: Glass box agent rules version {1} older than glass box client rules version {2}

**Explanation:** The glass box rules are not up to date.

**User response:**

1. Go to **AppScan > Help > Check for Updates**.
  - If a download is available, it will be automatically downloaded to your machine and installed.
  - If a download is not available, check Fix Central for the latest version and install it.
2. To update the agent rules on the △server△, open **Tools > GB Agent Management**, double-click on the server you need, and in the dialog box that opens, click **Update Rules**.
3. Restart the application server, for the change to take effect.

## Flash Log messages

The following table describes using the Flash Log for troubleshooting.

The Flash Log contains messages that may help you with troubleshooting. It is located at:

...IBM\AppScan Standard\Logs\AppScanFlashBrowser.log

## Error messages

Error message	Possible user response
Explore Failed: Couldn't reach initial state!	Try increasing Configuration > Advanced Configuration > Load movie wait time
Movie (NAME) load failed! (TIME) milliseconds timeout, (COUNT) try(s).	Try increasing Configuration > Advanced Configuration > Load movie wait time
current state failed, since Movie is not ready!	Try increasing Configuration > Advanced Configuration > Load movie wait time
Exception while sending status! (MESSAGE). Microsoft .NET framework Hotfix required! Contact IBM Support. Hotfix reference <a href="http://support.microsoft.com/kb/971521">http://support.microsoft.com/kb/971521</a>	Contact Support to get this hotfix.
Movie(NAME) version (NUMBER) not supported!	Recompile movie to version 7 or later.
Exception in Document Completed at URL(URL)! (MESSAGE)	Contact Support for the correct version of mshtml.dll

**CRWAD5410E** Cannot connect to glass box server {0} due to: Glass box agent rules were updated but restart is needed to complete the update process.

**Explanation:** The glass box agent rules were updated, and now the server must be restarted to load the new instrumentation rules.

**User response:** Restart the server.

**CRWAD5411E** Cannot connect to glass box server {0} due to: Glass box client rules version {1} older than glass box agent rules version {2}

**Explanation:** The glass box agent rules were updated, and now the server must be restarted to load the new instrumentation rules△.

**User response:**

1. Go to **AppScan > Help > Check for Updates**.
  - If a download is available, it will be automatically downloaded to your machine and installed.
  - If an automatic download is not available, check Fix Central (<http://www.ibm.com/support/fxcentral/>) for the latest version and install it.
2. Restart the application server, for the change to take effect.

Error message	Possible user response
Flash Player not Found!	Refer to "Flash Player upgrade" on page 7
Flash Player version not Supported!	Refer to "Flash Player upgrade" on page 7
Flash Player not configured!	Refer to "Flash Player configuration" on page 7
Exception while trying to load SWF file!	Refer to "Some Flash movies are not scanned" on page 303

## Warning messages

Warning message	Possible user response
Explore movie (NAME) stopped! Click limit (VALUE) reached.	Try increasing <b>Scan Configuration &gt; Explore Options &gt; Click Limit</b>
Explore movie (NAME) Stopped! Screen limit (VALUE) reached.	Try increasing <b>Scan Configuration &gt; Explore Options &gt; Screen Limit</b>
Exception while loading unsupported Flash Movie! (MESSAGE)	Republish the movie in Flash version 7 or later.
Parsing data from bridge failed due to unsupported movie(NAME)!	Republish the movie in Flash version 7 or later.
Exception while parsing data from bridge in unsupported movie(NAME) !	Republish the movie in Flash version 7 or later.
Movie (NAME) not supported!	Republish the movie in Flash version 7 or later.
Document Complete event not received from Browser!	Play the movie in a stand-alone version of Internet Explorer. If you don't see a "Done" message on the left side of the status bar, the link in the HTML page of the SWF file may be broken.
Movie (NAME) failed to load in (TIME) milliseconds for (NUMBER) try(s)! Reloading...	Try increasing <b>Configuration &gt; Advanced Configuration &gt; Load movie wait time</b>
No Browse Dialog found!	Recheck the specified strings in <b>Advanced Configuration &gt; Flash: File upload string</b> or <b>Flash: File download string</b> . Delete strings which do not identify buttons for upload or download.
Exception while trying to create file for upload! (MESSAGE)	No file for uploading has been configured. Configure one in <b>Advanced Configuration &gt; Flash: File upload path</b> .

---

## Glass box troubleshooting

Open a web page listing glass box agent messages and other troubleshooting tips.

<http://www.ibm.com/support/docview.wss?uid=swg21567723>

---

## Chapter 15. CLI

Certain AppScan functions may be performed using the Command Line Interface (i.e. by typing in a command at the command prompt, rather than using the Graphic User Interface). This may be particularly useful when you need to control AppScan automatically from within a script or batch file.

This section describes the syntax and options available using the CLI.

---

### Command structure

AppScan CLI commands consist of three parts:

1. The utility command: AppScanCMD
2. The specific command to be executed, for example: exec
3. The required option for the selected command, for example:  
/base\_scan <full path>/d <full path>

In the above example, the full command is:

```
AppScanCMD exec /base_scan <full path> /d <full path>
```

(This command will run a new scan using the configuration of the selected base scan, and save the results in the specified location.)

**Note:** Commands can be entered with no prefix (e.g. exec), or with a minus sign (e.g. -exec), or with a forward slash (e.g. /exec). For simplicity, examples in this section will all be given without any prefix.

**Note:** Command options must be prefixed by *either* a minus sign (e.g. -base\_scan) *or* a forward slash (e.g. /base\_scan). For simplicity, the examples following will all be prefixed with a forward slash.

---

## Commands

This section describes the application specific commands that can be run using the CLI.

### Exec command

The exec command creates a new scan with the starting URL specified (in /starting\_url, /base\_scan, or /scan\_template), runs it, and saves it. It can also optionally be used to generate and save a report of the scan.

To run the exec command, type exec, ex, or e at the command prompt, followed by the required command options, as described below.

**Note:** If no command is specified, the exec command runs by default.

## Parameters

The following parameters can be included in the exec command.

Parameter	Explanation
/starting_url   /surl   /su <full_path>	Sets the starting URL for the scan. If the starting URL is defined in the scan template or base scan you do not need to define it here.
/base_scan   /base   /b <full_path>	Specifies a source scan (must include full path), whose configuration will be used for the new scan.
/dest_scan   /dest   /d <full_path>	Specifies the destination where the new scan will be saved (must include full path). If no path is specified, the scan is saved to a Temp folder, and AppScanCMD will notify you of its exact location and filename.
/scan_template   /stemplate   /st <full_path>	Specifies the scan template file.
/old_host   /ohost   /oh <full_path> /new_host   /nhost   /nh <full_path>	These two parameters enable you to "search and replace" one host with another, in a base scan.
/login_file   /lfile   /lf <full_path>	Imports a saved login sequence.
/multi_step_file   /mstepfile   /mf <full_path>	Imports a multi-step operations file.
/manual_explore_file   /mexplorefile   /mef <full_path>	Imports a Manual Explore (EXD) file. <b>Note:</b> As of AppScan Standard Version 9.0.1, EXD files include response data. To import this data, add flag /ir. If the flag is <i>not</i> added, the response data will not be imported, and instead an Explore stage will be run (the saved requests will be sent) to gather <i>new</i> responses to analyze for testing.
/policy_file   /pfile   /pf <full_path>	Imports a test policy file.
/additional_domains   /adomains   /ad <domain>	Defines domains other than that of the starting URL to be included in the scan. If there are more than one additional domains, separate them with a comma or add multiple instances of the parameter.
/report_file   /rf <full_path>	Specifies the destination and name for the generated report (must include full path).  This field is optional. If not set, no report is generated.  Note that if /rt is defined as rc_ase no report file is needed, as the output is published to AppScan Enterprise.
/report_type   /rt <xml pdf rtf txt html rc_ase>	Specifies the report format. Default is XML.  rc_ase refers to AppScan Enterprise reports and the output will be published to AppScan Enterprise using the existing settings (to see these click File > Export > Publish to AppScan Enterprise > Connection Settings).
/min_severity   /ms <low medium high informational>	Specifies the minimum result severity to include in reports. (non-xml reports only).  Default is "low".
/test_type /tt <All   Application   Infrastructure   ThirdParty>	Specifies which type of tests to include in the report. Default is "All".

## Flags

The following flags can be included in the exec command. Including them is the equivalent of setting them from False to True.

Flag	Explanation
/continue   /c	Continue the scan.
/explore_only   /eo	Run an Explore stage only.
/include_responses   /ir	When importing Manual Explore data (EXD file) that includes response data, include the responses (use with /mef). <b>Note:</b> As of AppScan Standard Version 9.0.1, EXD files include response data. If you add this flag, and the file includes response data, it is used when analyzing for testing. If the file does <i>not</i> include response data, an Explore stage will be run (the saved requests will be sent) to gather <i>new</i> responses to analyze for testing.
/merge_manual_explore_requests   /mmr	Apply Explore-stage Redundancy Tuning settings to Manual Explore data, to help avoid duplicate requests (use with /mef).
/multi-step   /mstep	Test multi-step operations only.
/scan_log   /sl	Display the scan log during the scan.
/test_only   /to	Run a Test stage only.
/verbose   /v	Include progress lines in the output.

Below are some examples of complete commands.

### Example 1

This command will start a scan with the specified starting URL, using the Regular Scan template.

```
exec
-starting_url http://demo.testfire.net.scan
```

### Example 2

This will start an Explore stage only with the specified starting URL, using the Regular Scan template.

```
exec
-starting_url http://demo.testfire.net.scan
-explore_only
```

### Example 3

This scan includes a manual explore, multi-step operations, a recorded login and a test policy.

```
exec
-scan_template "D:\demo.testfire.net.scant"
-dest_scan "D:\demo.testfire.net.scan"
-manual_explore_File "D:\ManualExplore.exd"
-multi_step_file 'D:\MyMultistepOperation.seq'
-login_file 'D:\LoginSequence.login'
-policy_file 'D:\MyTestPolicy.policy'
```

### Example 4: Additional domains

Two or more additional domains can be defined in a single -additional\_domains parameter, separated by commas, or in separate parameters.

```
exec
-dest_scan "D:\demo.testfire.net.scan"
-scan_template "D:\demo.testfire.net.scant"
-manual_explore_file "D:\ManualExplore.exd"
```

```
-additional_domains "demo.testfire.net1,demo.testfire.net2,demo.testfire.net3"
-additional_domains "demo.testfire.net4"
-additional_domains "demo.testfire.net5"
-scan_log
```

OR

```
exec
-dest_scan "D:\demo.testfire.net.scan"
-scan_template "D:\demo.testfire.net.scant"
-manual_explore_file "D:\ManualExplore.exd"
-additional_domains "demo.testfire.net1"
-additional_domains "demo.testfire.net2"
-scan_log
```

### Example 5: Change hosts and ports

You can change both the host and the port of the scan template or base scan.

```
exec
-scan_template "D:\demo.testfire.net.scant"
-dest_scan "D:\demo.testfire.net.scan"
-manual_explore_file "D:\ManualExplore.exd"
-old_host 'http://demo.testfire.net:80'
-new_host 'http://demo.testfire.net2:8090'
```

## Report command

The report command loads a specified scan and generates a report.

To run the report command, type report, rep, or r at the command prompt, followed by the required command options, as described below. (Base scan and destination parameters are compulsory; other parameters are optional.)

Commands	Explanation
/base_scan   /base   /b <full_path>	Specifies the source scan from which to create the report (must include full path).
/report_file   /rf <full_path>	Specifies the destination and name for the generated report (must include full path).  Note that if /rt is defined as rc_ase no report file is needed, as the output is published to AppScan Enterprise.
/report_type   /rt <xml pdf rtf txt html rc_ase>	Specifies the report format. Default is XML.  custom refers to Word reports.  rc_ase refers to AppScan Enterprise reports and the output will be published to AppScan Enterprise using the existing settings (to see these click <b>File &gt; Export &gt; Publish to AppScan Enterprise &gt; Connection Settings</b> ).
/min_severity   /ms <low medium high informational>	Specifies the minimum result severity to include in reports (non-xml reports only). Default is "Informational", meaning that all severities are included in the report.
/test_type /tt <All   Application   Infrastructure   ThirdParty>	Specifies which type of tests to include in the report. Default is "All".

## Flags

The following flags can be included in the report command.

Flag	Explanation
/verbose   /v	Include progress lines in the output.

### Example 1

This command will publish the results of the specified base scan to AppScan Enterprise.

```
report
-base_scan "D:\demo.testfire.net.scan"
-report_type rc_ase
```

### Example 2

This command will start a scan with the specified starting URL, using the Regular Scan template.

```
report
-base_scan "D:\demo.testfire.net.scan"
-report_file D:\SecurityReport.pdf
-report_type pdf
-Scan_Log
-min_severity "informational"
```

## Help command

The help command prints the command usage described in this section.

To run the help command, type help, or h at the command prompt.

## Exit Status codes

When AppScan is run within scripts or batch files, the Exit Status code indicates whether or not operation was successful.

Code	Meaning
0	Finished Successfully
1	AppScan Startup Failed
2	Command Line Error
3	License Invalid
4	Load Failed
5	Scan Failed
6	Report Failed
7	Save Failed
8	General Error

## Launching AppScan from the command line

### Procedure

From the command line, type: AppScan.exe [<filename>]

- This command runs the AppScan executable, to launch the GUI, and you may add an optional parameter to call up a .scan or .scant file.

- If you include a filename in the command, the given scan or scan template is loaded as soon as the application starts.

---

## Chapter 16. Menus, toolbars and keyboard shortcuts

This section summarizes the main menus and toolbars.

---

### File menu

Used for creating, opening, and saving scans.

Command	Click to:
New	Create a new scan.
Open	Open a saved scan or scan template.
Save	Save the current scan or scan template.
Save As	Save the current scan or scan template with a new name.
Export > Scan Results as XML	Export the complete scan results as an XML file.
Export > Scan Results as DB	Export the complete scan results as a relational database. The database option exports the results into a Firebird database structure. This is open source, and follows ODBC and JDBC standards.
Export > Cross Scan Data	If you have defined specific issues as Noise (False Positive), you can export this information for use in scans run on other workstations. (You do not need to do this on your own workstation as the information is automatically saved and applied to future scans.)
Export > Publish results to AppScan Enterprise	Exports the results of an AppScan Standard scan to AppScan Enterprise, runs a Test stage on them, and compiles the results into a report.
Export > Create job in AppScan Enterprise	Exports an AppScan Standard configuration to AppScan Enterprise. Requires AppScan Enterprise version 9.0 or later.
Import > Explore Data	Load an exported manual explore file.
Import > Cross Scan Data	If specific issues have been defined as Noise (False Positives) on a different workstation and exported, you can import this information for use in scans run on your workstation, so that these issues will not be included in the scan results.
Page Setup	Define paper size, source, orientation and margins for the Print command.
Print Preview	Open a preview window showing the Application Tree or Result List (depending on current cursor location), as they will appear when printed using the Print command.
Print	Print the current contents of the Application tree <i>and</i> the Result list. (All nodes that can be seen by scrolling up or down the Application tree and Result list are included, but the nodes that are closed on the screen will appear closed in the printout.)
Filenames	Recently used files.
Exit	Exit AppScan.

---

### Edit menu

Used for customizing scan results.

Command	Click to:
Delete	Delete selected issue or remediation.
Severity	(Issues View only) Customize the severity level of the selected issue.

Command	Click to:
State	(Issues View only) Designate the selected issue as "Noise" (meaning that although AppScan classified it as an issue, in the context of your application it is not.) Issues designated as Noise can either be removed from the results altogether, or displayed with a strikethrough (View menu > Show Issues Marked as Noise).
Priority	(Remediation View only) Change the priority level of a remediation.
Find	Find strings, IDs, HTTP code, etc., in the current scan results. (Options depend on which of the three Views is currently selected.)

**Note:** Severity and Priority are mutually exclusive; only one is visible at any time, depending on the selected view.

## View menu

Used to determine how the main window appears and what data is displayed.

Command	Click to:
Security Issues	Display Security Issues view.
Remediation Tasks	Display Remediation Tasks view.
Application Data	Display Application Data view (failed requests, visited URLs, script parameters, and so on).
Arrange By >	Select a sort method for the Result List: Severity/Result/Name. Select <b>Reverse Order</b> to list results in descending rather than ascending order.
Layout >	Select Horizontal or Vertical layout of the main window panes.
Welcome Screen	Open the AppScan Welcome Screen.
Show Issues Marked as Noise	Issues AppScan finds, that are irrelevant in the context of your application, can be marked as Noise. This menu item toggles between displaying such issues with strikethrough text, or not at all. When selected, Noise appears with strikethrough; when deselected, Noise is not shown. See "Issue state: Open or Noise" on page 175
Non-vulnerable Variants	Opens the list of variants that are defined as non-vulnerable. See "Non-vulnerable variants" on page 191
Scan Log	Open the log of all actions performed by AppScan during the current scan. See "AppScan Log" on page 274
Customize Toolbar >	Select <b>Large Icons</b> to display larger toolbar icons Select whether the toolbar shows the name of every icon, of selected icons (default), or of none of the icons. (In any case, when you roll your mouse over an icon the tool tip shows its name.)

## Scan menu

Used to control the scan.

Command	Click to:
Full Scan	Start a full scan (Explore and Test stages) or continue a paused scan.
Pause	Pause current scan (whether Full Scan, Explore Only or Test Only). You can resume the scan later. You can also save a paused scan to continue at another time.

Command	Click to:
Re-Scan >	Rerun the current scan or scan stage. Select one of the sub-menu items:  Re-Scan (Full): Delete all scan results, and run a full scan using the current configuration.  Re-Explore: Delete all scan results and run an Explore stage only using the current configuration.  Re-Test: Delete Test results and run a new Test stage using the current configuration and Explore results.
Explore Only	Run an Explore stage only, without following it with the Test stage.
Manual Explore	Explore your site manually. See "Using AppScan" on page 115
Explore web Services	Opens Generic Service Client, so you can manually explore the application. The requests and responses are then used during the Test stage.
Test Only	Run a Test stage only (or continue a Test that was paused), without first running an Explore stage. This option is active only when there are already some Explore results.
Test Multi-Step Operations Only	If you have configured one or more Multi-Step Operations (see "Multi-Step Operations view" on page 79), and they constitute a significant subset of your site that you want to scan, you can test those sequences only. For details see "Scan Multi-Step Operations Only" on page 160  Note that Scan Expert does not run automatically before this function, even if configured to run before scans. If required, run it separately before Scan Multi-Step Operations Only (Tools > Run Scan Expert Evaluation).
Re-test issues found	
Clear All Scan Data	Delete all Explore and Test results, keeping only the Scan Configuration.
Run Scan Expert Evaluation	Scan Expert evaluates whether the current configuration is optimal for the application being scanned. (See "Scan Expert" on page 134)  This option runs a full evaluation: Scan Expert will briefly explore the application, analyze its responses, and suggest changes to the configuration to get the best results.
Run Scan Expert Analysis Only	This option runs only the Analysis stage, and is active only if there are already some scan results on which to base the analysis. Scan Expert will analyze the current results to determine whether the configuration is optimal.
Scan Configuration	Define the properties of a scan. See "Scan configuration dialog box" on page 42

---

## Tools menu

Provides various reporting and customization tools, including the IBM Security PowerTools.

Command	Click to:
Report	Create a report of the current scan. See Chapter 10, "Reports," on page 197
Manual Test	Create and send a test variant manually to a selected URL. See "Manual tests" on page 189
Run Result Expert	(Active only if there are some scan results:) Run Result Expert on the current results. See "Result Expert" on page 161
Scan Scheduler	Set time and frequency for automatic run of a scan.
User-Defined Tests	Define new tests for scans. See "User-defined tests" on page 226
Glass Box Agents	Define your server-side glass box agents so scans can include glass box scanning. See "Glass box scanning" on page 135
Extensions >	

Command	Click to:
Start Pyscan	Open Pyscan to control AppScan using the Python interface. See “Pyscan” on page 268
Extension Manager	Open Extension Manager to manage add-on applications. See “Extension Manager” on page 267
<b>PowerTools &gt;</b>	
Authentication Tester	Run Authentication Tester PowerTool to brute-force authentication tests.
Connection Test	Run Connection Test PowerTool to ping websites without using the ping protocol (which is blocked by many firewalls).
EncodeDecode	Run EncodeDecode PowerTool for string encoding/decoding between various formats.
ExpressionTest	Run ExpressionTest PowerTool to check regular expressions.
HTTP Request Editor	Run HTTP Request Editor PowerTool to edit and send HTTP requests, and view site responses.
Token Analyzer	Run Token Analyzer PowerTool to analyze the randomness of session tokens.
External Tools...	Click to open a dialog box from which you can: <ul style="list-style-type: none"> <li>• Adjust the order of the PowerTools on the Tools menu</li> <li>• Add additional external programs that can be opened from the Tools menu</li> </ul>
Options	Customize AppScan operations. See “Options dialog box” on page 217

---

## Help menu

Used for accessing documentation, getting support help, and getting a new license.

Command	Click to:
AppScan Help	Open Online Help; same as F1.
AppScan Getting Started	Open the PDF format Getting Started Guide (requires Adobe Acrobat Reader).
Accessibility Controls	Adds the Accessibility menu as the first menu (before the File menu) on the menu bar.
Accessibility Documentation	Opens the online Help at the Accessibility Controls page.
AppScan Publications Library	Opens the AppScan Publications Library page on the IBM website, containing a link to the AppScan User Guide. This is a printable, PDF version of the Help file (requires Adobe Acrobat Reader).
AppScan website	Open IBM Security's AppScan website in your browser.
License	Install or request a new license. Import AppScan Enterprise license permissions.
Support >	Go into Extended Support Mode, Encrypt Support Files, or the Support Resources web page.  Extended Support Mode creates detailed logs of actions performed to enable troubleshooting by IBM Support team. This can slow down AppScan, it should only be enabled when needed.
AppScan Log	Open the system log file.
Check for Updates	Send a request for new security assessment knowledge to be integrated into your version of AppScan.
Update Log	Open a log of live update installations.
About IBM Security AppScan	View general product information.

## Main toolbar

The icons on the toolbar offer quick access to frequently used features (that are also available from the menus).

Icon	Name	Click to:
	Scan >	(Available only if a scan is loaded and configured.) Opens a short Scan menu, with the following options:   Full Scan: Start a full scan (Explore and Test stages) or continue a paused scan.   Explore Only: Run an Explore stage only (or continue an Explore that was paused), without following it with the Test stage.   Test Only: Run a Test stage only (or continue a Test that was paused), without first running an Explore stage. Active only if there are already some Explore results.
	Pause	(Active only when a scan is running.) Pause current scan (whether Full Scan, Explore Only or Test Only).  You can resume the scan later. You can also save a paused scan to continue at another time.
	Manual Explore	Open the browser to the application's URL and manually browse the site, completing required parameters as you go. AppScan will then add this Explore data to its own, automatically collected Explore data, when creating tests for the site. For details see "Using AppScan" on page 115.
	Configuration	Open the Scan Configuration dialog box to configure the scan. For details see "Scan configuration dialog box" on page 42.
	Report	Create a report with the current scan data. For details see "Report overview" on page 197.
	Find	Search the results. For details see "Searching Results" on page 275
	Scan Log	Display the Scan Log during or after a scan. (Lists all actions performed by AppScan during the scan, as they occur.)
	PowerTools	Open one of the PowerTools, applications supplied with AppScan to help you with various tasks. For details, see "PowerTools" on page 236.

## View Selector

The three icons on the right of the toolbar toggle between the three views: Application Data, Security Issues, and Remediation Tasks.

Icon	Name	Click to display:
	Data view	Application Data view. For details see Chapter 7, "Results: Application Data," on page 167

Icon	Name	Click to display:
	Issues view	Security Issues view. For details see Chapter 8, "Results: Security Issues," on page 173
	Tasks view	Remediation Tasks view. For details see Chapter 9, "Results: Remediation Tasks," on page 193

## Browser toolbar

The icons on the toolbar of the embedded AppScan browser, used to display and save screenshots of application responses.

Button	Description
	Back
	Next
	Stop
	Refresh
	Go to the default starting URL.
	Make the current page the starting URL.
	Capture this page (available only when the browser is opened from the Screenshot tab in the "Detail Pane" on page 23).

## Keyboard shortcuts

AppScan utilizes the following keyboard shortcuts:

Shortcut	Function
F1	Open AppScan Online Help
F2	Open Data view
F3	Open Issues view
F4	Open Tasks view
F5	Start a full scan
[Shift] + F5	Pause scan
F10	Open the Configuration dialog box
[Ctrl] + N	Create a new scan
[Ctrl] + O	Open an existing scan
[Ctrl] + S	Save current scan
[Ctrl] + P	Print the current contents of the Application tree and Result list. (All nodes that can be seen by scrolling up or down the Application tree and Result list are included, but the nodes that are closed on the screen will appear closed in the printout.)
[Ctrl] + W	Open the welcome screen

## Accessibility controls

Describes all keyboard shortcuts and controls.

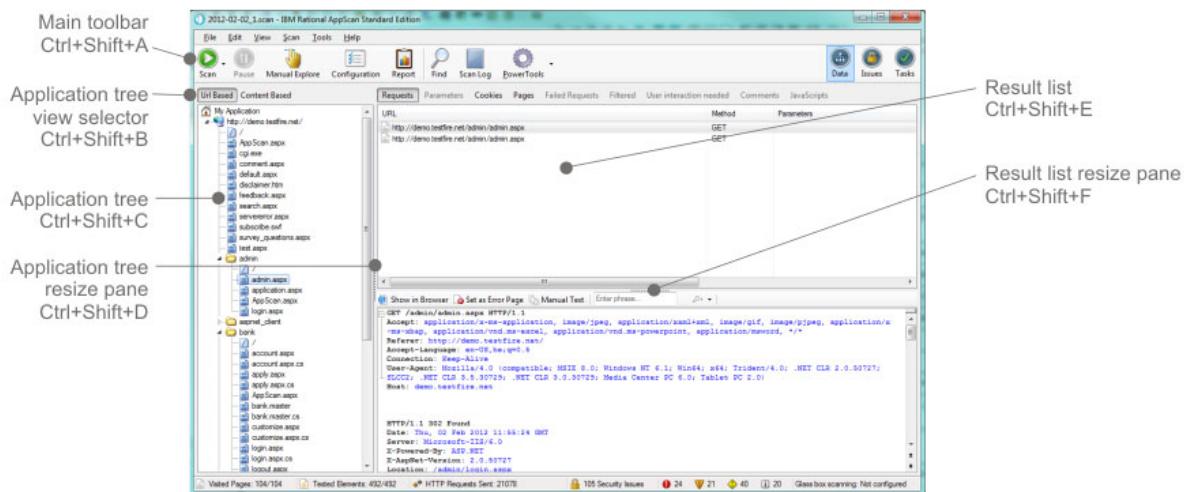
Full keyboard accessibility is activated by clicking **Help > Accessibility controls** so that it becomes checked. The Accessibility menu then appears to the left of the File menu, and additional keyboard control (described below) is enabled.

### Accessibility menu

The Accessibility menu appears only when **Help > Accessibility controls** is checked, but the shortcuts listed in it work even when it is not checked. The first three sections control the user interface views. The remaining sections are keyboard shortcuts from other menus.

#### All views

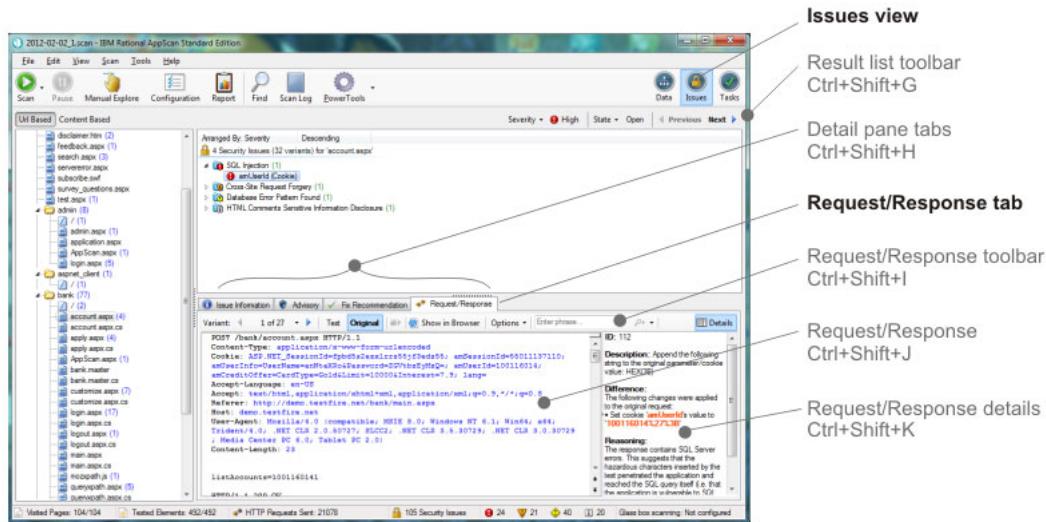
The following shortcuts apply to all views.



Command	Shortcut	Description
Main toolbar	Ctrl+Shift+A	Moves focus to the main toolbar. Use left and right arrows to select a toolbar icon. <b>Note:</b> The focus includes the three View Selector icons at the right end of the toolbar as well.
Application tree view selector	Ctrl+Shift+B	Moves focus to the Application tree view selector. Use left and right arrows to toggle between URL-based and Content-based views.
Application tree	Ctrl+Shift+C	Moves focus to the Application tree. Use up and down arrows to navigate. Use left and right arrows to open and close nodes.
Application tree resize	Ctrl+Shift+D	Moves focus to the border between the Application tree pane and the Result list. Use the arrow keys to adjust the panes.
Result list	Ctrl+Shift+E	Moves focus to the Result list. Use up and down arrows to navigate. Use left and right arrows to open and close nodes.
Result list resize pane	Ctrl+Shift+F	Moves focus to the border between the Result list and the Detail pane. Use the arrow keys to adjust the panes.

## Issues view

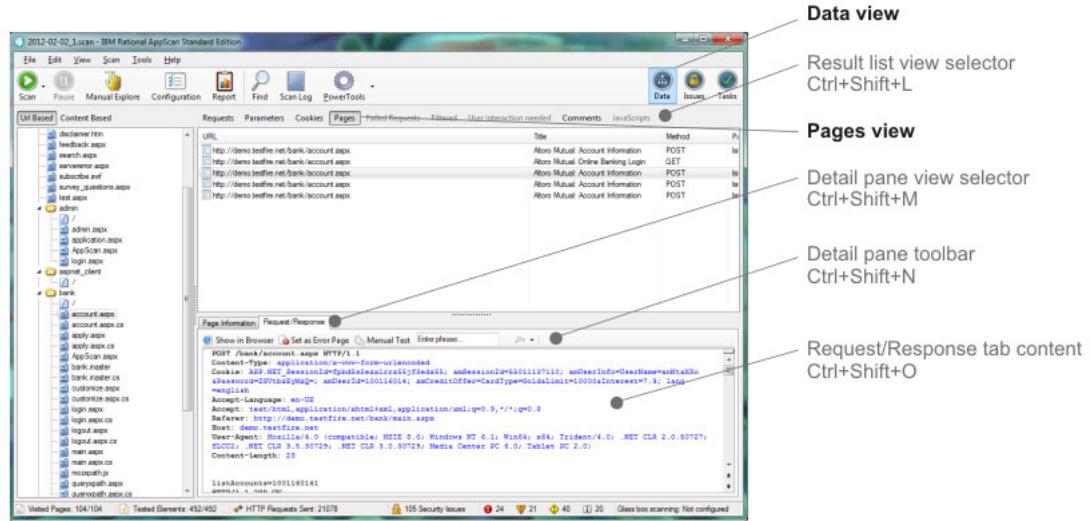
The following shortcuts apply to Issues view only.



Command	Shortcut	Description
Result list toolbar	Ctrl+Shift+G	In Issues view: Moves focus to the Result list. Use up and down arrows to navigate. Use left and right arrows to open and close nodes.
Detail pane	Ctrl+Shift+H	Moves focus to the Detail pane tabs.
Detail pane > Request/ Response toolbar	Ctrl+Shift+I	In Issues view: Opens the Request/Response tab toolbar in the Detail pane, and moves focus to its toolbar.
Detail pane > Request/ Response	Ctrl+Shift+J	In Issues view: Opens and moves focus to the Request/Response tab of the Detail pane.
Detail pane > Request/ Response details	Ctrl+Shift+K	In Issues view: Opens and moves focus to the Detail section (right-hand pane) of the Request/Response tab in the Detail pane.

## Data view

The following shortcuts apply to Data view only.



Command	Shortcut	Description
Result list view selector	Ctrl+Shift+L	In Data view: Moves focus to the Result list toolbar. Use left and right arrows to select Requests, Parameters, Cookies and so on.
Detail pane view selector	Ctrl+Shift+M	In Data view > Pages: Lets you select Page Information or Request/Response. Use left and right arrows to toggle.
Detail pane > Request/Response toolbar	Ctrl+Shift+N	In Data view > Pages: Moves focus to the Request/Response toolbar. Use left and right arrows to navigate.
Detail pane > Request/Response tab	Ctrl+Shift+O	In Data view > Pages: Moves focus to the content of the Request/Response tab.

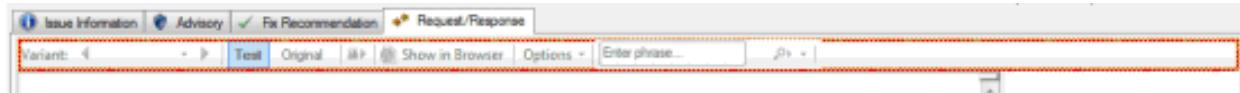
## Other keyboard shortcuts

The remaining shortcuts also exist in other menus, and are listed in “Keyboard shortcuts” on page 332

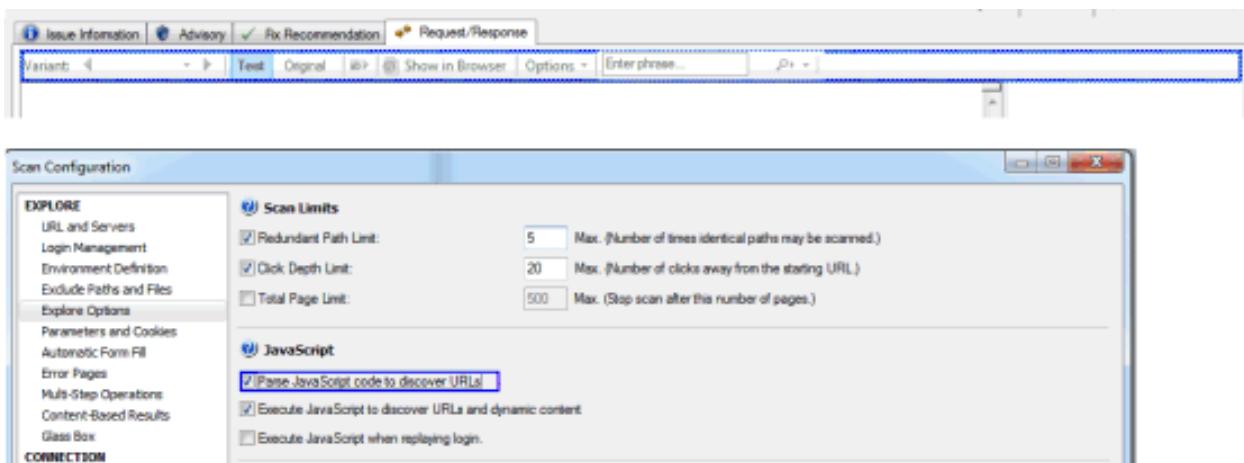
## Additional keyboard control

When **Help > Accessibility controls** is enabled:

- When you use an Accessibility shortcut to move the focus to a different part of the user interface, the new focus is outlined briefly in red.



- At any time you can see where the current focus is, both in the main user interface and in the Configuration dialog box, by clicking F7. The current focus will be outlined briefly in blue.



---

## Chapter 17. Glossary

Explains terms and acronyms used in the AppScan user interface and documentation.

**access control**

In computer security, the process of ensuring that users can access only those resources of a computer system for which they are authorized.

**action-based login**

This type of login replay reproduces the actions performed when you recorded the login sequence, and is usually the preferred login method.

**action-based login player**

A browser with two panes in which action-based login is replayed for verification and troubleshooting. The left pane shows the list of actions and highlights which is currently being performed; the right pane shows the result of the current action.

**advisory**

A document that contains information and analysis about a threat or vulnerability.

**application lifecycle**

The succession of stages a product goes through, from development to production.

**application server**

A server program in a distributed network that provides the execution environment for an application program.

**application test**

A type of test which focuses on application logic and issues resulting from insecure software development.

**application tree**

A tree-view display of a web application's structure, including directories and files.

**attack** Any attempt by an unauthorized person to compromise the operation of a software program or networked system. See also attacker.**attacker**

A user (human or computer program), that attempts to cause harm to an information system or to access information not intended for general access. See also hacker, attack.

**authentication**

The process of validating the identity of a user or server.

**Authentication Tester**

A brute-force-like testing utility. One of the PowerTools. It detects weak username-password combinations that could be used to gain access to a user's web application.

**authorization**

The right granted to a user to communicate with or make use of a computer system.

**back end**

The set of support components of a computer system, such as the database management system.

**black box**

When the output of an application is examined without reference to its internal code, the application can be described as a "black box", and the testing as "black box testing", because it treats the application as a "black box" the contents of which cannot be seen. Compare with "glass box".

**broken link**

A link that returns an invalid response when selected.

**brute force**

An attack by a program that tries every possible credential to compromise the security of a system.

**buffer** A reserved segment of memory used to hold data while it is being processed.

**buffer overflow**

An exploitation technique that alters the flow of an application by overwriting parts of memory. Buffer overflows are a common cause of malfunctioning software.

**case-sensitive**

Pertaining to the ability to distinguish between uppercase and lowercase letters.

**CGI** See Common Gateway Interface.

**character encoding**

A character set consisting of a code that pairs a sequence of characters from a given set with something else, such as a sequence of natural numbers, octets or electrical pulses. Encoding facilitates the storage and transmission of text through telecommunication networks.

**child node**

A node within the scope of another node.

**client** The user's workstation that is connected to a network. See also host.

**client-side**

Pertaining to an operation that is performed on the client application and not on the server.

**code injection**

A technique that introduces new code into an application. Code injection can be used by an attacker to introduce code into a computer program to change the course of execution.

**Common Gateway Interface (CGI)**

An Internet standard for defining scripts that pass information from a web server to an application program, through an HTTP request, and vice versa.

**communication timeout**

The intentional ending of an incomplete task after waiting a specified amount of time.

**concurrent login**

A login that occurs simultaneously with other logins.

**condition pattern**

In regular expressions, a pattern that the regular expression defines. The regular expression can be used to find items that match the pattern.

**cookie** Information that a server stores on a client machine and accesses during subsequent sessions. Cookies allow servers to retrieve specific information about clients.

**crawl** To search for information across various web pages on the Internet or on an intranet.

**cross-site scripting (XSS)**

An attack technique that forces a website to echo client-supplied data, which execute in a user's web browser.

**custom error page**

A feature of most web server software that allows the user to replace default error messages with messages that are custom designed for the application.

**CVE** Common Vulnerabilities and Exposures. An industry standard list that provides common names for publicly known information security vulnerabilities and exposures.

**CVSS** Common Vulnerability Scoring System. An open framework for scoring the risk associated with vulnerabilities.

**CWE** Common Weakness Enumeration. An industry standard list that provides common names for publicly known software weaknesses.

**database management system (DBMS)**

A software system that controls the creation, organization, and modification of a database and the access to the data that is stored within it.

**database service**

A service that provides the storage and retrieval of data in a database.

**DBMS**

See database management system.

**debug command**

A feature or command that assists in identifying programming errors during the software development process.

**delta** A difference, or an incremental value, between two instances.

**denial-of-service attack (DoS)**

In computer security, an assault on a network that brings down one or more hosts on a network such that the host is unable to perform its functions properly. Network service is interrupted for some period.

**depth** The number of clicks required for a user, or an automatic crawler, to get from a source page to a target page.

**directory indexing**

A web server feature that exposes contents of a directory when no index page is present.

**directory traversal**

A technique used to exploit websites by accessing files and commands beyond the document root directory.

**domain**

A subnetwork of clients and servers under the control of one security database.

**DoS** See denial-of-service attack.

**dump file**

The contents of memory without any report formatting.

**embedded browser**

The web browser that is embedded in AppScan and opens with a special toolbar for working with scans.

**encoding attack**

An exploitation technique that aids an attack by changing the format of user-supplied data to bypass sanity checking filters.

**encryption**

The process of transforming data into an unintelligible form in such a way that the original data either cannot be obtained or can be obtained only by using a decryption process.

**exclusion**

A parameter or process whose values are excepted during tests.

**executable**

A program file that is ready to run in a particular environment.

**Explore setting**

A setting that configures the parameters governing how an application will be explored by AppScan.

**Explore stage**

The stage of an AppScan scan during which the logic and objects of an application are identified, prior to testing.

**export** To save a copy of the current document, database or image into the file format required by a different application.

**extended support mode**

A mode that allows the user to record usage options and behavior, and to save the data in a file to send to technical support.

**false positive**

A test result classed as positive (indicating that the site is vulnerable to attack), that the user decides is in fact negative (not a vulnerability).

**fix recommendation**

The specific and technical details on fixing a web application to secure it against the issue that was discovered.

**Flash** A programming technique that enables movies and animation to display seamlessly in a web browser.

**form property**

A value used when forms are filled out automatically.

**full path name**

The name of any directory or file expressed as a string of directories and files beginning with the root directory.

**glass box**

Unlike "black box testing", which treats the application as a "black box" the contents of which cannot be seen, "glass box" testing "looks inside the box" by examining application's code. This means that glass box tests may pinpoint security vulnerabilities that black box tests could not.

**graphical user interface (GUI)**

A type of computer interface that presents a visual metaphor of a real-world scene, often of a desktop, by combining high-resolution graphics, pointing devices, menu bars and other menus, overlapping windows, icons and the object-action relationship.

**GUI** See graphical user interface.

**hard-coding**

The software development practice of embedding output or configuration data directly into the source code of a program or other executable object.

**hazardous character**

A character which is used for performing web application attacks, such as XSS or SQL injection.

**hidden parameter**

An HTML form parameter that is not rendered in the web page.

**host** A computer that is connected to a network and that provides an access point to that network. The host can be a client, a server, or both a client and server simultaneously. See also client

**HTML form element**

An element that allows the user to enter information, such as text fields, text area fields, drop-down menus, radio buttons, or check boxes, in a form.

**HTTP request**

A request sent to the site either during the Explore or Test stage of the scan.

**HTTP response**

A response sent by the server.

**ID** See identifier.**identifier (ID)**

One or more characters used to identify or name a data element and possibly to indicate certain properties of that data element.

**import**

To read a file in a format that is not native to the application in use.

**Industry Standards report**

A report of issues found on the user's web application and relevant information according to a selected industry standard. AppScan Industry Standard reports include SANS Top 20, OWASP Top 10, and WASC Threat Classification.

**in-session detection**

The detection of the in-session pattern in the responses AppScan receives, to verify that it is still logged in.

**in-session pattern**

A pattern identified in the login page, such as a logout link, that AppScan can use to verify that it is still logged in.

**insufficient anti-automation**

The result when a website permits an attacker to automate a process that should only be performed manually.

**interactive URL**

A URL that includes forms to be filled out manually by the user.

**invasive test**

An optional test which, if run on the application, may cause a denial-of-service situation.

**issue** A security risk to which a web application is vulnerable, or possibly sensitive information that is visible to unauthorized users.**Java applet**

An applet that is written in Java, and that can run in a web browser using a Java virtual machine (JVM).

**Java virtual machine (JVM)**

A software implementation of a processor that runs compiled Java code (applets and applications).

**link extraction**

The parsing or executing of code for discovery and collection of links from a web application.

**login sequence**

The sequence of user inputs that enables AppScan to log into your web application to scan it. It is recommended to record the login manually. AppScan then replays this sequence whenever it needs to log in during the scan. When you record a login sequence, AppScan analyzes both the actions and the requests. When replaying the login it attempts (by default) to reproduce the action-based login; if this is unsuccessful it will revert to the request-based login.

**malware**

Malicious software or executable code, often downloaded or received in the form of an innocent-looking file.

**manipulation**

A modification by an attacker of a data element, group of elements, action, or group of actions based on one or more properties. For example, modification of input by removing a required argument, or performing steps out of order.

**manual explore**

The process of manually crawling a web application to access and test parts of the site that are dependent on input from a real user.

**metacharacter**

ASCII character with special meaning during pattern processing. Such characters are used to represent single-byte or multibyte character patterns that can be matched during processing.

**multiphase scan**

A scan that consists of two or more phases.

**multi-step operation**

A sequence of two or more requests that must be sent in a specific order to access certain parts of the application. (Example: Add item to shopping cart > Enter payment details > Receive order confirmation.) Recording such multi-step operations as part of the scan configuration ensures that these parts of the site are scanned.

**network service**

A service that transmits data or provides conversion of data in a network.

**NTLM**

See Windows NT LAN Manager.

**numeric overflow**

The result from an arithmetic calculation that exceeds the space designated to hold it.

**parent node**

The node that contains the current node.

**parse** To break down a string of information, such as a command or file, into its constituent parts.

**path** The part of the URL which points to the location of an Internet resource.

**path filtering**

The process of filtering out or including pages according to set criteria.

**path traversal**

An attack technique that alters a document or resource location requested in a URL and forces access to files, directories, and commands that reside outside the web document root directory.

**pattern**

A method of describing text to be identified, using one or more regular expressions.

**PCI** See Peripheral Component Interconnect.

**penetration test**

A method of evaluating the security of a web application by simulating an attack by a hacker.

**Peripheral Component Interconnect (PCI)**

A local bus that provides a high-speed data path between the processor and attached devices.

**permission**

Authorization to perform activities, such as reading and writing local files, creating network connections, and loading native code.

**personal identification number (PIN)**

In Cryptographic Support, a unique number assigned by an organization to an individual and used as proof of identity. PINs are commonly assigned by financial institutions to their customers.

**phase** A process that includes the Explore stage followed by the Test stage of a scan.

**phase limit**

The maximum number of phases allowed in a scan. The limit is configurable.

**PIN** Personal Identification Number.

**platform**

The combination of an operating system and hardware that makes up the operating environment in which a program runs.

**port** An end point for communication between applications, generally referring to a logical connection. A port provides queues for sending and receiving data. Each port has a port number for identification.

**port listener**

A mechanism that allows the product to validate certain tests by listening to out-of-bound connections.

**Predictable Resource Location**

An attack technique used to uncover hidden website content and functionality. The attack searches for content in standard locations that is not intended for public viewing, such as temporary files, backup files, configuration files or sample files.

**privilege escalation**

The process of referring to scans that were run using different user privileges, in order to test whether privileged resources are accessible to users with insufficient access permissions.

**prompt**

A message or a displayed symbol that requests information or user action. The user must respond to allow the program to proceed.

**proxy** An application gateway from one network to another for a specific network application such as Telnet or FTP, for example, where a firewall's proxy Telnet server performs authentication of the user and then lets the traffic flow through the proxy as if it were not there. Function is performed in the firewall and not in the client workstation, causing more load in the firewall.

**redundant path limit**

The maximum number of times identical paths may be scanned in a scan, in order to reduce scan time and eliminate duplicate results.

**regular expression**

A set of characters, meta characters, and operators that define a string or group of strings in a search pattern.

**regulatory compliance report**

A report of issues found on a web application that do not comply with a selected regulation or legal standard. The regulations include legal Acts, Bills, and Laws of Canada, EU, Japan, UK, USA, and regulations of MasterCard and Visa. Custom regulatory compliance report templates can also be created.

**relative path**

A path that begins with the current working directory.

**remediation**

A suggestion for how to fix an issue.

**request-based login**

This type of login replay reproduces the requests that were sent when you recorded the login sequence.

**restriction**

A type of filter that limits a scan to listed URLs only.

**Result Expert**

An optional function that can be run after scanning to add CVSS settings, screen captures, and other information to the Issue Information tab of scan results.

**reverse engineer**

To analyze a device or system in order to learn details of its design, construction, and operation.

**risk analysis**

An analysis of the security issues found in a web application.

**risk assessment**

An evaluation of the benefits and consequences of an action or scenario.

**risk management**

The optimal allocation of resources to arrive at a cost-effective investment in defensive measures within an organization.

**role** A set of permissions.

**sanitize**

In web application security, to clean user input from harmful or hazardous characters, before using it.

**scan** The process of AppScan exploring and testing an application and providing the results.

**scan configuration**

A collection of AppScan settings that define the user's application/service, environment, and chosen scan methods.

**Scan Expert**

An optional function that explores application and network behavior, and recommends configuration changes to optimize scanning.

**Scan Expert analysis module**

A single check done by Scan Expert during its analysis.

**Scan Expert evaluation**

Scan Expert's evaluation of the user's configuration.

**scan template**

A scan configuration that can be loaded to use for a scan.

**scheduler**

A multithread, multiprocess background server designed to handle the scheduling and launching of jobs, based on a simple timing scheme.

**security audit**

A manual or systematic measurable technical assessment of a system or application.

**security risk**

The potential success of a threat and the damage that could ensue.

**sequence**

A list of recorded URLs.

**session**

A logical or virtual connection between two stations, software programs, or devices on a network that allows the two elements to communicate and exchange data. See also transaction

**session credential**

A string of data provided by the web server, stored within a cookie or URL, which identifies a user and authorizes that user to perform various actions.

**session fixation**

An attack technique that allows an attacker to fixate a user's session identifier and assume their online identity.

**session hi-jacking**

The compromise of a user's session by an attacker. The attacker could reuse this stolen session to masquerade as the user.

**session ID**

See session identifier

**session identifier (session ID)**

The compromise of a user's session by an attacker. The attacker could reuse this stolen session to masquerade as the user.

**session token**

An identifier that is sent by the browser as a parameter or a cookie, in order to correlate between a user and their current session on the web application. See also session identifier, transient token.

**severity rating**

The level assigned to an issue by the scan, indicating the security risk it represents.

**shell** A software interface between users and an operating system. Shells generally fall into one of two categories: a command line shell, which provides a command line interface to the operating system; and a graphical shell, which provides a graphical user interface (GUI).

**source code**

A computer program in a format that is readable by people. Source code is converted into binary code that can be used by a computer.

**spoofing**

The technique of faking the sending address of a transmission in order to gain illegal entry into a secure system.

**SQL** See Structured Query Language.

**SQL injection**

See Structured Query Language injection.

**stage** Part of a scan phase in which AppScan either explores or tests the site.

**State Inducer**

A tool that uses recorded URL sequences to induce specific server-side states needed for state-dependent tests.

**stateless protocol**

A protocol that does not maintain a relationship between commands. HTTP is an example of a stateless protocol.

**Structured Query Language (SQL)**

A standardized language for defining and manipulating data in a relational database.

**Structured Query Language injection (SQL injection)**

An attack technique used to exploit websites by altering back-end SQL statements through manipulating application input.

**syntax** The rules for the construction of a command or statement.

**test fix**

A temporary fix that is supplied to specific customers for testing in response to a reported problem.

**test policy**

A policy that limits the scan to certain categories and types of tests.

**Test request**

A request sent to the application during the Test stage of the scan. Test requests are designed to reveal security vulnerabilities.

**Test stage**

The stage of the scan during which the objects and logic of the scanned application are submitted

to a comprehensive barrage of typical, erroneous, and simulated-malicious usage techniques, resulting in a complete inventory of security vulnerabilities.

**thread** A stream of computer instructions that is in control of a process. In some operating systems, a thread is the smallest unit of operation in a process. Several threads can run concurrently, performing different jobs.

**threat** A security issue, or a harmful act, such as the deployment of a virus or illegal network penetration.

**threat class**

A group of security issues, classed together by WASC-TC categories. For each threat class, there are numerous specific tests; and for each test, numerous variants.

**transaction**

A request (to an application) and the response (from the application) that it generated.

**transient token**

A token whose value changes (usually a session token). Sending an expired transient token could result in AppScan getting logged out of the application it is testing, so it must keep them up to date. See also session token.

**Uniform Resource Locator (URL)**

The unique address of an information resource that is accessible in a network such as the Internet. The URL includes the abbreviated name of the protocol used to access the information resource and the information used by the protocol to locate the information resource.

**UNIX** A highly portable operating system that features multiprogramming in a multiuser environment. The UNIX operating system was originally developed for use on minicomputers, but was adapted for mainframes and microcomputers. The AIX operating system is IBM's implementation of the UNIX operating system.

**URL** See Uniform Resource Locator.

**user-defined test**

A test that is created by a user in addition to the tests that are automatically created and run.

**validation**

The process of verifying whether a certain test succeeded or failed to achieve its goal.

**vulnerability**

A security exposure in an operating system, system software, or application software component.

**web application**

An application that is accessible by a web browser and that provides some function beyond static display of information, for instance by allowing the user to query a database. Common components of a web application include HTML pages, JSP pages, and servlets.

**web browser**

A client program that initiates requests to a web server and displays the information that the server returns.

**web content**

Files and other resources that compose a website. web content may consist of image files, audio files, HTML files, JSP files, style sheets, database entries, or anything you can see on a website.

**web security**

The theory and practice of information security relating to the World Wide web, HTTP and web application software.

**web server**

A software program that is capable of servicing Hypertext Transfer Protocol (HTTP) requests.

**web service**

An application that performs specific tasks and is accessible through open protocols such as HTTP and SOAP.

**web Services Description Language (WSDL)**

An XML-based specification for describing networked services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information.

**Windows NT LAN Manager (NTLM)**

A protocol used in a variety of Microsoft network protocols for authentication purposes.

**WSDL**

See Web Services Description Language.

**XSS** See cross-site scripting.



---

## **Chapter 18. Notices for IBM Security AppScan Standard V9.0.1 documentation**

© Copyright IBM Corporation 2000, 2014.

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Programming interfaces: Intended programming interfaces allow the customer to write programs to obtain the services of IBM Security AppScan Standard Edition.

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.*

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan, Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 Japan*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who want to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*Intellectual Property Dept. for Security Software  
IBM Corporation  
5 Technology Park Drive  
Westford, MA 01886  
U.S.A.*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

## **Copyright license**

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2000, 2014.

If you are viewing this information in softcopy, the photographs and color illustrations may not appear.

## **Trademark acknowledgments**

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe is either a registered trademark or trademark of Adobe Systems Incorporated in the United States, and/or other countries.

Intel and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and JavaScript are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other product and service names might be trademarks of IBM or other companies.





**IBM**<sup>®</sup>

Printed in USA

SC27-6636-00

