TrafficShield™ Security Policy User Manual

version 3.1

MAN-0134-00

Service and Support Information

Product Version

This manual applies to product version 3.1 of TrafficShield™ Application Firewall.

Legal Notices

Copyright

F5 Networks, Inc. (F5) believes the information it furnishes to be accurate and reliable. However, F5 assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of F5 except as specifically described by applicable Control user licenses. F5 reserves the right to change specifications at any time without notice.

Trademarks

F5, F5 Networks, the F5 logo, BIG-IP, 3-DNS, iControl, GLOBAL-SITE, SEE-IT, EDGE-FX, FireGuard, Internet Control Architecture, IP Application Switch, iRules, OneConnect, Packet Velocity, SYN Check, Control Your World, ZoneRunner, uRoam, FirePass, and TrafficShield are registered trademarks or trademarks of F5 Networks, Inc. in the U.S. and certain other countries. All other trademarks mentioned in this document are the property of their respective owners. F5 Networks' trademarks may not be used in connection with any product or service except as permitted in writing by F5.

Export Regulation Notice

This product may include cryptographic software. Under the Export Administration Act, the United States government may consider it a criminal offense to export this product from the United States.

Export Warning

This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

FCC Compliance

This equipment generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules, which are designed to provide reasonable protection against such radio frequency interference.

Operation of this equipment in a residential area may cause interference, in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

Any modifications to this device, unless expressly approved by the manufacturer, can void the user's authority to operate this equipment under part 15 of the FCC rules.

Canadian Regulatory Compliance

This class A digital apparatus complies with Canadian I CES-003.

Standards Compliance

The product conforms to ANSI/UL 60950-1-2002 1st edition and Certified to CAN/CSA C22.2 No. 60950-1-3 first edition.

Table of Contents

6
Crawler

7
Learning - Testing & Fine Tuning the Policy

1

Introduction

- Product overview
- Document objectives
- How this manual is organized
- Audience and assumed knowledge
- Conventions
- Related documentation

Product overview

The F5® Networks TrafficShield™ Application Firewall is targeted at protecting mission-critical Web infrastructure against application layer attacks, and to monitor the protected web applications. These services complement the limited protection provided by firewalls, load balancers and other types of data and service protection devices, the TrafficShield security application analyzes traffic at network and application levels to handle a variety of threats, such as:

- Manipulation of cookies or hidden fields.
- Insertions of SQL commands or HTTP structures into user input fields in order to expose confidential information or to deface content.
- Malicious exploitations of the application memory buffer to stop services, to get shell access and to propagate worms.
- Unauthorized changes to server content via HTTP Delete and Put commands.
- Attempts aimed at causing the Web application to be unavailable or to respond slowly to legitimate users.
- Forceful browsing.

Document objectives

This manual explains how to set up a TrafficShield security policy and how to apply it to a Web application. The manual presents TrafficShield security application's security concepts, and shows how the concepts are implemented in the security policy context.

How this manual is organized

This manual consists of the following chapters:

Chapter 1 - Introduction: This chapter provides an overview of the F5® Networks TrafficShield™ Application Firewall product, describes the manual chapter organization and provides information about the color conventions used in the TrafficShield application, and about related documentation.

Chapter 2 - The Security Policy: This chapter explains how a TrafficShield security policy works, describes its components, and presents the Policy Browser, Crawler and Learning tools, that will help you to automatically collect the components.

Chapter 3 - TrafficShield Workflow: This chapter is your guide to the TrafficShield security policy workflow, it describes the steps to follow in order to create, adjust and maintain a security policy. Subsequent chapters explain each step in detail.

Chapter 4 - Accessing TSMS: This chapter explains how to access the TrafficShield Management Station (TSMS).

Chapter 5 - Policy Management Configuration: This chapter explains how to create and maintain policies, and describes the different components of the policies.

Chapter 6 - Crawler: This chapter guides you step-by-step through the procedure required to create an initial policy using the Crawler tool. This chapter also provides instructions on how to use the more advanced Crawler parameters.

Chapter 7 - Learning: Testing and Fine-Tuning the Policy: This chapter explains how to use the Learning tool to adapt the policy to real-life traffic requirements. It also covers the Policy editing feature, which allows you to view and manually adjust the entire security policy.

Chapter 8 - Monitoring: This chapter describes the tools that can be used by the network and policy administrators to monitor request traffic. It also explains how to use the TrafficShield security application monitoring tools to follow up potential attacks and workload.

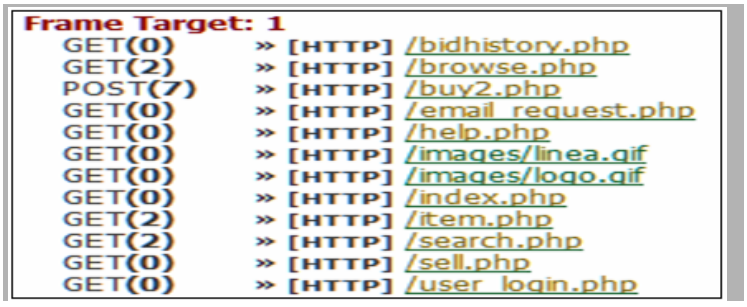
Glossary- The Glossary lists and defines relevant terms.

Audience and assumed knowledge

This manual is intended for the Web application security administrator or application owner. It assumes acquaintance with the nature of Web application attacks and a working knowledge of the Internet and of HTTP requests.

Conventions

Gold-colored lettered URLs point to referrer objects (see *Referrer* in the *Glossary* for definition). Green URLs belong to non-referrer objects.



Related documentation

The ***TrafficShield™ Installation and Configuration manual Version 3.1*** explains how to configure the deployed TrafficShield unit and its backup.

2

The Security Policy

- Concept
- How the policy works
- The security policy components
- The flow properties

Concept

The F5 TrafficShield Application Firewall uses positive security logic, in addition, complementary negative security logic is used in certain cases.

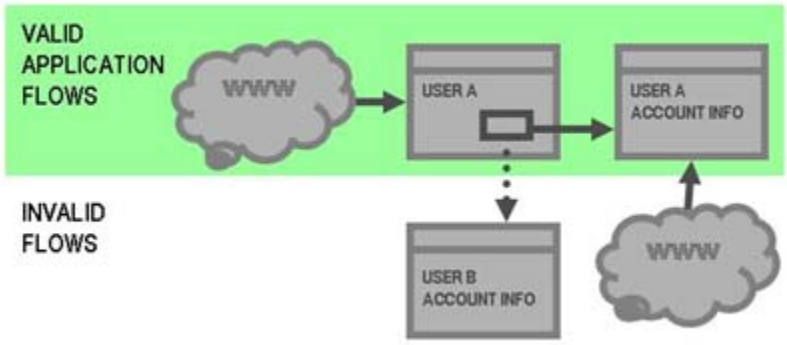
This means that all traffic is considered illegal unless it is specifically known to be legal.

The security policy is therefore a map of the application itself, containing all the application objects, flows, parameter, values and attributes that a user can make from any given point in the application.

The core of TrafficShield system's security functionality is the security policy. This policy determines which requests are valid and therefore can deny any request which does not match the policy's definitions. Depending on the work mode established, an invalid request can be blocked and reported, or only reported.

How the policy works

We call this map the "Application Flow Model." Think of it as a model of the entire application: every object, every parameter, and every value range for each parameter is part of the flow. By checking incoming traffic against the Application Flow Model, TrafficShield security application can screen out requests that do not follow the user behavior the application expects.



From every object in an application, a user may request access to a limited number of destinations. For example, when users log in to an online banking application, they are provided with several links to their respective accounts: savings, checking, and so on. They can click on each link to be directed to their personal account information and view it securely. This is the legitimate flow of the application, and this is the series of requests which are captured in the Application Flow Model.

Requests that are out of sequence or whose parameter values have been altered can be blocked once this security policy is in place. For instance, a user requesting an account information page, without first passing through login sequence, can be rejected, as this is not the correct order of the flow.

Likewise, a user who logs in and then tampers with the account links provided on a page (attempting to access other people's accounts) would be rejected since the parameter values have changed.

◆ Note

In each of these cases, the format and structure of the request are valid, according to the HTTP protocol. It is only within the specific context of the application that these requests can be considered malicious.

The security policy components

The main components of the security policy are described in this section.

Object types

The Object Types section lists the existing file types in the protected Web site. For example, a list of valid object types for a specific policy could be: **GIF JPG** and **HTML** only. If your policy contains the above list, then any request for a PDF file would be considered illegal.

Web objects

The Objects (files) section lists the existing objects in the protected Web site. For example, a list of valid objects could be: myPict.gif, myPict.jpg and myFile.html only. If your policy contains this list, then any request for yourFile.html would be considered illegal.

Application flows

The Application Flow (path) is the defined access path leading from one object to another object. For example, a list of valid flows would be:

- from abc.html to abc.gif, OK
- from abc.html to def.html, OK

If your policy contains this list, then any request that tries to access abc.gif from def.html would be considered illegal.

◆ Tip

Back flows are created automatically.

Flow parameters

The parameters used by the request. For example:

A list of valid parameters can be:
https://192.168.51.51:1043/dms/policy/pl_flows.php?m_id=0_4&uid=123

In this example we have a single parameter: m_id.

If your policy contains the above list, then any request that tries to read a variable with a different name from m_id would be considered illegal.

Please refer to the next section for more details.

Parameter value properties

The TrafficShield security application provides an option whereby you can define the allowed value format for each parameter of the request. For example, a list of valid parameters can be: https://192.168.51.51:1043/dms/policy/login?username=john&password=secret.

If your policy contains the above list, then calling this request with a value other than john would be considered illegal.

Character sets

A character set defines the allowed characters for the following request parts: Object, Parameter Name, HTTP header and User Input parameters per language.

If your policy contains a specific allowed character set that excludes the letter "Z" in the HTTP header part, then any request containing the letter "Z" in its header will be considered illegal.

Negative Regular Expressions

Negative regular expressions describe possible attacks.

For example: a regular expression that defines inserted scripts:

(?si)%3cscript\b

If your policy contains the above negative regular expression, then any request for a URL matching this list of directories will be considered illegal.

The policy build tools

The policy is an intelligent map of your Web application. It contains not only a list of the files included in the Web application but also other data such as the types of the files, the length of some crucial strings, allowed value ranges for parameters, and the relationships (links) between the files and the parameters passed from one file to another in a specific link.

You do not build this complex map yourself, which would be a tedious undertaking, especially if the Web application is updated frequently. TrafficShield security application provides the following tools for building this map:

- The Policy Browser collects important information about the site that the Crawler later uses while scanning the application. The user simply browses the application with it. The browser saves to a file the browsing information it encounters.
- The Crawler scans your application and builds a list of existing object types, objects, flows, parameters, and values, including objects generated by Java Script code. It can also use as input the file created by the Policy Browser.
- The Learning mechanism can analyze traffic from sources such as real live traffic, and the Crawler.

- The Policy Audit Tools feature allows you to see the entire policy built for the Web application. It is a visual representation of the application itself, which can be easily edited using common sense and application knowledge. Although a policy could, in theory, be built using just the Crawler and Learning, editing the policy is an effective way to ensure its accuracy.

The Crawler, Learning mechanism, and Policy Editing capabilities complement each other. The Crawler issues a preliminary map. Subsequently, the Learning tool shows you whether the Crawler's decisions are consistent with the requirements of real-life traffic, and allows you to further tune your policy until it is ready. For more details, please refer to Chapter 6, *Crawler* in this document.

What happens to illegal requests

When the TrafficShield security application diagnoses a request as illegal, it processes it according to what you have asked it to do: it warns you but lets the request through, or warns you and blocks the request.

◆ Note

Another possibility is that the TrafficShield security application will redirect to a customized blocking response.

By defining Ignored Items, you can set TrafficShield security application to also discard recurring illegal requests without posting a warning.

The flow properties

Target Object

In simple terms, this is the "to" side for a flow that runs "from" and "to" an object.

Referrer Object

This is the object from which the flow began its path to the Target Object.

Method

This is the action done on the Target Object. For example: GET, POST, PUT and Delete.

Target Frame

The Target Object will be loaded to this frame number.

Note

The TrafficShield user interface frames.

Has QS/PD

This flag indicates whether the HTTP/HTTPS request (for the requested object) has a query-string or a POST-data.

Check QS/PD

This flag indicates whether the TrafficShield security application should verify if the request QS/PD complies with the policy. If the flag is TRUE, it enforces the defined policy of the request's QS/PD; and if the is FALSE, it does not check the QS/PD.

Number of Parameters

Maximum number of parameters in the HTTP/HTTPS request.

Parameter List

This lists the parameters that can appear in the HTTP/HTTPS request.

3

TrafficShield Workflow

- Guidelines to workflow
- Preliminary stage
- Stage 1- Defining the web application
- Stage 2 - Creating a policy
- Stage 3 - Testing and fine-tuning the policy
- Stage 4 - Putting the policy into effect: blocking

Guidelines to workflow

This chapter is your guide to the TrafficShield security application workflow: it describes the steps to follow in creating, adjusting, and maintaining a security policy.

The following table provides a summary of the steps to follow, and the resources needed to implement them.

Stage	Resource Required	Time Required
Preliminary Stage: Installing and Configuring the TrafficShield unit	Network engineer	1-2 hours depending on the network infrastructure.
Stage 1: Defining the Web application	Network engineer	0.5-1 hour for small to medium Web applications and 3-4 hours for bigger and more complex Web applications.
Stage 2: Creating and modifying the initial policy.	Policy Builder: A person who has knowledge of the Web application.	2 hours to set up. Crawler may take several minutes to several hours to run the automatic process. (Allow 1 hour for all static pages, and several minutes for each dynamic script.)
Stage 3: Testing and fine-tuning the policy	Policy builder	1 hour a day for 1-2 weeks
Stage 4: Putting the policy into effect: Blocking	Policy builder	1-2 hours

Preliminary stage

This stage is done only once, the first time the TrafficShield security application unit is taken out of the box. This stage includes both the installation and the configuration of the unit.

The steps of the preliminary stage are described in the Installation and Configuration Manual.

Stage 1- Defining the web application

This stage includes:

Creating the Web application definition and defining the TrafficShield hardware units included in the Web application.

This step is described in the Installation and Configuration Manual. The remaining stages are described in this manual.

Stage 2 - Creating a policy

After defining the Web application it is necessary to populate a policy with the specific web application policy components.

This stage includes:

1. Defining a new policy
2. Running the Crawler

The Crawler automatically creates a preliminary security policy for the application. Typically, the Crawler maps most of the objects, flows, and parameter value ranges in a Web application, including those generated dynamically using Java Script and other client-side scripting means. This initial policy is never fully accurate, however. For instance, while the Crawler can determine parameter values for static parameters such as drop-down lists, it cannot always provide reasonable value ranges for user-input parameters. You can enter these finishing touches to the policy using the automated Learning mechanism and the Policy Management Configuration tools (stage 3).

Stage 3 - Testing and fine-tuning the policy

After creating the initial policy using the Crawler, you can expose the application to user traffic in a non-blocking, "what if" mode. This can be safe traffic, that is, traffic generated by users who are not potential attackers. This safe traffic is typically a small group of QA persons or the employees of your company. If the application is already active (i.e., a legacy application), you can apply the same procedure (again, in a non-blocking mode) and adjust the policy in order to maximize security and minimize the chance of false positives.

During the testing stage, TrafficShield security application captures the "illegal" requests and displays the appropriate information, such as URI lengths that exceed your expectations or attempts to access non-existing objects. Although you know what the values should be, and you may have entered them during your review, the real-life traffic may return unforeseen but legal user behavior and may lead you to further fine-tune the reviewed policy. This might involve adding missing objects to the policy, and adding parameters as well as parameter values. Through the real-life traffic, TrafficShield security application learns the real nature of legitimate requests and allows you to adapt the policy accordingly.

As real-life traffic is propagated through TrafficShield security application in none-blocking mode, the administrator can verify that:

- No false positive alarms have been posted.
- TrafficShield security application warns you in case real attacks are detected.

Stage 4 - Putting the policy into effect: blocking

You know that your policy is ready when all the alerts generated in the Learning tables represent invalid requests, such as one-off requests for invalid information or automated scripting attacks. The absence of false warnings ("false positives", that is, warnings on requests that are actually legal) means that your policy contains all the necessary objects and flows, and that all of the parameters are set to values that are characteristic of non-harmful, real-life traffic.

The next step is to activate TrafficShield security application's Blocking Mode. This can be done gradually, as the policy is more mature and tested. Through a set of simple checkboxes, you tell TrafficShield security application what to block. For example, by activating the "Illegal Object Type" blocking, TrafficShield security application will consider illegal any request referring to a file whose type is not included in the policy.

Any warnings that the Learning tool might return after you activate all of the desired blockings should be considered as potentially harmful behavior warnings. For more information about warnings generated after a first revision of the policy, please refer to Chapter 5 *Policy Management Configuration*.

4

Accessing TSMS

- Logging into the TSMS application

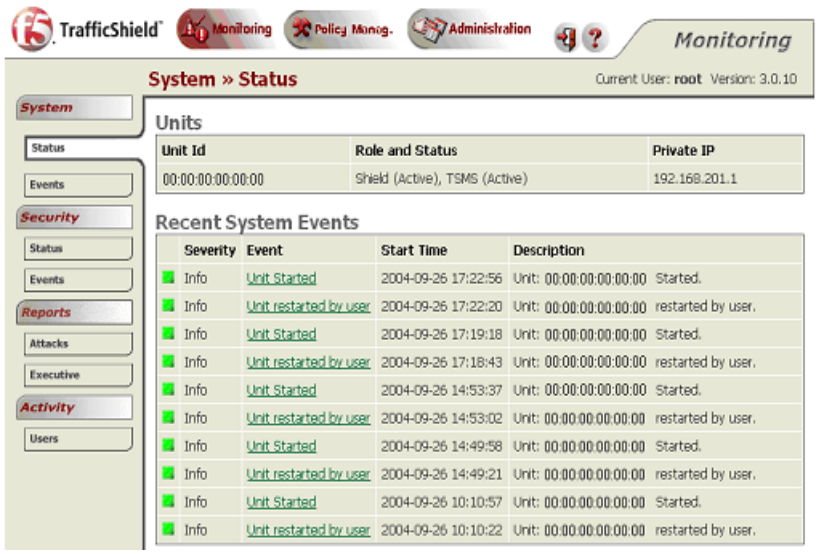
Logging into the TSMS application

This chapter explains how to access the TrafficShield Management Station (TSMS). The TrafficShield Management Station (TSMS) is a Web-based tool built in to the TrafficShield Application Firewall. You use the TSMS to run Configuration Administration operations.

1. On a PC from which the TrafficShield unit can be reached, use your Web browser to connect to the TrafficShield management portal. Point the browser to the TS Private or Permanent IP specified during the initial configuration script. Use custom SSL port 1043: https://ip.add.re.ss:1043
   A security alert message may appear.

2. Click **Yes** to continue.
   The logon page opens.

3. Enter the TrafficShield security application Web Administrator's user name and password that you defined earlier, and click the Login button.
   The TrafficShield system opens. It defaults to the Monitoring page



.

5

Policy Management Configuration

- Scope
- Add a new policy
- Policy properties
- Blocking Policy table
- Other policy activities

Scope

This chapter explains the procedure for creating a new policy. Note however, that the configuration process, explained in the installation manual, always creates a default policy. This means that by now you already have at least one policy defined in the TrafficShield Management Station (TSMS), either empty or populated. You can manually modify the default policy or re-run the Crawler in order to further update the policy.

◆ Tip

The Crawler also creates a default policy for the Web application.

◆ Tip

After any changes are made to the Policy, it is important to click the Set Active Policy button to re-activate the policy with the changes.

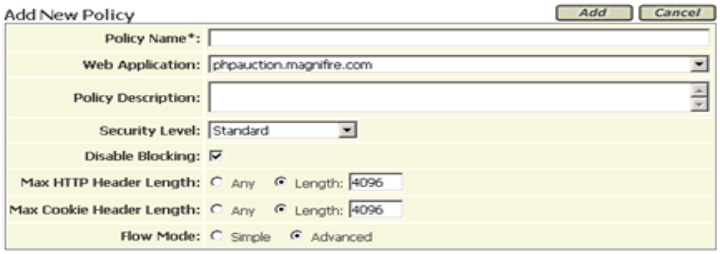
◆ Note

A policy record can be created only if at least one Web Application entry was created. For more details on how to define Web applications in the TrafficShield security application, please refer to ***Web Applications****,* ***TrafficShield™ Installation and Configuration Manual Version 3.1****.*

Add a new policy

1. Navigate to the **Policy Management** tab > **Policies List** tab.
   A list of existing policies appears. If you ran the TrafficShield configuration wizard, the first time you access this page you will see the policy you defined or selected via the wizard.

2. Click the **Add** button.
   The Add New Policy page opens.



3. Enter the information described below and click Save to save your information. This will automatically open the Policy Properties tab.

   Policy Name
   Enter a name for this policy. You can use any name.

   Web Application
   Specify the address (www…) of the Web application to which this policy will be applied.
   You can define different policies for the same Web application but only one policy can be active for a certain Web application at any given time.

   Policy Description
   Optionally, enter a few words that describe this policy.

   Security Level
   The default security level is Secure. Each level contains a different set of violation- driven actions.

   Tip: *You must save the policy before you can view the Custom security level and edit the violation driven actions. For more information, please refer to* ***Blocking Policy table***, on page 5-9.

Disable Blocking
See ***Blocking Policy table***, on page 5-9.

Max HTTP Header Length
The maximum length a request processed by this policy is allowed. '0' means unlimited length. Initially this field will be populated by the Crawler. The value can be changed manually by the user or automatically by the Learning process.
By choosing the Any button, any HTTP header length will be allowed.

Max Cookie Header Length
The maximum length a cookie processed by this policy is allowed. By choosing the Any button, any Cookie header length will be allowed.

Flow Mode
Two flow modes are available: Simple and Advanced. The Simple flow mode is the default mode.
The flow mode is applied in the Policy Properties screen.

- By selecting the Simple button in the Flow mode area, the user is instructing the TrafficShield system to create a simplified policy, where all objects are defined as entry points. This is true whether the user uses the Crawler to create the policy or decides to manually create a policy.
- By selecting the Advanced button in the Flow Mode area, the user instructs TrafficShield system to automatically create the policy.

◆ Tip

Always maintain the same Flow Mode option that was used to initially create a specific policy. We do not recommended that you switch back and forth between Simple and Advanced flow modes.

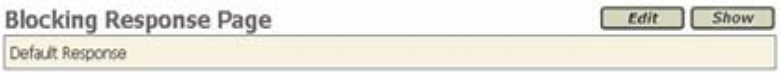
Policy properties

When you save a new policy record, the policy properties appear. You can access the properties of a policy also by clicking the Policy Properties tab, in the left Navigation Panel.

Editing the current policy's properties

Blocking Response Page

Responses returned by the Web server to requests can be verified against the negative regular expressions applied to the "Server response data". See ***Chapter 6 - File type associations***, on page ***17*** in this document. In cases where the response evaluates to the negative regular expression, a default response is returned, but you can replace it with a customized response.

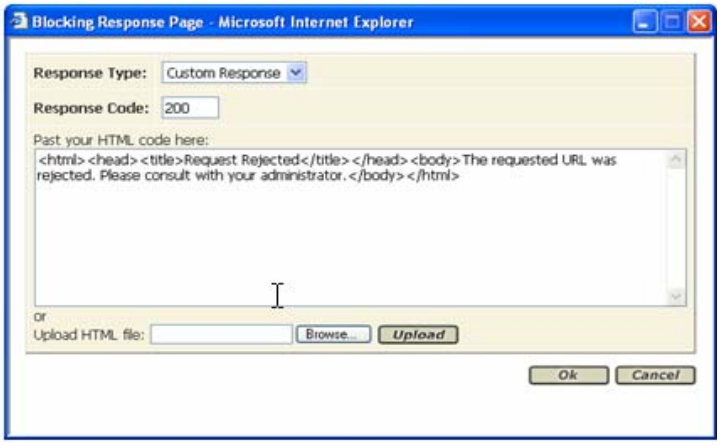


The response is an HTML page. You can build the page here or use a page stored elsewhere.

Click the Show button to display the current blocking response page in a popup window.

To edit a response page

1. In the Blocking Response Page section, click the Edit button. The Blocking Response Page opens.
2. Upon Completion, click the OK button to save your changes.



Response Type

This field defines the type of response page that will be displayed to the user. If you select the default response, you can see its HTML code but you cannot change it. The possible values that can be selected here are:

- Default Response - This is the default web page in the TrafficShield security application.
- Redirect URL - This means that instead of a web page, the TrafficShield security application returns to the user an HTTP redirect URL.
- Custom Response - This means that the user has defined that this is the page the TrafficShield security application will returned to the user.

Response Code

Do not change the Response Code.

Paste your HTML code here

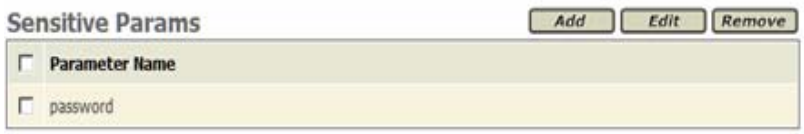
You can either paste or type the page's HTML code into the "Paste your HTML code here". Or upload a file in the next field.

Upload HTML file

Use the browser button to select the HTML file that will serve as the response page, and click the Upload button to load the file as the response page.

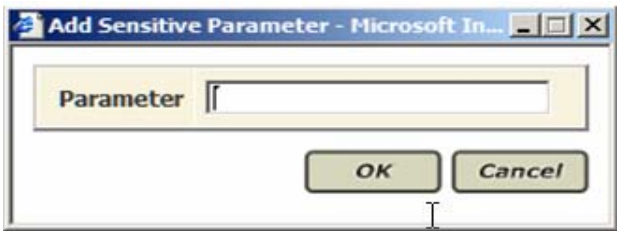
Sensitive parameters

All incoming requests, valid and invalid, are stored in TrafficShield security application in plain text format. Some requests may include user input, such as a password or a credit card number, that you may not want to store once the request has been processed (a string of asterisks will be stored instead of the actual value). You can avoid storing this sensitive data by entering the names of the input fields in the Sensitive Parameters sections.



To specify a sensitive parameter:

1. Click the **Add** button.
   The Add Sensitive Parameter box opens.



2. In Parameter, enter the name of a sensitive field.
   Enter the name of the input field exactly as defined in the request. For example:
   http://siterequest.com/bank.php?account=12345
   If you define the field account to be a sensitive parameter, it will be displayed in the following manner:
   /bank.php?account=XXXXX
3. Click OK.

Tip

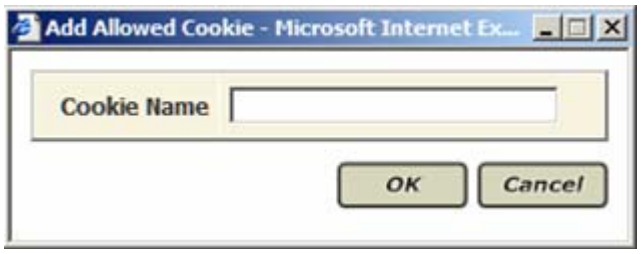
*Upon installation, a sensitive parameter called **password** is created by default.*

Allowed modified cookies

You can set the policy to ignore certain cookies included in the request even if they do not meet the expected criteria. This is done in the Allowed Modified Cookies section by simply listing their names.

To define an allowed cookie:

1. Click the **Add** button.
   The Add Allowed Cookie box opens.



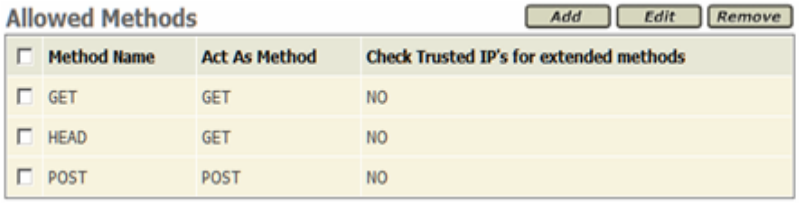
2. In **Cookie Name**, enter the name of an allowed cookie.
   Enter the name of a cookie exactly as it is expected to appear in the request.
3. Click **OK**.

Allowed methods

TrafficShield security application accepts certain methods upon installation. The default methods are listed in this section when you first access it. See example below.

TrafficShield security application considers as invalid all requests that use HTTP methods other than those listed in the Allowed Methods section.

You can set other HTTP methods valid by adding them to the list.



To allow an additional method:

1. Click the Add button.
2. The Add Allowed Method window opens.
3. Enter the new method's information and click OK to save and return to the Policy properties window.
   -Or-
   To exit the window without saving the information, click Cancel.

   Method Name
   Select the name of an allowed method.

   Act as Method
   Select the mode of operation allowed for the additional method.

   Check trusted IPs for extended methods
   Check the "Check trusted IPs for extended methods" checkbox to allow this additional method only if it appears in requests sent by one of the trusted IPs.
   Clearing this checkbox will make the method valid in all incoming requests. For details about trusted IP addresses, see the *Web Applications* chapter in the ***TrafficShield™ Installation and Configuration Manual Version 3.1***.

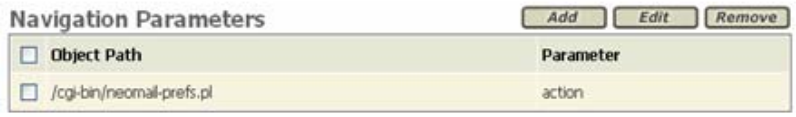
Navigation Parameters

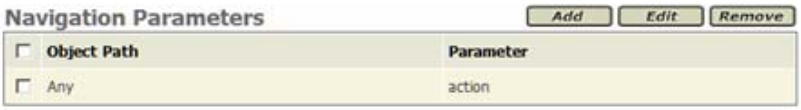
In some Web applications, pages can be dynamically built by server-side scripting. In such cases, pages are generated based on parameters passed to the Web server.

To allow TrafficShield security application to identify those otherwise "invisible" pages and to build the appropriate flows, you need to specify the exact names of the parameters passed to the server. The parameter names are specified in the Navigation Parameters section.

Note

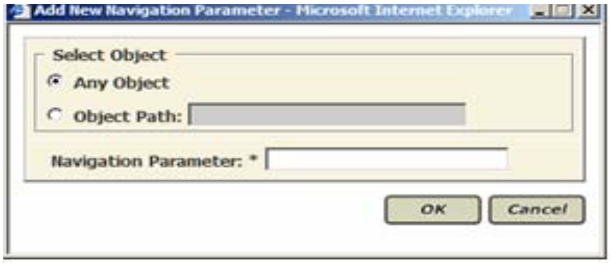
The two examples below demonstrate how the user can define a specific object path plus parameter, or if the policy contains a common parameter used by more than one object path, how the user will need to define a general Navigation path: Any and the common parameter name, as displayed below.





To specify a navigation parameter passed to the web server for dynamic page building:

1. Click the Add button.
   The Add New Navigation Parameter window opens. Enter the new navigation parameter's information and click OK to save.



2. In Select Object, select one of the following:

   Any Object
   If the Web application consists of just one physical page (the index page), select Any Object.

   Object Path
   If the Web application contains physical pages and dynamic page building starts from one of them, select Object Path and enter the URL of that object.

3. In Navigation Parameter, enter the name of the parameter passed to the Web server for page building purposes.

Blocking Policy table

This section describes in detail the Blocking Policy table.

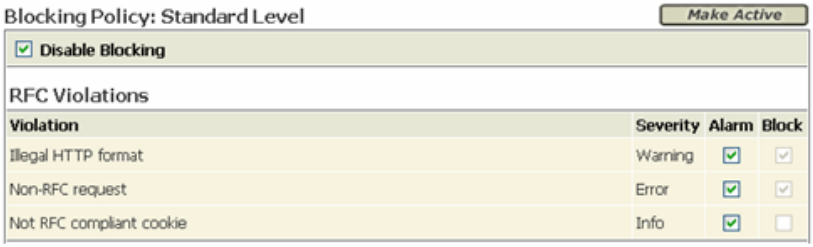
To navigate to this table, the user should choose the Policy management > Policy Properties tab.

This table is accessed when the user clicks the Edit button next to the Security Level field in the Policy Properties section.

Tip

In order to customize the security level, user may edit one of two default security levels. When the security level is saved it, the changed security level is called the "Custom" security level.

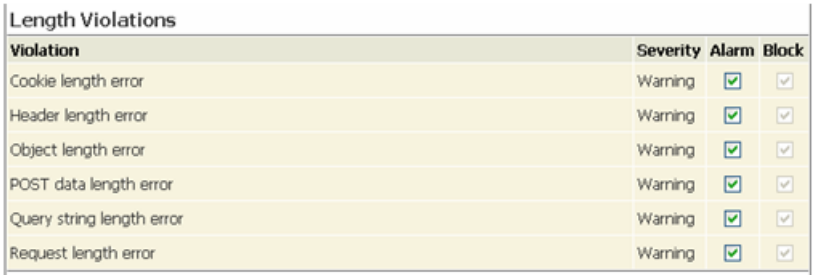
Each blocking category is described separately.



RFC violations

Filter	Description
Violation: Illegal HTTP format	Request line is illegal in the following cases: - Method, resource or HTTP version is missing. - HTTP version is not HTTP/1.0 or HTTP/1.1 - Host header is missing the method in the request. See "Methods" in the Policy section and "Trusted IPs". See ***Trusted IPs for Extended Methods***, on page ***7*** in this chapter.
Non RFC request	Binary Data in the user input contradicts user input type or method.
Not RFC compliant cookie	Cookie format does not follow RFC.
Illegal access to method by not allowed IP	Request was received from a Client IP that is not allowed to use the method in the request. See "Methods" in the Policy section and "Trusted IPs". See ***Trusted IPs for Extended Methods***, on page ***7*** in this chapter.
Illegal domain (Web Application)	Host header value doesn't match any of the Web application FQDNs or Aliases defined in the TSMS.
Illegal entry point	The requested resource is not an acceptable entry page to the Web Application.
Illegal flow to object	The transition from the previous resource to the requested one is illegal.
Illegal method	The method is not defined in the policy properties as an allowed method.
Illegal object type	Requested resource type (extension) is not defined in the policy.
Non existent object	Requested object is not listed in the policy. To better understand, please refer to ***Non-existent object***, on page 7-9 in this manual.

Length violations

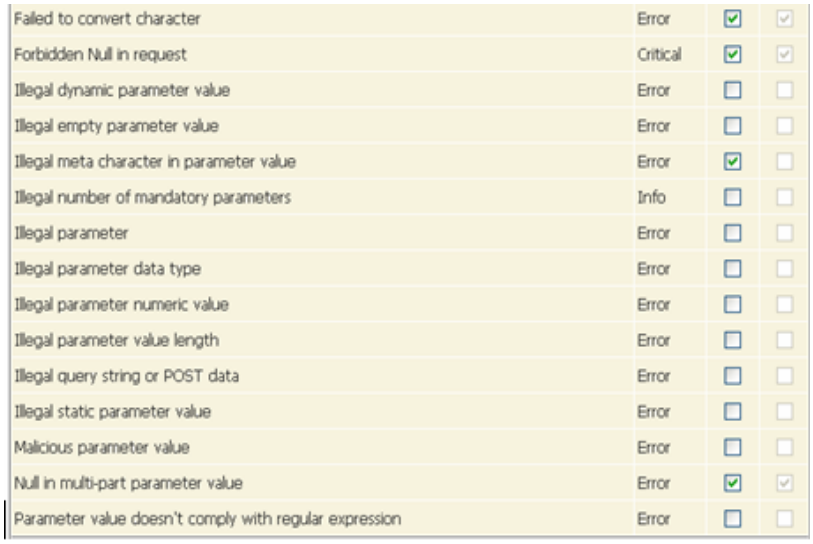


Length Violations

Violation	Severity	Alarm	Block
Cookie length error	Warning	☑	☑
Header length error	Warning	☑	☑
Object length error	Warning	☑	☑
POST data length error	Warning	☑	☑
Query string length error	Warning	☑	☑
Request length error	Warning	☑	☑

Filter	Description
Cookie length error	Cookie header value length exceeds the threshold set in the policy.
Header length error	Header name + value length exceeds the HTTP Header Length set in the Policy Properties.
Object length error	Resource name length exceeds the policy limit.
POST-data length error	Request method is POST and the user input data length exceeds the policy limit.
Query-string length error	Request method is GET and the user input data length exceeds the policy limit.
Request length error	Request length exceeds the maximum request length defined in the policy.

Input violations



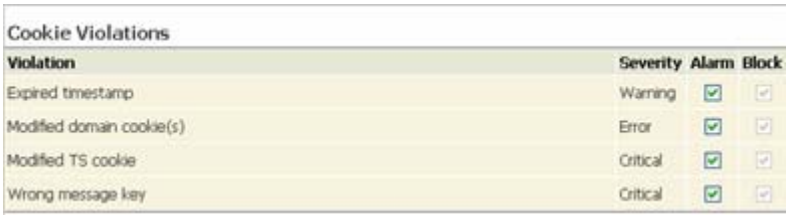
Failed to convert character	Error	☑	☑
Forbidden Null in request	Critical	☑	☑
Illegal dynamic parameter value	Error	☐	☐
Illegal empty parameter value	Error	☐	☐
Illegal meta character in parameter value	Error	☑	☐
Illegal number of mandatory parameters	Info	☐	☐
Illegal parameter	Error	☐	☐
Illegal parameter data type	Error	☐	☐
Illegal parameter numeric value	Error	☐	☐
Illegal parameter value length	Error	☐	☐
Illegal query string or POST data	Error	☐	☐
Illegal static parameter value	Error	☐	☐
Malicious parameter value	Error	☐	☐
Null in multi-part parameter value	Error	☑	☑
Parameter value doesn't comply with regular expression	Error	☐	☐

Filter	Description
Failed to convert character	Some characters in the object or user input cannot be mapped into the Latin-1 characters table.
Forbidden Null in request	Forbidden null byte in request.
Illegal dynamic parameter value	Parameter value doesn't match the dynamically generated pool of legal values.
Illegal empty parameter value	Empty is not allowed for the specific parameter value.
Illegal meta character in parameter value	The parameter value contains a character that is set to "N" (false) in the Administration > Character Sets > User Input: language
Illegal number of mandatory parameters	The number of mandatory parameters in the flow is different from the number of mandatory parameters defined in the policy.
Illegal parameter	Parameter is not defined in the flow.
Illegal parameter data type	Parameter value differs from the type assigned to the parameter in the policy.
Illegal parameter numeric value	Numeric (decimal or integer) parameter value exceeds the value range set for it in the policy.
Illegal parameter value length	Parameter value length exceeds the length limitation set for it in the policy.

Filter	Description
Illegal Query-String or POST-Data	Request contains user input not expected to be found in the flow.
Illegal static parameter value	Parameter value doesn't match any of the values in the Static pool of values for a given parameter.
Malicious parameter value	Parameter value matches one of the regular expressions describing common web attacks, i.e., XSS, SQL injection.
Null in multi-part parameter value	NULL character found in the parameter non-binary type in multi-parted POST-data.
Parameter value doesn't comply with regular expression	The Parameter value doesn't evaluate to the positive regular expression which defines the valid values for this parameter.

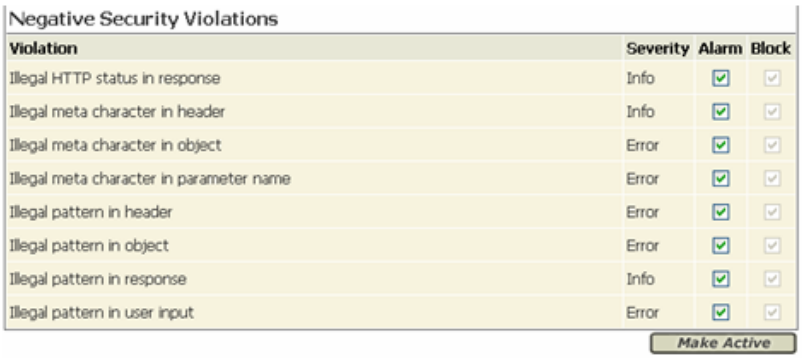
Cookie Violations

This Cookie Violations category is divided into four cookie violation sub-categories: Expired timestamp, Modified Domain Cookie(s), Modified TS Cookies, and Wrong message key. See table below for details.



Violation	Description
Expired timestamp	TrafficShield cookie was returned after the TTL expired.
Modified Domain cookie(s)	The modified domain cookies.
Modified TS cookie	Suspected tampering with the cookie served by TrafficShield system.
Wrong message key	Suspected TrafficShield cookie hijacking.

Negative security violations



Filter	Description
Illegal HTTP status in response	Server responded with HTTP status of type 4XX or 5XX. Statues 400,401,404,407,503 are not included in this rule.
Illegal meta character in header	The HTTP header value contains a character that is set to "N" (false) in the Administration > CharSets > HTTP Headers field.
Illegal meta character in object	The Object part of the URI contains a character that is set to "N" (false) in the Administration > Character Sets > Object Path field
Illegal meta character in parameter name	The parameter name contains a character that is set to "N" (false) in the Administration > Character Sets > Param Name.
Illegal pattern in header	One of the HTTP header values evaluates to at least one negative regular expression applied to the "Header value". See the Negative RegExp section in the TrafficShield Unit Installation/Configuration manual.
Illegal pattern in object	Evaluates to a negative regular expression applied to the Object part of the URI.
Illegal pattern in response	Data in the server response matches negative regular expression applied to "Response". Violation triggering is done by setting the "Check Response" flag of a specific object type to true.
Illegal pattern in user input	Evaluates to a negative regular expression applied to the "Key-value pairs". Test is done on user input for both POST and GET methods.

During the Learning stage, the alarms should diminish. At this point you can be confident that all missing objects have been added, and other attributes are attuned to real-life traffic requirements. The blocking mode should be activated only after monitoring traffic without any Learning alarms for several days.

The trigger for activating the Blocking mode is any point in time that the user can reasonably assume that the policy is accurate: meaning, all resources are present and all attribute values meet the requirements of legitimate real-life traffic and, therefore, any further alarm should be considered as suspicious.

After activating the blocking mechanism, illegal requests may continue to appear in the Learning pages: you can still accept their suggestions if they are justified, or you can alternatively clear them out.

Blocking by categories

Blocking is implemented by telling the TrafficShield security application what to consider as illegal.

An illegal request is a request whose content contradicts the policy settings. Therefore, most filtering attributes correspond to policy attributes that you are familiar with. For example, by filtering "Illegal file types" you instruct the TrafficShield security application to consider a request as invalid if it tries to access an object of a type not included in the policy.

You do not have to activate all of the available blockings.

To set blocking categories:

1. Access the Policy Management and select the relevant policy from the Policies List tab.
2. Press the Policy Properties tab on the left side menu or the Edit button above the policy list to open the Policy Properties window. The properties displayed belong to the currently chosen policy.
3. In Security Level, select one of the standard levels, or select Custom, if this security level already exists.

   The Standard level provides minimal blocking and the High Security level provides comprehensive blocking. The Alarm/Block set of flags of both levels may be edited and saved as a Custom security level.

   The rest of this procedure relates to the Custom option.
   If you want to disable blocking temporarily, check the Disable Blocking checkbox in the Policy Properties tab, clearing the box reactivates the selected blockings.

4. Go over each blocking category and define what the TrafficShield security application should do when an illegal request matches the category's definitions. The options are:

   Alarm
   Check the Alarm checkbox to instruct the TrafficShield security application to only post an alarm to the Security Events log and the Learning pages without blocking the Web application user.

   Block
   This option acts like "Alarm", but the request that triggered the violation is blocked.

   You can check both boxes. Some Block boxes are checked and grayed, meaning that requests that commit that specific violation are always blocked.

5. Click the Make Action button, and then the Set Active Policy button.

Using Learning in Blocking Mode

After you enable the blocking mechanism, the Learning system continues to analyze traffic. The requests that end up in the Learning tabs are those that contradict the policy. You can still accept some or all of them if they warrant policy changes, or clear them if they do not.

Other policy activities

There are several other activities that you may want to use with your policies. You have the option to:

- Edit a policy
- Remove a policy

Edit a policy

There are two ways to choose the existing policy you would like to edit:

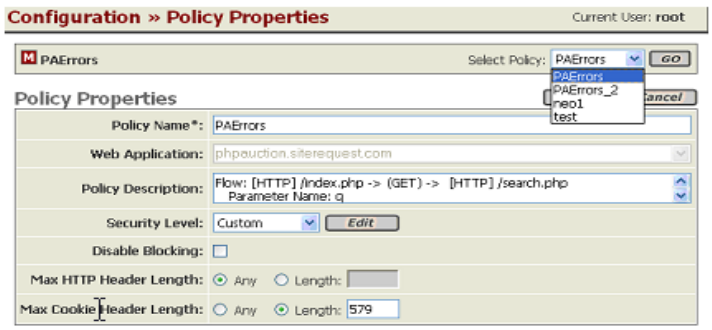
To choose a policy via the Policies List:

1. Policy Management > Policies List tab. Select the relevant policy to edit by checking the radio button at the left of the policy name.

2. Click the Edit button.
3. The policy properties window is automatically displayed for viewing or modifying.

To choose a policy via the Policy Properties window:

1. Select Policy Management > Policies Properties tab.
2. Select the relevant policy from the pull-down list Select Policy and click the Go button.
   The policy properties window is automatically updated to the selected policy for viewing or modifying.



Remove a policy

To remove a policy

1. Select Policy Management > Policies List tab.
2. Select the relevant policy to remove by checking the radio button at the left of the policy name.

Tip

You cannot remove a policy if it is active. Since it is not possible to deactivate an already activated policy, you will need to return to the Administration > Web Application tab and activate another policy that also belongs to the same Web Application. Then you can return to the Policies List tab and remove the relevant policy.
If the policy you want to remove is the only policy related to this Web Application, you will need to remove the Web Application.

3. Click the Remove button.
4. Click OK to remove the policy.

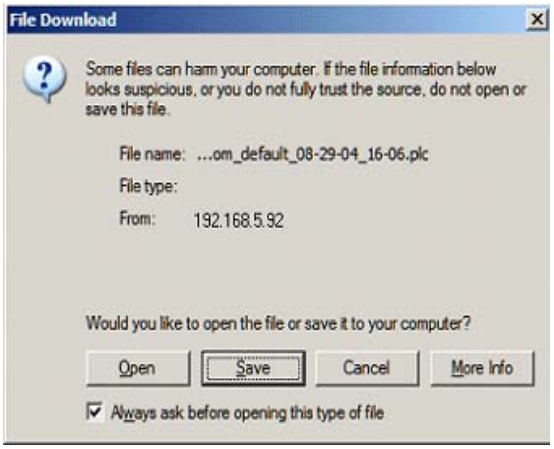
Export/Import a policy

There are different reasons for using the Export/Import policy. The export/import feature can be used to export a policy and then import it, assigning it to a different Web application in the process.

This feature can also be used as a sort of backup and roll-back point in the policy life cycle.

To export a policy:

1. In the Policy Management tool, select the Policies List tab and click the Export button.
   The Standard File Download dialog box opens.



2. Click the Save button and save the policy file.

To import a policy

1. In the Policy Management tool, select the Policies List tab and click the Import button.
   The Import Policy page opens.

2. Fill out the Import Policy page.

   For Web Application
   To populate this field, select one of the following:

   - Select Decide Automatically to assign the imported policy to the Web application from which it was exported.
   - Select another Web Application to assign the imported policy

   Choose the File
   In Choose the File, use the browser to select the file to import.

3. Click the Go button.

◆ Note

The imported policy appears in the Policies List. If the imported policy exists in the current TrafficShield security application environment, it is renamed (a sequential number is added to the end of the policy name).

Copy a policy

The purpose of this option is to quickly duplicate policies or create policies that differ only in a few details

To copy a policy:

1. In the Policy Management tool, select the Policies List tab and click the Copy button.
   The Copy Policy page opens.

2. Verify that the relevant Policy has been selected.
3. Change the selected policy in the Select Policy pull-down list.
4. Click the Go button to change the selected policy.
5. The New Policy Name field in the Copy Policy window will be automatically updated accordingly.
   You can edit the New Policy Name if required.
6. Click the Go button to copy the policy
7. In the Policies List tab verify that the newly copied policy is added to the list.

6

Crawler

- Crawler overview
- Populating the policy using the Crawler
- Configuring and launching the Crawler
- Setting the active policy of a web application
- Crawler Learning tool

Crawler overview

This chapter explains how to configure, start, and manage the TrafficShield security application Crawler tool. It also guides you through the steps needed to create an initial policy using the Crawler tool. You use the Crawler to scan your application and build a preliminary map of your Web application. This chapter also provides instructions on how to use the more advanced Crawler parameters.

Populating the policy using the Crawler

The TrafficShield security application Crawler automatically populates the security policy with the components of the Web application such as the HTML files, the picture files, the form fields, the links, and the flows that lead from one object to the other.

When you run the Crawler for the first time on a policy, it populates the policy with the current objects (application elements). The next time you run the Crawler:

- It collects only the objects that were added after the last run.
- It can be instructed to place the newly-added objects in a series of tables instead of adding them to the policy. This allows you to examine the new objects and decide what to do with them - add them to the policy or reject them. For additional details, please refer to the Data Collection with Policy Browser section in this document.

Configuring and launching the Crawler

The Crawler can be configured in many ways.

First time users should activate the Crawler Wizard. The Crawler Wizard icon is located under Policy Management > Policy Properties > Build Tools. The Wizard will guide the user through a configuration stage, and enable the user to start the Crawler.

Advanced users may prefer to manually edit the Crawler settings and manually start the Crawler.

If your Web application has several entry points, you can instruct the Crawler to scan the application from each entry point separately. This is the advised method if your Web application site is combined from two or more unconnected parts.

To configure and/or start the Crawler:

1. Select the relevant policy for which the Crawler settings will apply, Policy Management > Policies List.
2. Open the policy for editing by selecting the policy you want to work on and clicking on the Policy Properties tab or the EDIT button.
3. Go to the Build Tools section and, per your desired work mode, begin to work with the Crawler.

Configuring and starting the Crawler using the Wizard

Crawler scheduling

You can run the Crawler manually, or you can set the Crawler to run periodically. This is defined in the Crawler Scheduling section.

To set a schedule:

1. Select one of the following options:
   - **Run on user request**
     Use this option if you want to run the Crawler at your command. You can run the Crawler at any time you choose, you just click its Start button in the Build Tools section
   - **Run every... minutes**
     Use this option to automatically run the Crawler every X minutes. Click the button, and in the Run every… minutes box, type the number of minutes you want between Crawler cycles. [For instance, if you want the Crawler to run every 10 minutes, type 10.]
2. In the Crawler Scheduling window, click the **Save** button to save your settings, and continue.
   Or you can click the **Cancel** button to exit the Wizard without saving your selections.

Start points

The Crawler starts the data collection process from a URL. This is the start point.

The start point is usually the Web application's home page. However, you may instruct the Crawler to start scanning sections of the application from other points as well, in case the application includes sub-applications that cannot be accessed through the home page, but only directly from a sub-URL.

To add Crawler start points:

1. Click **Add**.
   A new line is added to start points list. The **Add New Crawler Start Point** dialog box opens.
2. In the **Domains** drop down list, select the domain to which the start point belongs.
   A start point can be specified either as part of this Web application's Fully Qualified Domain Name, or as part of one of its aliases. Select the domain or the alias to use. You must make a selection.
   The selected domain or alias appears in the Start Point text field.
3. Add the start point (a file name) to the end of the domain or alias string in the Start Point text field.
   The resulting string must be a valid path specification, or it will be rejected.
4. Repeat this procedure to define all relevant starting points.

Form filler

Since the Crawler emulates user behavior, it submits data, in Web application pages, in the same way users do.

Each time the Crawler is activated, it populates the Form Filler Parameters Table with previously undefined parameter names.

If this is the first time you start the Crawler, all parameters are new to the Crawler and therefore it will most likely fail to submit any forms.

The next logical stage is to enter the crucial values needed to properly submit forms, for example: user name, passwords, etc. Sometimes the fields' names are not self-explanatory and you will need to consult the web application programmer.

If you know what crucial parameters and values should be defined before running the Crawler the first time, you can enter them to help the Crawler utilize the Web application on the first run.

To use this feature, you specify the names and data types of the fields as well as the values the Crawler should enter in them.

To add a customized parameter:

1. In the Custom Parameters section, click the **Add** button and an empty line is displayed.
2. In the Parameter Name and Parameter Type, specify the name of the field and its data type.
3. In the Parameter Value, specify the value you want the Crawler to enter in the field.
4. Click OK.

Page not found criteria

When a request to a non-existing page comes in, Web applications return the standard HTTP 404 error page. This page may be exploited to stage attacks. To prevent this, some Web applications may use error pages of their own that don't return the HTTP 404 status code. They do this so that their content can be controlled and verified.

If your Web application uses such custom-tailored error pages, you need to supply a text string that the pages contain, so that the Crawler can identify them as a valid error message page and add it to the policy. If the "page not found" criteria is not defined, the Crawler will attempt to identify it by itself.

When an error occurs, the policy makes sure that only an error page whose content is recognized is returned to the request's sender.

TrafficShield security application can recognize an error page by its filename or by text included in its <TITLE> or <BODY>.

◆ Tip

In re-direct cases: The Crawler always follows the re-direct link. The Crawler identifies the page behind the link and avoids the link if the identified page is included in the Page Not Found list.

To identify a customized error page:

1. Click the **Add** button.
   A new empty line of page not found criteria is added.
2. In **Apply to**, select one of the following options to identify the error page:
   - Full Object Name
     Its full file name. In Search Item, enter the file name.
   - HTML Title
     The text entered in its <TITLE> section. In Search Item, enter the text.
   - HTML Body
     Any string of text that appears in its <BODY> section. In Search Item, type the string.
3. In **Search Item**, enter the indicated value and click OK.

Logout pages

If the Web application contains a page designed to log the Web application visitor out, you need to instruct the Crawler not to follow the logout link as this will cause the Crawler to log out of the application before has fully scanned the application. In fact, many Web applications have an "exit" or "logout" link right in their home page, which would cause the Crawler to exit as soon as it enters the application. To prevent this, use the Logout Pages section to identify the logout points that the Crawler should avoid.

◆ Note

The logout page will be added to the policy.

To define a logout point:

1. Click the **Add** button.
   A new empty line of Logout Pages is added.
2. In **Logout Pattern (URL)**, enter the relative path of the logout page.
3. Click **OK**.

Properties

The Properties section provides additional instructions to the Crawler. For example, you can instruct the Crawler to analyze Java Script code included in the Web Application or to skip it.

Enter the following information. Click the Save button in the Properties window to save your entries.

Analyze JavaScript

Check this box to instruct the Crawler to analyze the JavaScript code included in the Web application. This is useful if the scripts contain links that can be followed, or if they include fields that need to be filled.

Clear the box if JavaScript analysis is not necessary.

Accept un-trusted SSL certificates

An un-trusted SSL certificate is used by the Web application and this checkbox option is checked, the Crawler accepts the SSL certificate and continues scanning.

Clear this box to instruct the Crawler to accept only trusted certificates.

Create back flows

As the Crawler runs, it always registers the page that follows a certain page over a link, thus adding the application flows to the policy. You can access each such flow definition and further configure it in order to establish rules of passage from one page to another.

By checking this box, you instruct the Crawler to also register in the policy all flows in the opposite direction, in which case you can also impose rules on navigating backwards (which occurs when the visitor uses the Back button).

Create cache flows

Cache flows are created around cacheable objects. The flow is created from the first non-cacheable referrer object around the cacheable object. The parameters of the incoming flow will be added to the newly created cache flow.

When no previous non-cacheable referrer object is found, the cacheable object itself becomes the entry point and the flow is added.

Min. delay between worm requests to web application (in sec.)

The Crawler is a mechanism that can be likened to a central unit sending out multiple probes to the different areas of the Web application in order to register Web application components simultaneously. Each probe behaves as if it were a real user, following links and filling in forms, and therefore increases traffic.

The probes can be sent in quick or slow succession. Quicker bursts create more traffic. A burst is measured in terms of the number of seconds to wait before sending the next probe. If your Web application is active and currently serving visitors, consider increasing this value in order to slow down the Crawler.

Number of threads to be used by the Crawler

This parameter also relates to simultaneous probe activity. A smaller number decreases the Crawler's bandwidth consumption, leaving more bandwidth to actual visitors.

Number of times the Crawler fetches requests with the same structure

Applications usually have many identical structures where only the parameter values differ. The following examples illustrate identical links passing different parameter values:

http://www.myapp.htm?par=111

http://www.myapp.htm?par=222

http://www.myapp.htm?par=333

To reduce crawling time and traffic you can instruct the Crawler to scan only a few of such identical structures and not all of them, assuming that all others behave in the same way.

Specify the number of samples you deem it sufficient for the Crawler to scan. A higher value yields a more accurate policy with longer crawling times.

Maximum number of requests generated for each form by the form iterator

When the Crawler encounters a form, it processes it as many times as the number of pre-defined parameter values included in it. For example, a drop-down list containing ten values causes the Crawler to process the form ten times, each time with a different value. However, you can reduce crawling time and traffic by instructing the Crawler to process only a few of the values and not all of them.

Specify the number of samples you deem it sufficient for the Crawler to process from the same form with different values. A higher value yields a more accurate policy with longer crawling times.

Emulate browser

If your Web application is set to work only with a given Internet browser, set the relevant browser name.

This name will be used to select the user-agent header data.

Default character set for user input fields

Select the character set in which data is normally entered in the form fields of the scanned application. This value will be used as the default value for all new policy fields added by the Crawler.

HTTP authentication

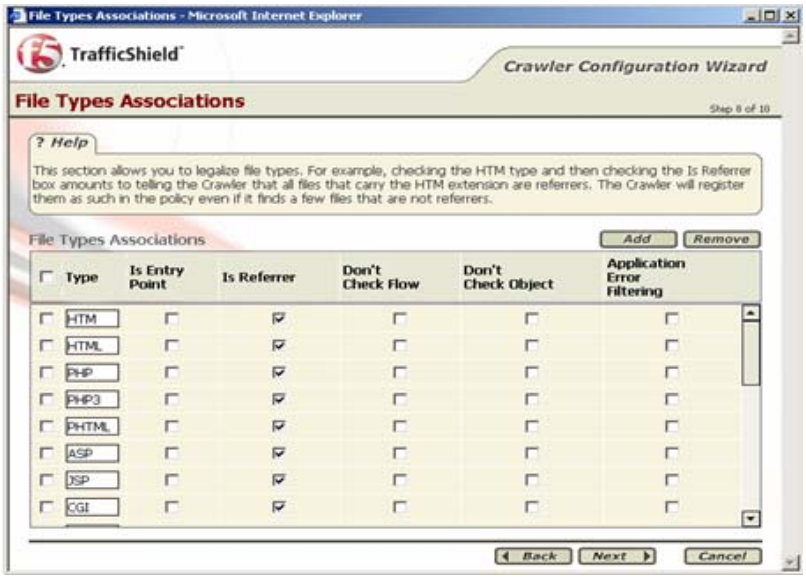
Use this option only if your Web application uses HTTP authentication.

Specify the user name and password the Crawler should supply in order to access the server where the Web application resides.

File type associations

This section provides a list of file types frequently used in a Web application and their most common usage in the Web application.

It allows you to configure file types globally, thus saving tedious manual configuration in the policy. For example, you can instruct the Crawler to define all BMP files as files that do not have a flow.



If the list does not include a file type, you need to configure it.

- Click the Add button, add a file extension, and click OK.

The defaults provided in this page cover the most plausible eventualities, but you can adapt them to your needs by checking or clearing boxes.

A description of the file type configuration parameters follows.

Is Entry Point
Check this box if all files of this type can be entry points to the Web application.

Is Referrer
Check this box if objects of this object type may refer to other files. For example, HTML pages containing a link or CGI files calling another file, are waverers. Pictures and sound files cannot be waverers because these objects never contain links to other objects and are not web pages.

Don't Check Flow
Check this box if you don't want the system to check the flows to objects of this file type.

Don't check object
Check this box to if you don't want the system to check the requests referring to files of this type.

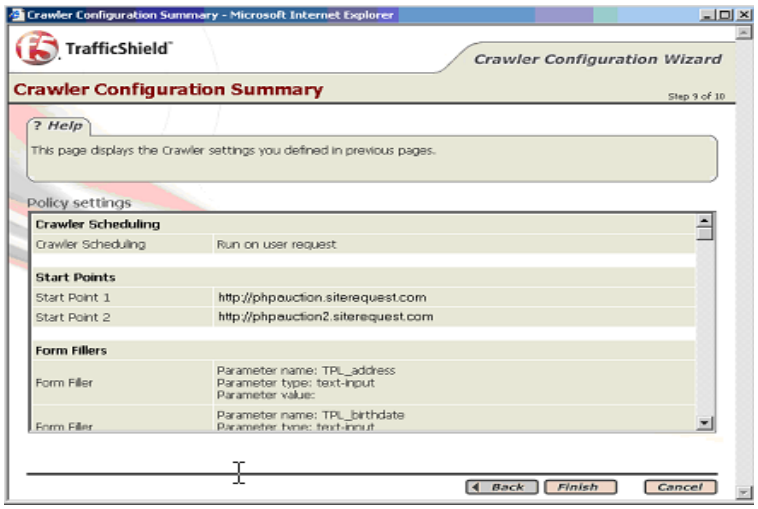
Note

This will also be applied to files that do not exist in the application.

Crawler configuration settings

This page displays the Crawler settings you defined in previous pages.

To modify the configuration click the Back button until you reach the relevant step, and modify the data.



To manually configure the Crawler:

1. Click the Settings button. The Crawler settings window appears. Each group of parameters is displayed in a separate box.
2. Enter the Crawler settings as described in the subsequent sections.
3. Return to Policy Properties by clicking the Back button, found on the upper left side of the policy properties window.

Crawler scheduling

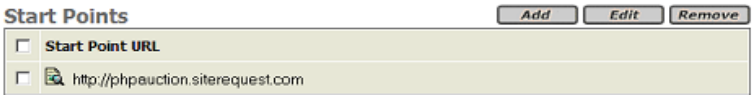
In the Crawler Scheduling section, you can define whether to run the Crawler manually, or set it to run periodically.

To set a schedule:

1. Select one of the following:
   - **Run on user request**
     To set the crawler run on user request, Click any time you want on the Crawler Start button that you can access through the Build Tools section.
   - **Run every x... minutes**
     To set the Crawler to automatically run every x... minutes, in the Run on user requestSelect this option to run the Crawler whenever you want by clicking its Start button in the Build Tools section.
2. Click the **Save** button in the **Crawler Scheduling** window.

Starting points

The start point is a URL from which the Crawler starts the data collection process.



The start point is usually the application's home page. However, you may instruct the Crawler to start scanning sections of the application from other points as well in case the application includes sub-applications that cannot be accessed through the home page, but only directly from a sub-URL.

To add Crawler starting points:

1. Click the Add button in the Start Points section.
   The Add New Crawler Start Point dialog box opens.
2. In the Domains drop down list, select the domain to which the starting point belongs.
   - A start point can be specified either as part of this Web application's Fully Qualified Domain Name or as part of one of its aliases. Select the domain or the alias to use. You must make a selection.
   - The selected domain or alias appears in the Start Point text field.
3. Add the starting point (a file name) to the end of the domain or alias string in the Start Point text field.
   The resulting string must be a valid path specification or it will be rejected.
4. Repeat this procedure to define all relevant starting points

Form filler

Since the Crawler emulates user behavior, it submits data, in Web application pages, in the same way users do.

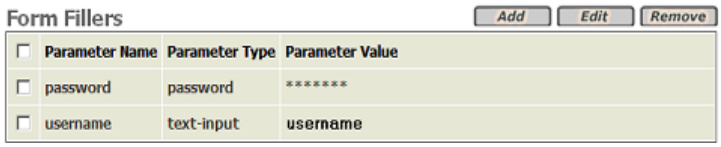
Every time the Crawler is started, it populates the Form Filler Parameters Table with previously undefined parameter names.

If this is the first time you start the Crawler, all parameters are new to the Crawler and therefore it will most likely fail to submit any forms.

The next logical stage is to enter the crucial values needed to properly submit forms, for example: surname, passwords, etc. Sometimes the fields' names are not self-explanatory and you will need to consult the web application programmer.

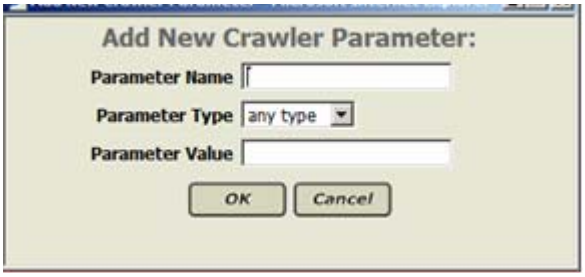
If you know what crucial parameters and values should be defined before running the Crawler the first time, you can enter them to help the Crawler utilize the Web application on the first run.

To use this feature, you specify the names and data types of the fields as well as the values the Crawler should enter in them.



To add a customized parameter:

1. In the Custom Parameters section, click the Add button. The Add New Crawler Parameter dialog box opens.



2. In Parameter Name and Parameter Type, specify the name of the field and its data type.
3. In Parameter Value, specify the value you want the Crawler to enter in the field.
4. Click OK.

Tip

If the parameter in question is a password type, you will be asked to enter the value twice. The value will not be displayed.

Page not found criteria

When a request to a non-existing page comes in, Web applications return the standard HTTP 404 error page. This page may be exploited to stage attacks. To prevent this, some Web applications may use error pages of their own that don't return the HTTP 404 status code. They do this so that their content can be controlled and verified.

If your Web application uses such custom-tailored error pages, you need to supply a text string that the pages contain, so that the Crawler can identify them as a valid error message page and add it to the policy. If the “page not found” criteria is not defined, the Crawler will attempt to identify it by itself.

When an error occurs, the policy makes sure that only an error page whose content is recognized is returned to the request's sender.

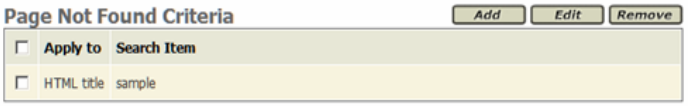
TrafficShield security application can recognize an error page by its filename or by text included in its <TITLE> or <BODY>.

◆ Tip

In re-direct cases: The Crawler always follows the re-direct link. The Crawler identifies the page behind the link and avoids the link if the identified page is included in the Page Not Found list.

To identify a customized error page:

1. In the Page Not Found Criteria section, click the Add button. The “Add new page not found criteria” box opens.



2. In “Apply to”, select one of the following options by to identify the error page, and in Search Item enter the indicated value.
   - **Full Object Name**
     In Search Item, enter the file name.
   - **HTML Title**
     The text entered in its <TITLE> section. In Search Item, enter the text.
   - **HTML Body**
     Any string of text that appears in its <BODY> section. In Search Item, type the string.
3. Click OK.

Logout pages

If the Web application contains a page designed to log the Web application visitor out, you need to instruct the Crawler not to follow the logout link, as this will cause the Crawler to log out of the application before it was fully

scanned. In fact, many Web applications have an "exit" or "logout" link right in their home page, which would cause the Crawler to exit as soon as it enters the application. To prevent this, use the Logout Pages section to identify the logout points that the Crawler should avoid.

Tip

The logout page will be added to the policy

To define a logout point

1. In the Logout Pages section, click the Add button.
   The "Add new logout page" box opens.
2. In "Logout Pattern (URL)", enter the relative path of the logout page.
3. Click OK.

Data collection with policy browser

The Policy Browser collects data that the Crawler can later use as a sort of fine-tuning input. The Policy Browser also overcomes browsing obstacles.

The data is collected by simply browsing the application as you would browse it with a regular browser. The browsing information processed by the browser is stored in a file. It is advisable to use the Policy Browser extensively and let it collect as much data as possible to later help the Crawler create a more accurate policy.

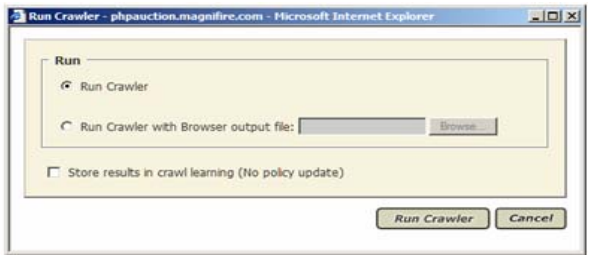
For instructions on how to download the policy browser and how to create the input file refer to the *Downloads* section in Chapter 6 *Administration*, of the ***TrafficShield™ Installation and Configuration Manual Version 3.1***.

Running the Crawler

To manually start the Crawler

1. Select the relevant policy for which the Crawler settings will apply, from the **Policy Management > Policies List**.
2. Open the policy for editing by selecting the policy you want to work on and clicking on the Policy Properties tab or the EDIT button.

3. In **Build Tools**, click the Crawler's **Start** button.
   The Run Crawler dialog box opens.



4. Select the appropriate options and click Run Crawler to run the Crawler, or Cancel to exit without running the Crawler.

   Run Crawler
   Choosing this radio button runs the Crawler as is, without the additional information supplied by the Policy Browser.

   Run Crawler with policy browser output file
   Run the Crawler and also use Web application details pre-recorded by the Policy Browser. Click the Browse button and select the Policy Browser's output file. For additional information on how such a file is created please refer to the Data Collection section in this Chapter.

   Store results in crawl learning (no policy update)
   Check this checkbox to activate the Crawler Learning process. For more details, please refer to *Crawler Learning tool* section on page 19 of Chapter 6 , at the end of this chapter.

5. Click the Run Crawler button.
   The Crawler starts collecting data.
   While the Crawler is running, you can click the **Status** button to open a window where you can see how the operation is progressing.

   The message "Running" appears at the top of the window while the Crawler is still running. During this time, the dialog box displays the number of objects and flows that have been scanned and identified. Click the Status button to display the current status, without waiting for the next automatic refresh operation. The status

window title changes to "Finished" when the operation ends. You can also monitor the process by accessing the other tabs in the navigation bar on the left.

Policy-specific negative regular expressions

When you create a new policy, the policy automatically inherits all of the negative regular expressions defined in the Administration tool, and these expressions are listed in this tab.

Existing policies do not inherit expressions that have been created after them. You can add policy-specific negative regular expressions by choosing the tab under **Configuration > Negative RegExp** and add them just like adding default Regular Expression.

For more details, see the *Assigning Expressions* section in Chapter 6, *Administration*, of the ***TrafficShield™ Installation and Configuration Manual Version 3.1***

◆ Tip

Violations created due to Negative Regular Expressions are related to illegal pattern violations.

Setting the active policy of a web application

At any given time, TrafficShield security application enforces only one of the available security policies. The security policy according to which the Web application is currently protected is called the active security policy.

You need to set the active security policy in the following cases:

- Before opening the Web application to user traffic, for testing or for regular business.
- Every time that you enter a change in the policy. If you do not re-activate the policy, the latest changes are not reflected to the Web application. A policy that has not been activated after it has been modified is marked with the  icon.
- Whenever you switch from one policy to another.

To activate a policy:

1. Select the **Administration** button.
2. Click the **Web Applications** tab.
   The defined Web applications are listed.
3. In the Active Policy drop-down list, select the security policy to apply to the Web application.

   When you select a policy, TrafficShield security application automatically selects the Web application by marking its radio button.
4. Click the **Set Active Policy** button.

Crawler Learning tool

This section explains how to use the Crawler Learning tool and how to adapt the policy using the Crawler Learning tool's output.

The Crawler Learning tool enables the user to scan the Web application in a learning mode.

When we use the Crawler in a non-learning mode, the Crawler populates the policy with the new items.

When the Crawler is set to work in a Learning mode, it populates the crawler learning tables with the new items instead of directly populating the policy tables.

You can then review the data and accept object types, objects and flows that were found by the Crawler and then add or reject them.

Crawler Learning tabs are identical to the Learning tabs. Both Learning and Crawler Learning populate the forensics section.

First-time usage: Crawler Learning can be used to update an existing policy or to initialize a policy. When updating a policy, the Crawler works in update mode and writes all the incrementally new items to the Crawler Learning tables. It doesn't change the existing policy items. When populating an empty policy, all items appear in the Crawler learning tables. In both cases you need to accept the item if you want to add it to the policy.

Second time usage: Unlike the regular Learning, once the Object is accepted and added to **Configuration > Web Objects** tab, all relevant flows are not automatically added to the policy. In order to add the relevant flows, you will need to re-run the Crawler or the Crawler learning.

◆ Tip

If an item is rejected permanently, it is moved to Forensics > Ignore Items. This affects the Learning stage as well. For more details, please refer to the Ignored requests section on page 46 of Chapter 7 .

7

Learning - Testing & Fine Tuning the Policy

- Overview
- Access violations
- Length violations
- Input violations
- Negative security violations
- Cookie violations
- Forensics
- Policy component editing
- Policy audit tools

Overview

After automatically generating a policy using the Crawler and making any manual changes needed, you are ready to test and refine the policy in real-life conditions, through the Learning tool and the Policy editing tools.

This chapter explains how to use the Learning tool to adapt the policy to real-life traffic requirements. It also covers the Policy editing feature, which allows you to view and manually adjust the entire security policy.

Learning tool

The Learning tool was created so you could fine tune the Crawler-created security policies. This is relevant both for the first activation of the TrafficShield security application and as an ongoing tool as well.

In each case, the Learning screens are actually suggesting changes to the policy which would include all future requests of this nature. You can accept objects or flows that were rejected by the TrafficShield security application, and reject changes to the policy that were caused by actual attacks which were screened out.

◆ Tip

Customize your blocking definitions to temporarily allow some violations to go through until the Learning fine-tuning is more complete.

First-time usage: As the Web application is new; you may prefer to run an initial test in safe conditions. Such conditions can be created by opening the Web application to a limited number of visitors like Quality Assurance (QA) and employees of your organization (persons who are not potential hackers). Initially, the alarms help you adjust policy attribute values until you are sure that the policy is usable. Any invalid request that might come after the Learning stage can justifiably be considered illegal and treated as such. In fact, after the initial testing period you can use the Learning tool to track real attacks.

Ongoing usage: If the Web application to be protected is already in use, a portion of the live traffic can be diverted through the TrafficShield security application to the Learning tool.

As visitors move through the Web application, the TrafficShield security application captures requests that contradict your current policy settings, and posts alarms to the Learning tool pages.

The Learning tool checks that all objects that are supposed to exist in your Web application are indeed present (for example, all links lead to objects that exist in the Web application). It also checks that the attributes specified for policy objects, such as URI lengths or allowed meta characters, are realistic.

In all the Learning windows, the fine-tuning changes can be applied to a specific policy:

The Learning fine-tuning changes can also be applied to a Web application which will affect all related policies:

In the Learning tool, the System displays recommendations to the user on how to fine tune the policy. Learning is not an analysis tool. There are situations that will be recorded in the Forensic that will not be converted into a policy. Note that the Learning tool saves Learning recommendations in the Learning tables at account level for the whole account. If a policy is deleted, the learning recommendations will be saved and displayed.

If you have several policies that are related to the same web application, in order to build a policy, you must first ensure that the policy is active, and then select its radio button in the Policies List screen.

Learning duration

The aim of the Learning process should be to generate traffic on all pages, to click all links, to fill all form fields, and so on. For new web applications, standard customer workflow routines can be used for Learning. For live applications, even a 15 minute test might supply valuable information that will help you fine-tune the policy. Obviously, the longer the test, the greater the opportunities to capture information that may help you establish a safer policy.

Selecting the flow mode

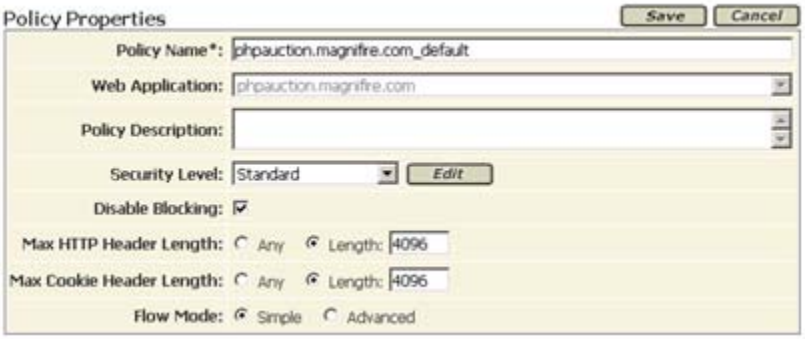
Two flow modes are available: Simple and Advanced. The Simple flow mode is the default mode.

The flow mode is applied in the Policy Properties screen.

- By selecting the Simple button in the Flow Mode area, the user is instructing the TrafficShield system to create a simplified policy, where all objects are defined as entry points. This is true whether the user uses the Crawler to create the policy or decides to manually create a policy.
- By selecting the Advanced button in the Flow Mode area, the user instructs TrafficShield system to automatically create the policy.

◆ Tip

Always maintain the same Flow Mode option that was used to initially create a specific policy. We do not recommended that you switch back and forth between Simple and Advanced flow modes.



Auto Accept build tool

The Auto Accept Build tool enables the Security Manager to adapt the policy to accept automatically specific illegal requests recorded in the Forensics and make them legal.

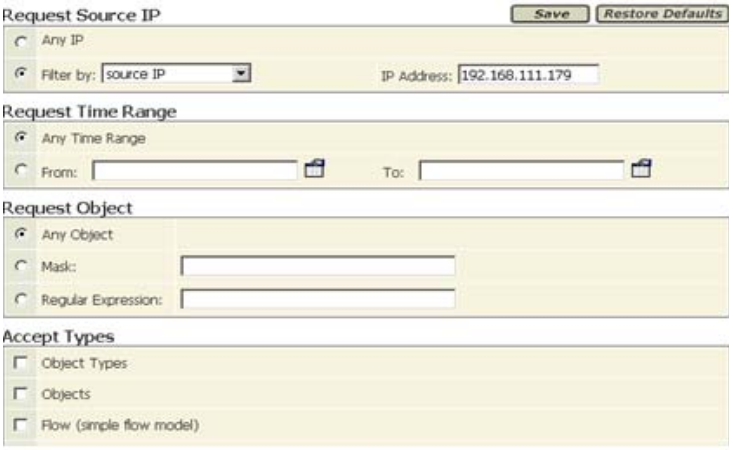
◆ Note

The Auto Accept tool must be handled with ultimate care due to its immediate and comprehensive impact on the policy, as it automatically includes the selected violations into the policy, making them legal. In this aspect it is distinguished from the Learning procedure, which provides only hints about the violations and requests the user to accept each of them manually into the policy.

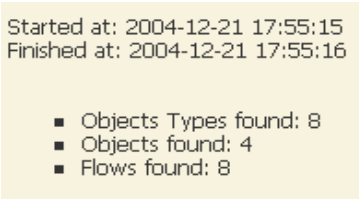
To access the Auto Accept tool

1. From the Policy Management tool, click Policy Properties > Build Tools.

2. Click Settings to open the Settings screen.



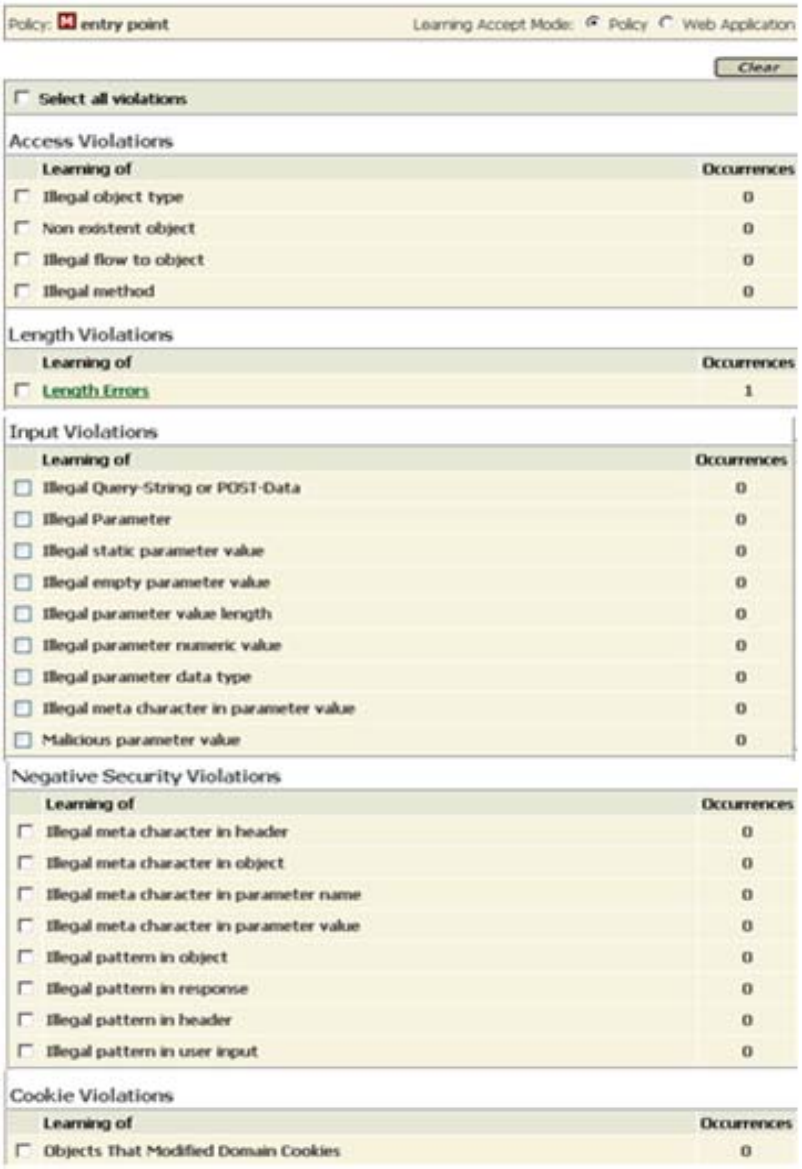
3. Select the appropriate Request source IP, Request Time Range and Requested Objects. These are the filters according to which the requests will be filtered. In the Request Object section you can limit the filtering by Mask and Regular Expressions.
4. In the Accept Types section, define the objects that you wish the policy to accept as entry points.
5. Once the settings are completed, click **Save** to save the settings.
6. Click the Back button on the top left side of the screen.
   You are returned to the previous screen.
7. Click the Start button.
   You are required to confirm the Auto Accept run.
8. Click Run Auto Accept button.
   The Auto Accept process starts running and upon completion, an information message appears, providing information about the process.



Accessing the Learning data

To access the Learning data:

- In the Policy Management Module, select Learning> Real Traffic.
  The Real Traffic screen opens and a comprehensive list of violations groups appear.



◆ Note

*The **M** that appears in the top menu next to the Policy name indicates that a modification has been done to the policy. Although all changes made to the policy were recorded in the database, they are not yet implemented until you activate the policy by clicking the Set Active Policy button in the Administration > Web Applications tab. Until then, the policy will act according to its previously defined parameters.*

If actual violations occurred for a specific violation, then the violation appears in green and it is underlined and linked.

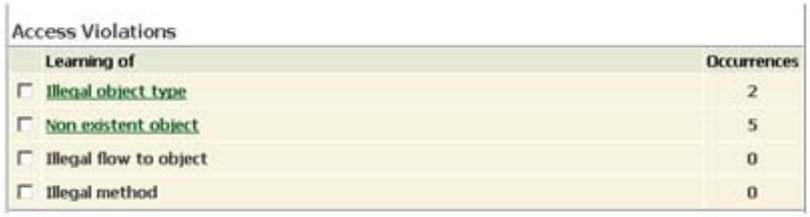
- Select the policy violation you wish to review.

Violation grouping

Violations detected by the TrafficShield Security module are grouped as follows:

- Access Violations
- Length Violations
- Input Violations
- Negative Security Violations
- Cookie Violations

Access violations

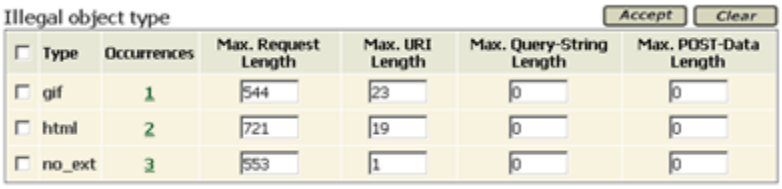


This section is divided into four parts:

- Illegal object type
- Non existent object
- Illegal flow to object
- Illegal method

Illegal object type

The Illegal object type window lists information about requests that referenced object types not found in the Web application. The object type is considered undefined unless you define it in the **Configuration > Object types** section.



It is possible to manually change the value of some of the parameters. If the parameter is editable, it will appear as a user input box.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Type

Check the checkbox for the relevant Object (file) type that you want to add to the policy.

Occurrences

This number indicates the number of request occurrences that were rejected for this type of violation.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

For more details please refer to the *View Full Requests Information window* section, on page 48 in this chapter.

Max. Request Length

The maximum request length received from all the requests for this object type.

Max. URI Length

The maximum URI length received from all the requests for this object type.

Max. Query String Length

The maximum Query string length received from all the requests for this object type.

Max. POST Data Length

The maximum Post data length received from all the requests for this object type.

Available actions for Illegal Object Type

Accept

Clicking the Accept button adds the changes to the policy.
Accept means that you have decided that the request reflects a real-life situation that warrants a change in the policy.

The undefined objects types will appear under the **Configuration > Object Types** section**.**

When you accept an Object type, the non-existent object window is automatically populated and displayed with all the objects belonging to all the requests for this object type. For example; if you accepted an HTML object type, all HTML requests' objects will now appear in the non-existent object window. See the next section to learn more about how to accept a non-existent object.

◆ Note

Requests with the accepted object types will still not be allowed by the TrafficShield security application until all the request's components have been "learned".

Clear

Clicking Clear deletes the selected entries in this learning window without changing the policy. The confirmation window appears.

Permanently reject items from learning

Select the "Permanently reject items from learning" checkbox to delete the request and instruct the TrafficShield security application not to register again identical requests. The deleted request is stored in the Forensics > Ignored Items.

Note

After transferring the requests to Ignored Items, all similar requests for all policies that belong to this Web application will ignore these requests.

WARNING

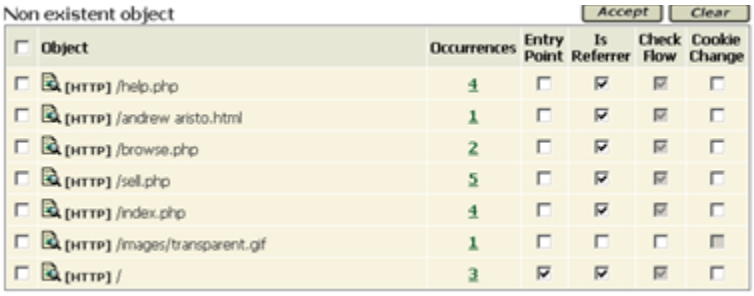
If you only want to apply this clear to this specific policy - don't check this checkbox. For example: if you checked this checkbox for HTML requests, all HTML requests (even rejected requests coming in for other policies will be ignored).

Tip

To change this decision after clicking Ok, you can go to Policy Management > Forensics > Ignored Items tab to unset the ignore decision. For more details, see the Forensics section in this document.

Non-existent object

The Non Existent Object window lists information about requests that referenced objects that are not found in the policy.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Object

This column displays the name of the non-existent object.

Occurrences

This number displays the number of requests and values that caused this violation.

Entry Point

An entry point is a page through which a visitor enters the Web application, for example, by typing its URL in the browser's address box or by selecting its URL from a favorites list.

By checking this checkbox you instruct the TrafficShield security application to consider this object as a valid entry point.

Is Referrer

Check this box if files of this type may refer to other files. For example, HTML pages containing a link or CGI files calling another file, are referrers. Pictures and sound files cannot be referrers because they do not link to any other pages.

Check Flow

The Application Flow (path) is the defined access path leading from one object to another object. For example, a list of valid flows would be:

from abc.html to abc.gif, OK

from abc.html to def.html, OK

If your policy contains the above list, then any request that tries to access abc.gif from def.html would be considered illegal.

Check this checkbox to instruct the TrafficShield security application to verify that the object was accessed by a legally defined flow.

Some of these checkboxes are checked by default and cannot be cleared by the user.

If you clear the checkbox, the object can be requested from any place in the Web application or even when the user is outside the scope of the application.

Cookie Change

Select this checkbox if the object modified one of the Web application cookies in order to prevent false positive alarms on cookie poisoning.

Available actions for non-existent objects

Accept

Clicking the Accept button adds the changes to the policy. Accept means that you have decided that the request reflects a real-life situation that warrants a change in the policy.

Clear

Clicking Clear deletes the checked entries from this learning window without changing the policy. A confirmation window is displayed.

Permanently reject items from learning

Select the "Permanently reject items from learning" option to delete the request and instruct the TrafficShield security application not to register again identical requests in the Learning tables. The deleted request goes to Forensics > Ignored Items.

Note

After transferring the requests to ignored items, all similar requests for all policies that belong to this Web application will ignore these requests.

WARNING

***I**f you only want to apply this clear to this specific policy, don't check this checkbox.*

Tip

To change this decision after clicking Ok, you can go to Policy Management > Forensics > Ignored Items tab to unset the ignore decision. For more details, see the Forensics section, on page 44 in this chapter.

Illegal flow to object

The Illegal Flow to Object screen is divided into two sections:

- Illegal Flow to Object
- Illegal Entry Point.

The Illegal Flow to Object screen lists the flows that were requested but were not found in the policy. In this case too, you can configure the query string and POST data settings of the Illegal flow to object flow and include them in your policy by clicking the Accept button.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Flow

This is the name of the Application Flow (path) which defines the access path leading from one object to another object.

Note

The  *indicates that the object is not a referrer. If the object should be defined as a referrer, go to the Policy Management > Configuration > Web Object window, and modify the definition of the object so that it is defined as referrer. Only after this operation is completed, it is possible to accept the violation.*

Method

This is the HTTP method used in the Request. For more details refer to RFC-2610 (HTTP).

Occurrences

This field displays the number of illegal flow to object violation occurrences.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Frame Target

This is the index of the HTML frame targeted by the flow. It is not recommend that you change this value unless you know that you want to specifically load this object into a specific frame.

User Input

User Input fields allow the user to enter a valid value that overrides the defaults.

The value 99 is a default frame index which indicates that the target object is loaded into the same frame as where the referrer object is presented. An empty value in the Frame target is allowed and accepting this empty value accepts it automatically under the 99 value.

Click the magnifying glass icon next to the frame target value to open its screen

Allow QS/PD

Check this box if a request that accesses the selected object via this specific flow may also carry a query string or POST method.

Check QS/PD

Check this box to instruct the TrafficShield security application to perform validity checks on the query string or POST data if allowed in the previous step.

If you clear the checkbox, the object can be requested from any place in the Web application or even when the user is outside the scope of the application.

Available actions for illegal flow to object

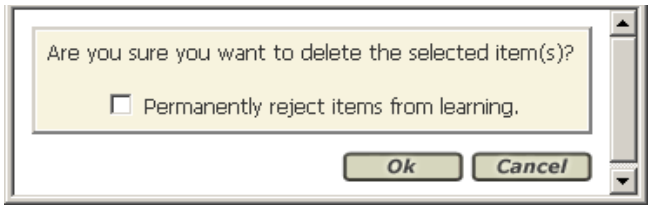
Accept

Clicking the Accept button adds the changes to the policy.

Accept means that you have decided that the request reflects a real-life situation that warrants a change in the policy.

Clear

Clicking Clear deletes the checked entries from this learning window without changing the policy. The confirmation window is displayed.



Permanently reject items from learning

Check the "Permanently reject items from learning" checkbox to delete the request and also instruct the TrafficShield security application not to register again identical requests in the Learning tables. The deleted request goes to Forensics > ignored items.

Note

After transferring the requests to ignored items, all similar requests for all policies that belong to this Web application will ignore these requests.

WARNING

***I**f you only want to apply this clear to this specific policy, don't check this checkbox.*

Tip

To change this decision after clicking Ok, you can go to Policy Management > Forensics > Ignored Items tab to unset the ignore decision. For more details, see the Forensics section in this document.

Illegal entry point

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Flow

This is the entry point access to the object.

Method

This is the HTTP method used in the Request. For more details refer to RFC-2610 (HTTP).

Occurrences

This field displays the number of illegal flow to object violation occurrences.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Frame Target

This is the index of the HTML frame targeted by the flow. It is not recommend that you change this value unless you know that you want to specifically load this object into a specific frame.

User Input

User Input fields allow the user to enter a valid value that overrides the defaults.

An empty value in the Frame target is allowed and accepting this empty value accepts it automatically under the 1 value.

Click the magnifying glass icon next to the frame target value to open its screen.

Allow QS/PD

Check this box if a request that accesses the selected object via this specific flow may also carry a query string or POST method.

Check QS/PD

Check this box to instruct the TrafficShield security application to perform validity checks on the query string or POST data if allowed in the previous step.

Available actions for illegal entry point

Accept

Clicking the Accept button adds the changes to the policy.

Accept means that you have decided that the request reflects a real-life situation that warrants a change in the policy.

Clear

Clicking Clear deletes the checked entries from this learning window without changing the policy. The confirmation window is displayed.

Permanently Reject items from Learning

Check the "Permanently reject items from learning" checkbox to delete the request and also instruct the TrafficShield security application not to register again identical requests in the Learning tables. The deleted request goes to Forensics > ignored items.

Note

After transferring the requests to ignored items, all similar requests for all policies that belong to this Web application will ignore these requests.

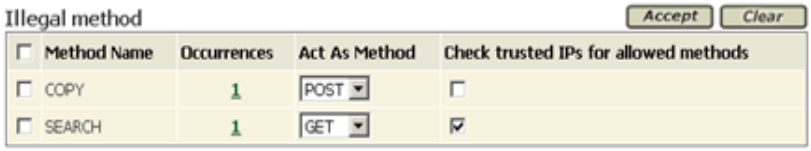
WARNING

If you only want to apply this clear to this specific policy, don't check this checkbox.

Tip

To change this decision after clicking Ok, you can go to the Policy Management > Forensics > Ignored Items tab to unset the ignore decision. For more details, see the Forensics section in this document.

Illegal method



Checkboxes

The first column contains checkboxes used to mark the relevant entry

Method Name

Describes the Method name

Occurrences

Displays the number of illegal methods occurrences detected.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.

- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Act as Method

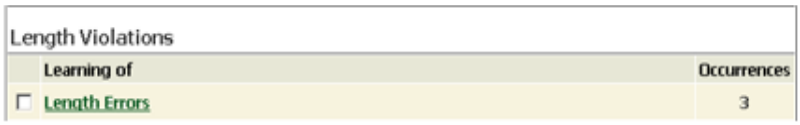
For each Method, set a corresponding GET or POST option.

Check Trusted IPs for Allowed Methods

Selecting this checkbox instructs the TS Security Mechanism to check for Trusted IP numbers that are allowed to use the method.

Length violations

Length violations are detected as Length Errors.


Length Violations

	Learning of	Occurrences
☐	Length Errors	3

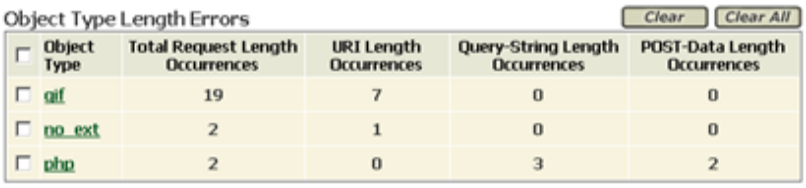
This section lists the requests that exceeded a length setting.

This section is divided into two categories:

- Object Type Length Errors
- Header Length errors

Object type lengths errors

This section lists the requests that exceeded a length setting.


Object Type Length Errors

Clear | Clear All

	Object Type	Total Request Length Occurrences	URI Length Occurrences	Query-String Length Occurrences	POST-Data Length Occurrences
☐	gif	19	7	0	0
☐	no_ext	2	1	0	0
☐	php	2	0	3	2

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Object Type

Select the checkbox for the relevant Object (file) types that you want to clear. If you want to define and accept this object type length, you will need to click on the relevant Object type link, and the Requests Lengths for "object" type window will be displayed. For more details, see the Accept Requests Lengths section in this chapter.

Total Request Length Occurrences

The Total Request Length is the sum of the URI, Query string and POST data lengths in a specific request.

URI Length Occurrences

The maximum URI length received from all the requests for this object type.

Query-String Length Occurrences

The maximum Query string length received from all the requests for this object type.

POST-Data Length Occurrences

The maximum Post data length received from all the requests for this object type.

Available actions for object type length errors

Clear

Deletes the checked entries from this learning window without changing the policy. A confirmation window is displayed.

Clear All

Deletes all entries from this learning window without changing the policy, regardless of whether their checkbox is selected or not. A confirmation window is displayed.

Click the Object Type Link

Displays the object type length window.

«php» object type lengths — Accept All

Length Type	Current Max Length	Detected Max Length	Detected Average Length	Occurrences	Accept
Total Request Length	1075	2764	2764.0	1	3593 Accept
URI Length	14	16	15.5	2	20 Accept
Query-String Length	5	67	22.2	6	87 Accept
POST-Data Length	0	2016	2016.0	1	2620 Accept

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Length Type

There are four length types. The Total Request Length is the sum of the other three types.

Current Max Length

The length set in the policy. For example, the Current Max Length (column) for URI Length (row) indicates the valid length defined in the policy for the URI section of the request.

Detected Max Length

This value indicates the highest length value that has been detected for a specific policy object.

Detected Average Length

This value indicates the average length value that has been detected for a specific length object. If the average length is very different from the Max length, this could indicate a problem that requires further investigation.

Occurrences

This is the number of requests that have been rejected for violating the length constraints.

Clicking the number opens the Full Request Information window that contains all the technical details of all the violations related to the longest request.

User Input

User Input fields allow the user to enter a valid value that overrides the defaults.

Actions available for accept requests lengths

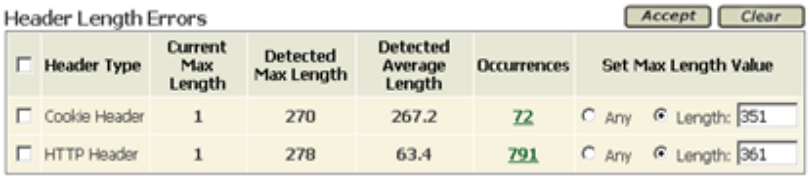
Accept
Choose the Accept button on the relevant length type row if you decide that the returned statistics reflect a real-life situation that warrants a change in the policy. You can also decide to manually define the new length in the user input field in the Accept column. The decision should be based on an in-depth understanding of your Web application.

Accept All
Choose the Accept All button if you decide that all the length types displayed reflect a real-life situation that warrants a change in the policy. You can also decide to manually define all new lengths in the user input fields in the Accept column. The decision should be based on an in-depth understanding of your Web application.

To return to the Real Traffic tab

Click the arrow button on the top left corner.

Header length errors



There are two possible Header length violations:

- HTTP Header
- Cookie Header

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Header Type

Check the checkbox for the relevant Header type that you want to clear or accept. If you want to define and accept this Header type length, you will need to click on the relevant Header Type, and the Requests Lengths for Header Type window will be displayed. For more details, see Actions available for accept requests lengths, in this chapter.

Current Max Length

The valid length defined in the policy for the Header length.

Detected Average Length

This value indicates the average Header length that violated the Header length constraint.

Occurrences

This number displays the number of requests that caused this violation.

Set Max Length Value

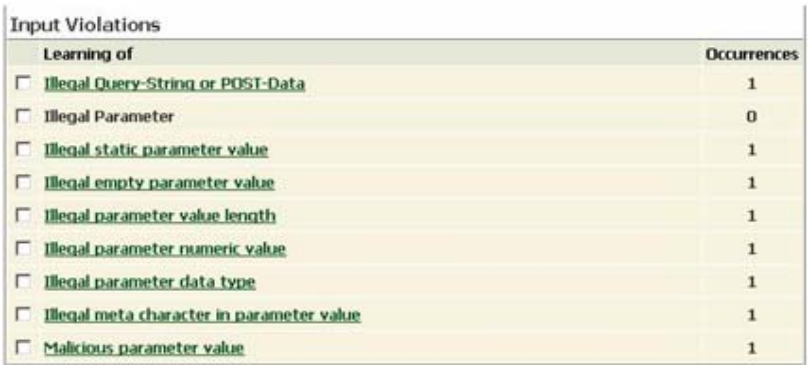
You can manually change the maximum length allowed for the Header Type or select the Any option to allow any length.

Actions available for Header Length

Accept
Clicking the Accept button adds the changes to the policy. Accept means that you have decided that the request reflects a real-life situation that warrants a change in the policy. It is possible to manually change the value of some of the parameters.

Clear
Clicking Clear deletes the entry from this learning window without changing the policy. A warning message appears asking to confirm the deletion.

Input violations

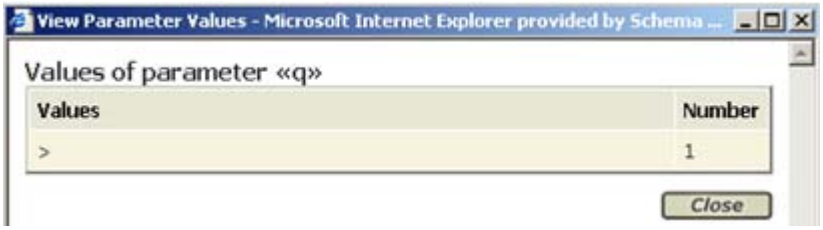


Input violations are classified by the TrafficShield Security Module as follows:

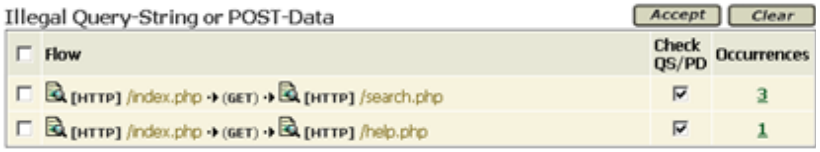
- Illegal Query-String or POST-Data
- Illegal Parameter
- Illegal static parameter value
- Illegal empty parameter value
- Illegal parameter value length
- Illegal parameter numeric value
- Illegal parameter data type
- Illegal meta character in parameter value

This field displays the number of illegal flow to object violation occurrences.

- If you click the linked occurrence number, a View requested objects window appears containing a list of all the objects that caused this violation.
- If you click an object link, the View full request information window appears showing all the technical details of all the violations related to the specific request.



Illegal query-string or POST-data



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Flow

This is the name of the Application Flow (path) which defines the access path leading from one object to another object.

Check QS/PD

Check this box to instruct the TrafficShield security application to perform validity checks on the query string or POST data.

Occurrences

This number displays the number of requests that caused this violation.

Available actions for illegal query-string or POST-data

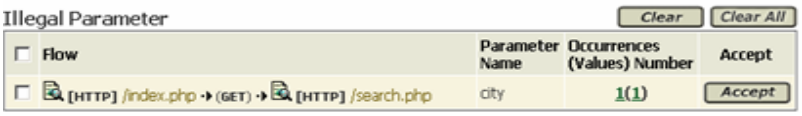
Accept

Clicking the Accept button adds the changes to the policy. Accept means that you have decided that the request reflects a real-life situation that warrants a change in the policy. It is possible to manually change the value of some of the parameters.

Clear

Clicking Clear deletes the entry from this learning window without changing the policy. A warning message appears asking to confirm the deletion.

Illegal parameter



The Illegal Parameter window lists the parameters that can appear in the request but are not defined for a specific flow.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Flow

This is the name of the Application Flow (path) which defines the access path leading from one object to another object.

Parameter Name

This is the name of the undefined parameter.

Occurrences (Values) Number

This field displays the number of illegal parameter occurrences.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

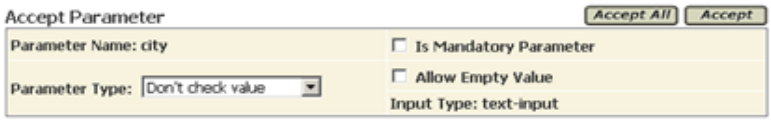
Available actions for Illegal parameter

Clear

Click Clear to clear the specific entry from the learning window without changing the policy

Accept

To accept the violation case and make it legal for future occurrences, click Accept. The Accept Parameter window appears.



Illegal static parameter value

This screen shows static parameters that carried a value not included in the value list defined in the policy.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Name

This is the name of the illegal static parameter value.

Parameter Flow

This is the name of the parameter flow (path) which defines the access path leading from one object to another object.

Occurrences (Values) Number

This field displays the number of illegal flow to object violation occurrences.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Available actions for illegal static parameter value

Accept
To accept the violation case and make it legal for future occurrences, click Accept. The Accept Parameter window appears.

Clear
Click Clear to clear the specific entry from the learning window without changing the policy

Clear All
Click Clear All to clear all the entries from the learning window without changing the policy.

Click the parameter name link to accept the violation, this will open a new window.

Illegal empty parameter value

This window displays the list of parameters that violated the not null value definition. (The field was empty when it should have contained a value.)

The decision whether a specific parameter can be left empty or not is dependent on the web application.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Name

Lists the parameters where Value Error was found.

Parameter Flow

This is the name of the parameter flow (path) which defines the access path leading from one object to another object.

Occurrences (Values) Number

This number displays the number of requests and values that caused this violation.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Available actions for illegal empty parameter value

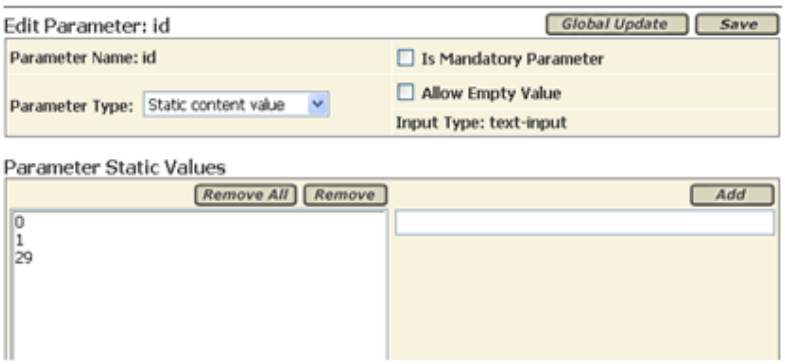
Clear

Click Clear to clear the specific entry from the learning window without changing the policy.

Clear All

Click Clear All to clear all the entries from the learning window without changing the policy.

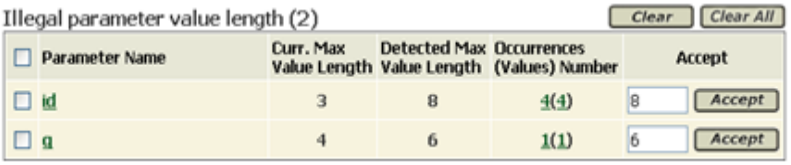
Click the parameter name link (id) at the top of the screen, this will open the following screen where you can edit the parameter value.



Illegal parameter value length

Note

This violation is relevant only for the Parameter Type: User Input Value.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Name

Lists the parameters where the illegal parameter Value length error occurred.

Current Max Value Length

The maximum length value permitted for this parameter.

Detected Max Value Length

The maximum length value permitted for this parameter

Occurrences (Values) Number

This number displays the number of requests and values that caused this violation.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

User Input

User Input fields allow the user to enter a valid value that overrides the defaults.

Available actions for illegal parameter value length

Accept

To accept the violation case and make it legal for future occurrences.

Clear

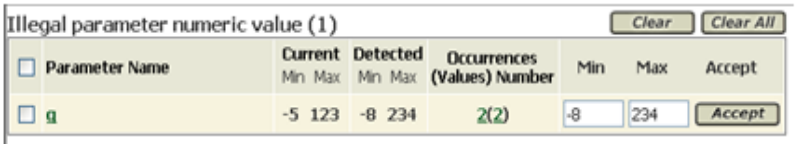
To clear the specific entry/entries from this learning window without changing the policy.

Clear All

To clear all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm the operation.

Illegal parameter numeric value

The Illegal Parameter Numeric Value screen lists the errors that may occur for the request parameters' values if the defined parameter is decimal or integer in the policy. This window provides statistical information regarding the types of parameter numeric value problems that have been detected.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Name

This is the name of the illegal parameter value length.

Current Min Max

The minimum and maximum numeric values permitted for this parameter.

Detected Min Max

The detected minimum and maximum numeric values detected in the violation.

Occurrences (Values) Number

This number displays the number of requests and values that caused this violation.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

User Input - Min

This is the minimum value received from all the request parameters with this violation type. It is possible to manually change the value of this field.

User Input - Max

This is the maximum value received from all the request parameters with this violation type. It is possible to manually change the value of this field.

Available actions for illegal parameter numeric value

Accept

To accept the violation case and make it legal for future occurrences.

Clear

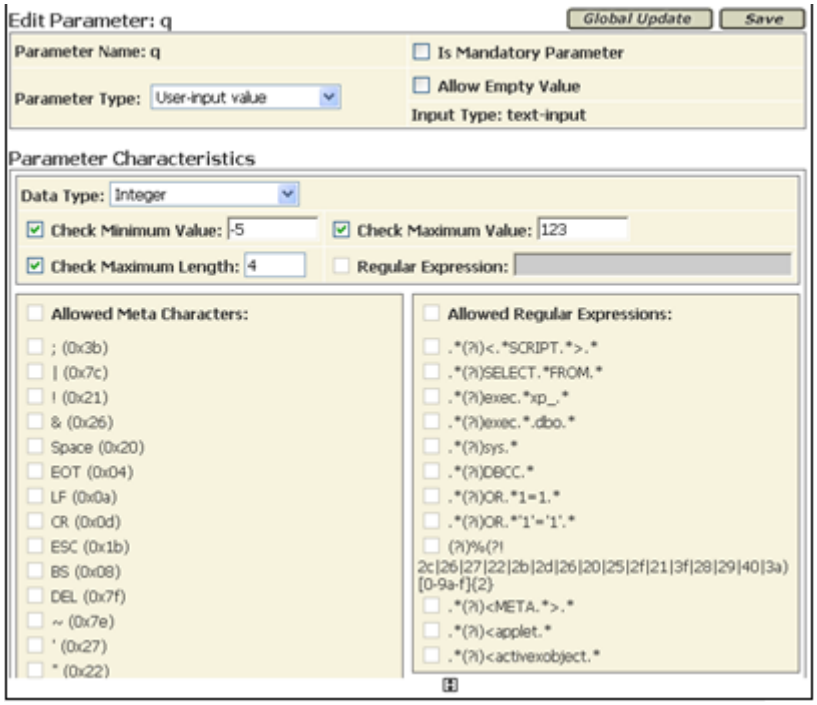
To clear the specific entry/entries from this learning window without changing the policy.

Clear All:

To clear all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm the operation.

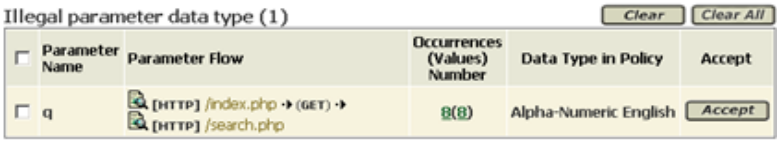
To open the Flow of parameter window.

Click the parameter name to open the Edit Parameter screen.



Illegal parameter data type

This screen shows parameters whose data type is different from the data type defined for them in the policy.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Name

Lists the parameters where the illegal parameter data length error occurred.

Parameter Flow

This is the flow where the parameter value error occurred.

Occurrences (Values) Number

This number displays the number of requests and values that caused this violation.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Data Type in Policy

This field displays the data type that was detected by the value error.

Available actions for illegal parameter data type

Accept

To accept the violation case and make it legal for future occurrences.

Clear

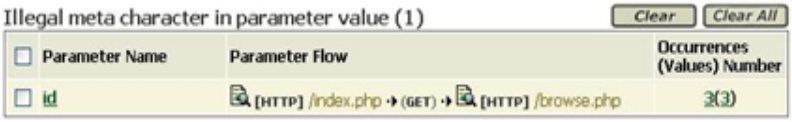
To clear the specific entry/entries from this learning window without changing the policy.

Clear All:

To clear all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm the operation.

Illegal meta character in parameter value

The parameter name contains a character that is set to "N" (false) or "C" (check) in the Administration > Character Sets > User Input: Defaults.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Name

Lists the parameters where the illegal character in parameter value error occurred.

Parameter Flow

This is the flow where the parameter value error occurred.

Occurrences (Values) Number

This number displays the number of requests and values that caused this violation.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Available actions for illegal meta character in parameter value

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Clear All

To delete all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm this.

Click the Parameter name to open the following screen:

The color legend at the top of the screen is applicable all through the Security Policy tool.

- Red - blocks the character.
- Blue - flags the character.
- Black - allows the character.

Available actions for editing illegal meta character in parameter value

Accept

To accept the violation case and make it legal for future occurrences.

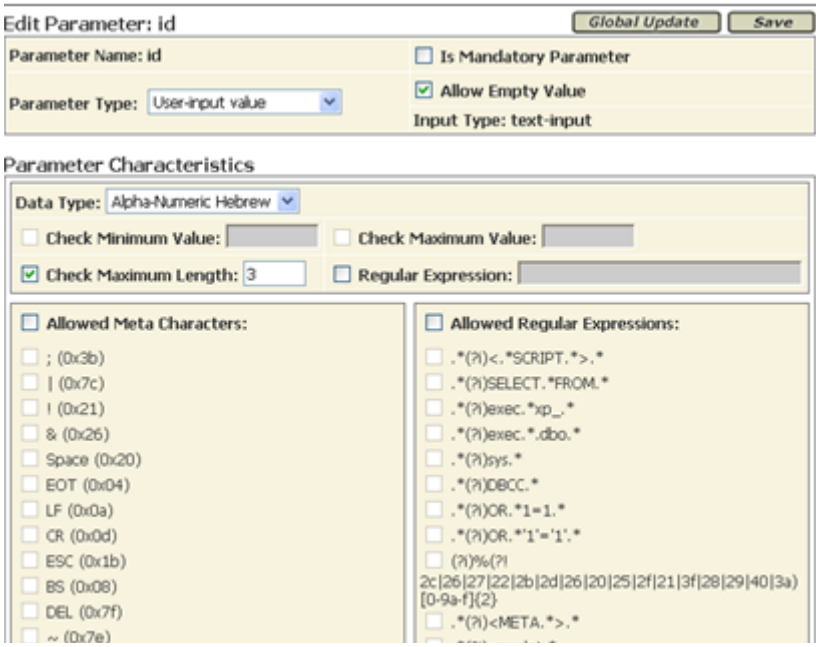
Clear

To clear the specific entry/entries from this learning window without changing the policy.

Clear All:

To clear all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm the operation.

Click the parameter name link to open the Edit Parameter window.



Edit the parameter as required.

Malicious parameter value

This is the parameter that contains a regular expression value that is not defined as an allowed regular expression for this parameter.

Checkboxes

The first column contains checkboxes used to mark the relevant entry

Parameter Name

Lists the parameters where the malicious parameter value error occurred.

Parameter Flow

This is the flow where the parameter value error occurred.

Occurrences (Values) Number

This number displays the number of requests and values that caused this violation.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Available actions for malicious parameter value

Clear

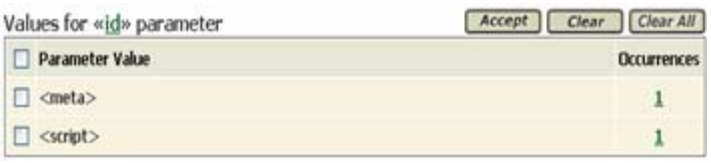
Click Clear to clear the specific entry/entries from this learning window without changing the policy.

Clear All

Click Clear All to delete all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm this.

To edit the parameter definitions:

Click the parameter name link, and the Values for the specific parameter window are displayed.



Parameter Value

The parameter value where the violation occurred

Occurrences

This number displays the number of requests that caused this violation.

Available actions for editing the malicious value parameter definition

Accept

To accept the violation case and make it legal for future occurrences.

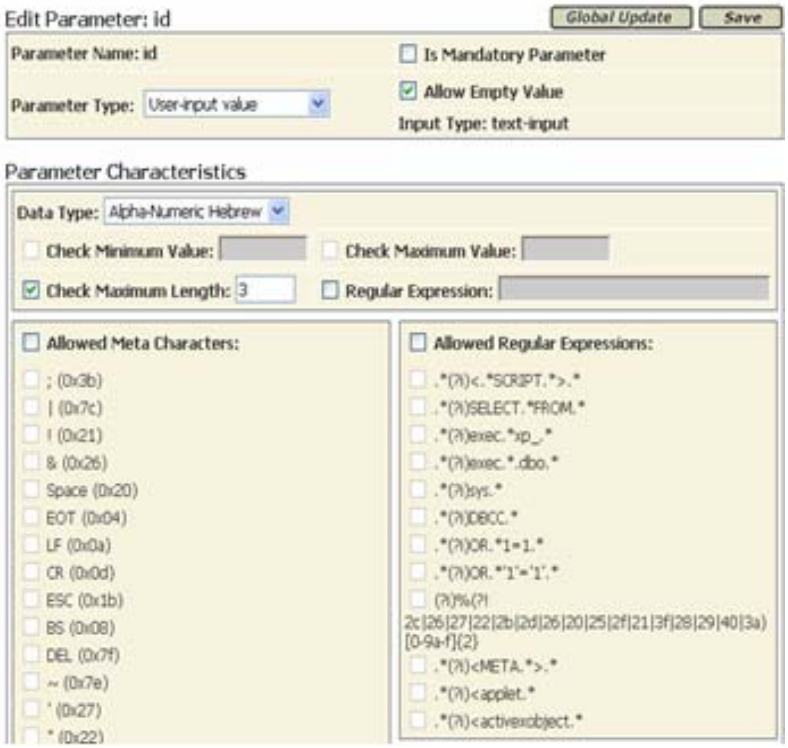
Clear

To clear the specific entry/entries from this learning window without changing the policy.

Clear All:

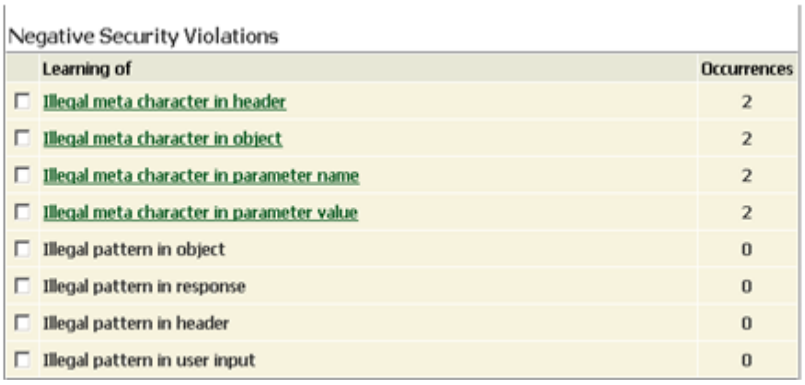
To clear all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm the operation.

To open the Edit parameter window, click on the specific parameter name.



Negative security violations

Negative Security Violations are linked whenever a character or regular expression that is not allowed in the TrafficShield security application default configuration lists is detected.



Negative Security Violations

	Learning of	Occurrences
☐	Illegal meta character in header	2
☐	Illegal meta character in object	2
☐	Illegal meta character in parameter name	2
☐	Illegal meta character in parameter value	2
☐	Illegal pattern in object	0
☐	Illegal pattern in response	0
☐	Illegal pattern in header	0
☐	Illegal pattern in user input	0

The Negative Security Violations are classified as follows:

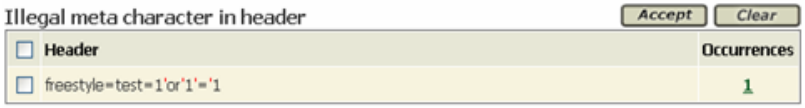
- Illegal meta character in header
- Illegal meta character in object
- Illegal meta character in parameter name
- Illegal meta character in parameter value
- Illegal pattern in object
- Illegal pattern in response
- Illegal pattern in header
- Illegal pattern in user input

Illegal meta character in header

This violation is detected whenever a meta character is detected in the Header.

The list of legal meta characters can be found in the Character Set tab of the configuration section.

Accepting a header that contains an illegal meta character or more modifies the Action for all the illegal meta characters found in the header from No to Yes in the Configuration > Character Sets > Header Charset list in the Policy Management tool.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Header

The Header name that has been detected as containing a meta character.

Metachars

The illegal meta characters that were detected.

Occurrences

This number displays the number of requests and values that caused this violation.

- If you click on the linked occurrence number, a **View requested objects** window appears containing a list of all the objects that caused this violation.
- If you click an object link, the **View full request information** window appears showing all the technical details of all the violations related to the specific request.

Accept

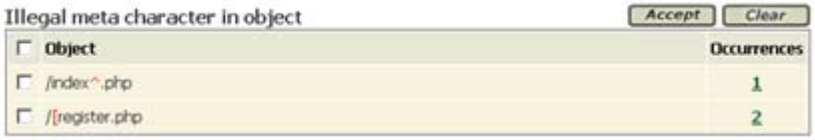
To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Illegal meta character in object

This violation is detected whenever a meta character is detected in the Object.



The list of legal meta characters can be found in the Character Set tab of the configuration section.

Accepting a header that contains an illegal meta character or more modifies the Action for all the illegal meta characters found in the header from NO to Yes in the Configuration > Character Sets > Object Charset list in the Policy Management tool.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Object

The object in which the illegal meta character was detected.

Occurrences

The number of occurrences of the violation.

Available actions for Illegal meta character in object

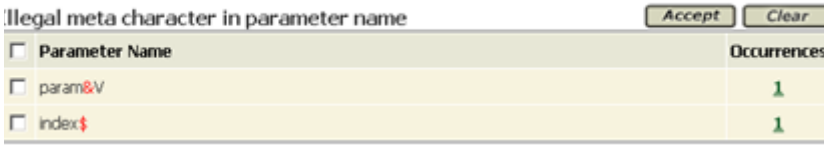
Accept

To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Illegal meta character in parameter name



This violation is detected whenever a meta character is detected in the Parameter name.

The list of meta characters can be found in the Character Set tab of the configuration section.

Accepting a header that contains an illegal meta character or more modifies the Action for all the illegal meta characters found in the header from NO to Yes in the Configuration **Character Sets > Parameter name list** in the Policy Management tool.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Name

The name of the parameter in which the illegal meta character was detected.

Occurrences

The number of occurrences of the violation.

Available actions for illegal meta character in object

Accept

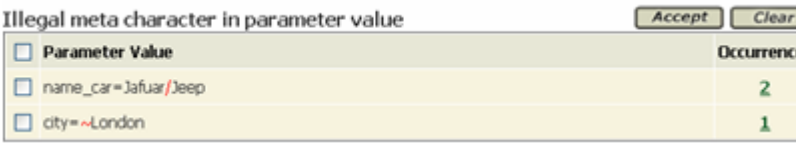
To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Illegal meta character in parameter value

This violation is detected whenever a meta character is detected in the parameter value.



The list of meta characters can be found in the Character Set tab of the Configuration section.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Value

The parameter value in which the error value occurred.

Available actions for illegal meta character in parameter value

Accept

To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Illegal pattern in object

This violation is displayed whenever an illegal pattern is detected in the Object.

Accepting an object that contains an illegal pattern or more deletes all the regular expressions found in the object in the Configuration > Negative RegExp Default list referring to objects in the Policy Management tool.

Checkbox

The first column contains checkboxes used to mark the relevant entry

Object

The name of the object in which the illegal pattern occurred.

Occurrences

The number of occurrences of the violation.

Available actions for illegal pattern in object

Accept

To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Illegal pattern in response

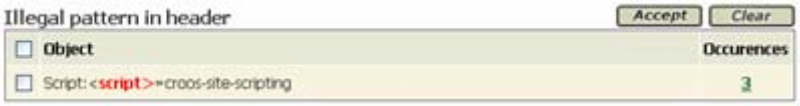
This violation is signaled whenever an illegal pattern is detected in the Response.

The list of legal patterns for the response can be found in the Configuration > Negative RegExp default list referring to responses in the Configuration section of the Policy Management tool.

Accepting a response that contains an illegal pattern or more deletes all the regular expression found for the responses from No to Yes in the Configuration > Negative RegExp default list referring to responses in the Policy Management tool.

Illegal pattern in header

This violation is signaled whenever an illegal pattern is detected in the Header.



The list of legal patterns that can be used in the Header can be found in the Configuration > Negative RegExp list in the Configuration section of the Policy Management tool.

Note

The Check Response should be set to true for relevant object types in Configuration > Object Types page in order to identify violations of this type.

Accepting a response that contains an illegal pattern or more deletes all the regular expressions found in the list for the Configuration > Negative RegExp default list in the Policy Management tool.

Checkboxes

The first column contains checkboxes used to mark the relevant entry

Object

The name of the header in which the illegal pattern was detected.

Occurrences

The number of occurrences of the violation.

Available actions for illegal pattern in header

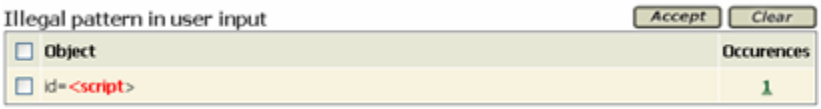
Accept

To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Illegal pattern in user input



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Object

The name of the user input in which the illegal pattern was detected.

Occurrences

The number of occurrences of the violation.

Available actions for Illegal pattern in user input

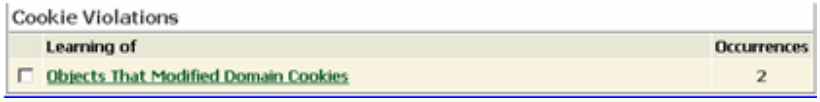
Accept

To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Cookie violations

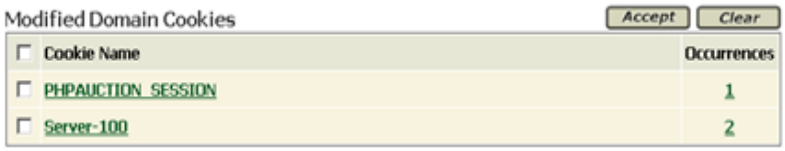


This category contains one section:

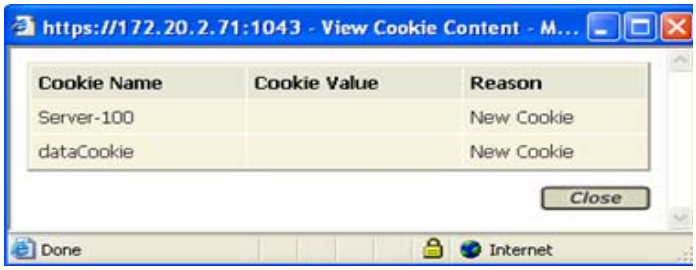
- Objects that modified domain cookies.

Modified domain cookies

This violation category is divided into two cookie violation sub-categories: **Modified Domain Cookies** and **Objects That Modified Domain Cookies**.



Click on the cookie name. The cookie content is displayed.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Cookie Name

This is the attribute part of the cookie name value pair (name=value).

Occurrences

The number of occurrences of the violation.

Available actions for Modified domain cookies

Accept

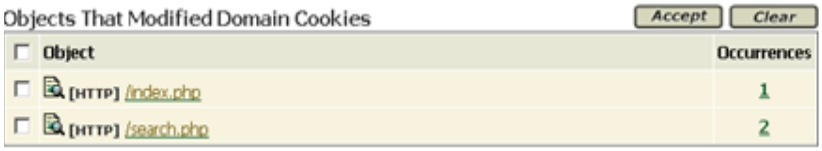
To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Objects that modified domain cookies

This screen lists the objects that modified domain cookies.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Object

This is the name of the Object that modified the Domain Cookie.

Occurrences

This is the number of times that the object modified the Domain Cookie.

Available actions for Objects that modified domain cookies

Accept

To accept the violation case and make it legal for future occurrences.

Clear

To clear the specific entry/entries from this learning window without changing the policy.

Click on the Object link

To view the cookie contents.

Forensics

This section explains how the user can review all the requests that caused at least one violation error. Each request is mapped to the learning tables, and the user can locate the full content of the specific request in order to do further investigation and to have a better understanding of the problem.

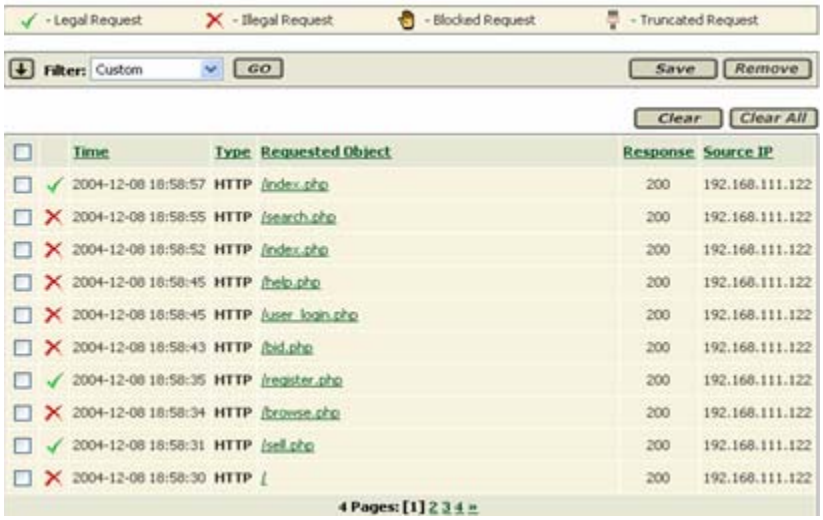
All the requests that violate the policy settings always go to the Illegal Requests table in the Forensics section. The other Forensic tables store deleted or legalized requests.

You can select multiple forensic entries using the Forensic filters tool located at the top of all the Forensics windows.

Illegal requests

You can view requests that contradict the policy in the Illegal Requests window. In addition, these requests are automatically categorized according to their content and registered in the appropriate Learning tables as well.

For example, a request for an illegal flow is registered in Forensics - Illegal Requests and also in Learning - Undefined Flows.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Blocked column

The second column may contain a red X which indicates that this request was blocked.

Time

This box displays the date and time of the request.

Type

This shows the protocol of the request (HTTP/HTTPS).

Requested Object

This field displays the requested URI.

◆ Note

Click a specific object to view the full contents of the request, and the View Full Request Information window is displayed.

Response

This field is the server HTTP response status.

Source IP

This is the IP address of the client machine that issued the request.

Available actions for illegal requests

Clear

Clicking Clear deletes the checked entries from this window without changing the policy. A confirmation window is displayed.

Clear All

Clicking Clear All deletes all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm this.

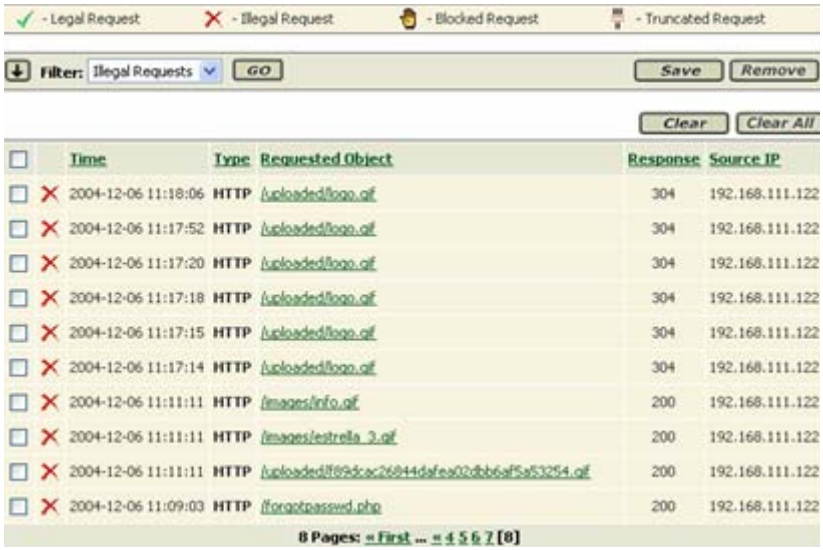
Available actions for length errors

To change the permanent ignore decision, check the checkbox next to the relevant item, and click the relevant Clear button.

The next time a new request causes a violation it will not be ignored, and will appear in the corresponding Learning windows, and the full request contents will be viewable in the Illegal Requests window.

Ignored requests

This section deals with requests that were actually illegal but could not be mapped into the illegal request tables since the Object Type, Object, or Flow match one of the Ignored Items entries.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Blocked column

The second column may contain a red X which indicates that this request was blocked.

Time

Date and Time of request

Type

Protocol of the request (HTTP/HTTPS)

Requested Object

This field displays the requested URI.

Note

Click a specific object to view the full contents of the request, and the View Full Request Information window is displayed.

Response

This field is the server HTTP response status.

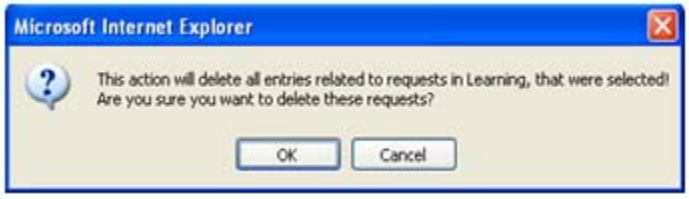
Source IP

This is the IP address of the client machine that issued the request.

Available actions for ignored requests

Clear

Clicking Clear deletes the checked entries from this learning window without changing the policy. Clicking Clear deletes the checked entries from this window without changing the policy. THe following confirmation window is displayed.



Clear All

Clicking Clear All deletes all entries from this learning window without changing the policy, regardless of whether they are checked or not. You will be asked to confirm this.

Viewing the Full Request Information window

You can access this window from both the Learning and the Forensic areas.

Following is an example of the type of information displayed when you choose to open this window.

Ignored items

This section explains the origin of the items listed in the Ignored Items window.

While using the Learning capabilities to fine-tune the policy object types, objects and flows may be either accepted or cleared. When a user chooses to clear any of the items in the above list, the user is asked whether he would like to "permanently reject the item from learning." This instructs the TrafficShield security application not to register duplicate identical requests in the Learning tables. The deleted request goes to the Forensics > Ignored Items screen.

View Full Requests Information window

All new requests containing an Object Type, an Object, or a Flow that match an entry in this window are ignored and do not appear in the Illegal Requests window.

The new ignored request is displayed in the Ignored Request window.

Request Violations

This is the list of all encountered violations created by the request.

Flags Requested Object

This is the object part of the request URI.

Response Code

This is the HTTP web server response status.

Full Request

This section displays the entire request including the HTTP headers and user input (Query String or post Data).

Referrer Objects

These are the referrer objects according to the policy.

Parameter Violations

This is the list of input violations per parameter in the request, where applicable.

Support ID

This is the unique identifier of the illegal request.

Cookie Name

This is the attribute part of the cookie name value pair (name=value).

Cookie Value

This is the value part of the cookie name value pair (name=value).

Reason

This is the reason that caused the violation.

Available actions for ignored items

Close

Close the window and return to the previous screen.

Accept

To accept the violation case and make it legal for future occurrences.

Run Auto-Accept

Run the Auto-Accept function.

Policy component editing

This section explains how to manually edit the policy components. The assumption is that the TrafficShield security policy has already been created by using a combination of tools: the Policy Browser, the Crawler, and the Learning tool.

We do not recommend to manually create a security policy from scratch, due to the enormous complexity of the task, although theoretically it is possible.

The Crawler builds a usable policy that checks all the objects and flows of the Web application. Manual intervention may be needed if you want to override the definitions generated by the Crawler. For example, you may remove an object from the policy if you do not want TrafficShield security application to check requests that refer to it or you can enter regular expressions to enhance the checks.

Most of the modifications made to a policy are typically done through the Learning tables. For example, you can add a missing object through a single click, once the Learning process has determined that the object should be part of the policy.

Refer to the beginning of this chapter for more details on the Learning process.

Adding Object types

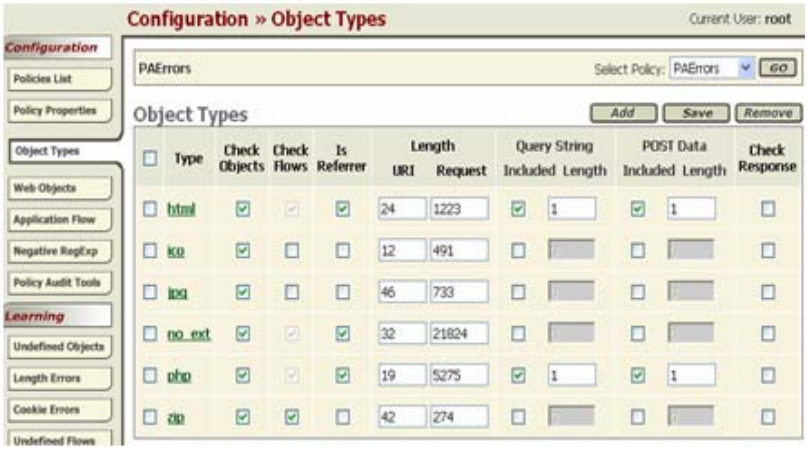
The Object Types tab lists the existing file types in the protected Web site. For example, a list of valid object types for a specific policy could be: **GIF**, **JPG** and HTML only. If your policy contains the above list, then any request for a **PDF** file would be considered illegal.

The extensions are listed here to enable you to decide how the policy should react to requests that refer to files that have these extensions.

Each entry in the table is composed from the object type, and the object type's set of flags and values. When adding a new object to the policy, this set of flag and values is the default settings applied to the object.

◆ Note

A special entry of "no_ext" file type is created in the object type table to handle the following cases: Objects with no file extension, and Objects with file extensions longer than 8 characters.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Type

This is the file extension. Clicking on the object type link leads you to a list of Web objects of this type.

◆ Note

The Type field is case-sensitive; for example you can add both html and HTML and they will be treated as different object types.

Check Objects

If this checkbox next to an object is selected, TrafficShield security application checks requests for this object type to verify that the actual object exists in the Web application or is accessible via the application flow.

If this checkbox is not selected, TrafficShield security application lets through requests for this object type without checking whether the actual object exists in the Web application or is accessible via the application flow.

◆ Tip

If the Web application changes frequently,(i.e., a set of objects in the Web application are changed frequently) it is not a good idea to clear this box, in order to avoid massive warnings and rejections. We recommended that you read the Allowed Objects RegExp - Object list relaxation section to learn how to define a less strict set of Web application objects.

Check Flows

The Application Flow (path) is the defined access path leading from one object to another object.

Check this box to instruct the TrafficShield security application to test whether the requested object from a given object type is a legal flow.

For example, a list of valid flows would be:

- From abc.html to abc.gif, OK
- From abc.html to def.html, OK

If your policy contains the above list, then any request that tries to access abc.gif from def.html would be considered illegal.

If you clear the box, any request accessing this file is considered as legal, even if it did not originate from a legal flow.

Application flow model

TrafficShield security application maps all the possible user actions in a web application, including parameter and values. Any non-recognized action can then be considered an attack, and blocked.

Is Referrer

Check this box if objects of this object type may refer to other files. For example, HTML pages containing a link or CGI files calling another file, are referrers. Pictures and sound files cannot be referrers because these objects never contain links to other objects and are not web pages.

Length URI

This field defines the maximum legal length of the object's full path for this object type.

Length Request

This field defines the maximum legal length of the entire request.

Query String Included

Check this checkbox if requests for objects of this object type may include user input in the query string part of the request. A query string requests data in the format …abc.html?Name=John.

◆ Tip

If the query string is empty, i.e., nothing is written after the question mark, the TrafficShield security application considers the request as an empty query string.

Query String Length

This field defines the maximum legal length of the user input in the query string part of the request. For example: In the following request, abc.html?Name=John&X=2, the actual query string length is 13 (Name=John&X=2).

Post Data Included

Check this checkbox if requests for objects of this object type may include user input in the post data part of the request.

Post Data Length

This field defines the maximum legal length of the post request user input data.

Check-Response

Check this checkbox to activate Server response filtering by the TrafficShield security application. If checked, the html body of the response will be tested vs. the Negative Regular expression applied to the Server response. See the Negative RegExp section in this chapter.

To add an object type manually:

If the Web application includes objects of a type not listed here, you can add them manually.

1. In Object Types, click the Add button.
   The Add Object Type popup window opens.

2. Enter the file extension and click OK. (Type the extension without the period that appears in front of the extension.)
3. In the Object Types page, review the flags and values and set the policy for this object type, as explained above.
4. To save the changes, check the left checkbox next to the relevant entries and click the Save button.

◆ Note

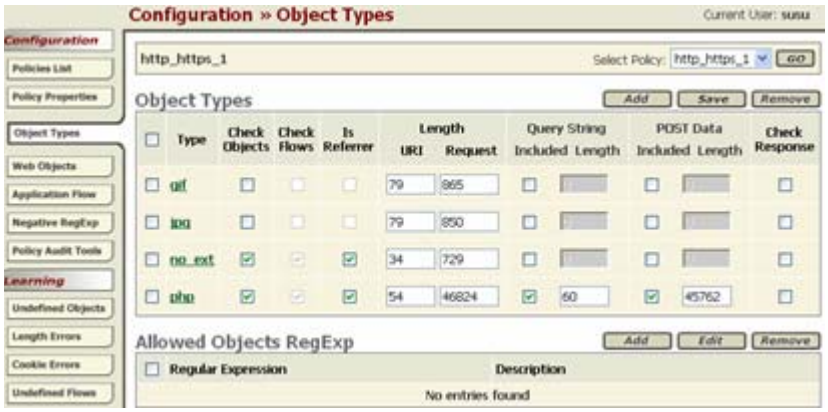
In order to remove an object type, check the left checkbox next to the relevant entries and click the Remove button. All existing objects of this object type and all relevant flows and parameters will be removed from the policy.

Allowed objects RegExp - Object list relaxation

The object list for a specific object type is enforced by the TrafficShield security application. If the Check Object flag is set for a specific object, any request containing an object that is not on the list will create a "non-existent object" violation.

This section explains how to lessen this severe restriction for a specific object type.

This situation is inconvenient if the Web application is dynamic and the set of objects of a given object type changes frequently. Adding and editing the object list manually or via the Learning process may become a complicated and endless task.



To resolve this problem, it is possible to define regular expressions describing the set of possible objects.

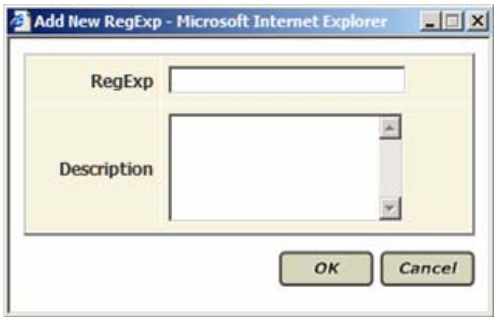
To define expressions as a set of possible objects:

In the Allowed Objects RegExp section (located at the bottom of the Object Types window) follow these steps:

1. Set "check objects" to true.
2. Define regular expression(s) describing the set of possible objects as explained below.

To add a regular expression:

1. Click the Add button.
   The Add New RegExp dialog box opens.



2. In RegExp, enter the expression. For example, if the policy contains objects a.gif and b.gif only, the regular expression *\.gif$ will allow any object of a gif object type.
3. Click OK.

Defining Web objects as entry points

After reviewing the object types, you can examine each object separately and fine-tune the security attributes for each of them.

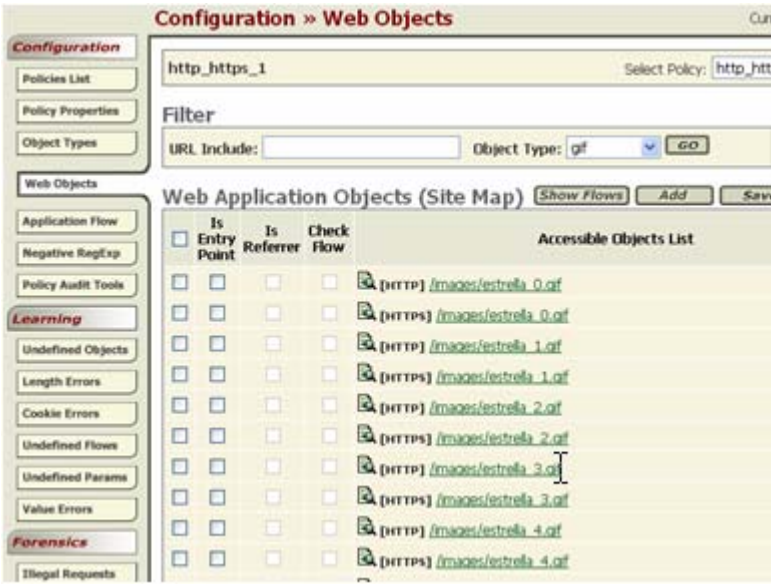
An important policy decision to make at this stage is to decide whether a certain object is an entry point or not.

An entry point is a page through which a visitor should enter the Web application as designed by the Web Master of the application; for example, by typing its URL in the browser's address box, or by selecting its URL from a favorites list.

Your Web application may have several entry points. By defining objects that are entry points, you prevent an attacker from entering your Web application without passing through the "front door."

To access the object list relating to a specific object type

- Choose Policy Management > Configuration > Web Objects. Choose the relevant object type in the drop down menu, and click the GO button. The list of objects responding to your choice are displayed.



URL Include (Filter bar)

Use this field to view a subset of the object list. For example; type a string to list all the objects containing this string.

Note

Each object in the list has a prefix which indicates the protocol (HTTP/HTTPS) through which this object may be requested. This may cause the same object to be displayed twice in the object list if relevant to both protocols.

Tip

This search is case-sensitive.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Is Entry Point

The Crawler defines some objects as entry points during its run. These objects are likely to be bookmarks or they were pre-defined as entry points in the Policy Management > Policy Properties > Crawler-Settings > File Types Associations. We recommend that you review these entry point definitions.

Is Referrer

Check this box if this object may refer to other objects. For example, HTML pages containing a link or CGI files calling another file, are referrers. Pictures and sound files cannot be referrers because these objects never contain links to other objects, and are not web pages.

Check Flow

The Application Flow (path) is the defined access path leading from one object to another.

Check this box to instruct the TrafficShield security application to test whether the requested object is a legal flow.
For example, a list of valid flows would be:

- From abc.html to abc.gif, OK
- From abc.html to def.html, OK

If your policy contains the above list, then any request that tries to access abc.gif from def.html would be considered illegal.
If you clear the box, any request accessing this file will be considered as legal even if it did not originate from a legal flow.

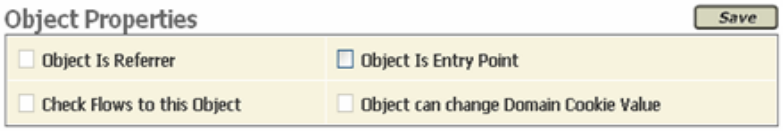
Accessible Objects List

Object list that answers the filter criteria. To open the Object Properties Window for a specific object in the list, click the object link. This window is divided into three parts:

- Object Properties
- Flows to Object
- Dynamic Flows from Object

Object properties

This section defines the object flags as displayed in the upper level, Web Objects tab.



Object is Referrer

Check this box if this object may refer to other objects. For example, HTML pages containing a link or CGI files calling another file, are referrers. Pictures and sound files cannot be referrers because these objects never contain links to other objects, and are not web pages.

Object is Entry Point

The Crawler defines some objects as entry points during its run. These objects are likely to be bookmarks, or they were pre-defined as entry points in the Policy Management > Policy Properties > Crawler-Settings > File Types Associations. We recommend that you review these entry point definitions.

Check Flows to this Object

The Application Flow (path) is the defined access path leading from one object to another.

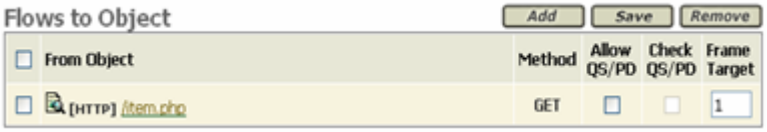
Check this box to instruct the TrafficShield security application to test whether the requested object is a legal flow.

Object can change Domain z Value

If the object is a referrer, then this box can be checked. If the domain cookie was changed on the client side (i.e., Java script function execution by browser), then the TrafficShield security application will fail any request if this checkbox is not checked for this object and the object is a referrer in the incoming request.

Flows to object

This section summarizes the flows to the object.



Checkboxes

The first column contains checkboxes used to mark the relevant entry.

From Object

This column lists the objects from which the object could be accessed.

Note

Click the object link to view the flow properties.

Method

This column specifies the method through which the object should be accessed.

Allow QS/PD

Check this checkbox to define whether user input is allowed.

Check QS/PD

If user input was allowed, then check this checkbox to enforce user input validations.

Frame Target

This is the index of the HTML frame targeted by the flow. We do not recommend that you change this value unless you know that you want to specifically load this object into a specific frame.

Note

The value 99 is a default frame index which means that the target object is loaded into the same frame as where the referrer object is presented.

Dynamic flows from object

Some flows cannot be foreseen because they involve a constantly changing set of objects. For example: a zone of the application where various users store files that external wizards can access, involves unpredictable flows if the users remove or add files daily.

In such cases, you can use the Dynamic Flows from Object section to legalize access to the changing sets of files.

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Prefix

This field is a fixed substring of the html source page. It may be a name of a section in combination with html tags; for example: "<h3 >Flows2Object</h3 >".

RegExp Value

This field defines a set of objects in the above mentioned dynamic group.

Suffix

The suffix is similar to the prefix. For example: <form name="dynamic_flows" >.

Note

The Prefix and Suffix instruct the TrafficShield security application of the boundaries that enclose the set of dynamic object links in a page. The TrafficShield security application uses the RegExp value as a pattern evaluate each object in the set between the boundaries.

Displaying web application objects

To show the objects' flows:

1. Click the select checkbox for the objects you want details on.
2. Click the Show Flows button to display a list of flows in the Flow List Window for the checked objects.

   The Flow List window displays the list of checked objects. Each object can be expanded to display the outgoing flows. For more details please see the following section on Application flow.

.

Adding a Web object

To add an object manually:

1. If you want to manually add an object without running the Crawler again, click the Add button and the Add New Object window opens.

2. In the Object Path field, enter the full resource path starting with the slash [/].
3. In the Protocol field, specify the protocol to be used to access the object.
4. In the Web Objects tab, review and edit the flags and values for the new object.
5. Check the modified entry's checkbox, and click the **Save** button.

Removing a Web object

To remove an object:

1. In the Web Application Objects list, check the relevant objects to be removed.
2. Then click the Remove button. You will be asked to confirm the removal.

Application flow

The Application Flow is the defined access path leading from one object to another object.

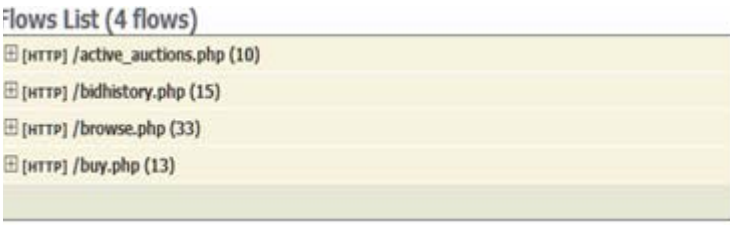
These flows are populated from various sources: The Crawler generates a map of the flows from within the Web application, by scanning the links and references within the objects. The Learning process results in acceptance of new flows. It is also possible to manually add and edit application flows.

To access the application flow

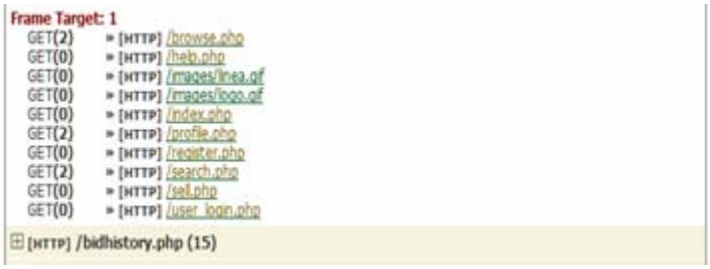
The Application Flow can be accessed in any of the following three ways:

1. Choose Policy Management > Configuration > Web Objects tab

2. Then click the desired object's URL link. The Flows to object section of the page, displayed now, lists the objects from which the selected file can be reached.
3. Click the "From Object" link to display the Application Flow window.
4. Choose Policy Management > Configuration > Web Objects tab
5. Then check the checkbox to the left of the relevant object (you can check more than one, if you want) and click the Show Flows button. This displays, at first, a list of the objects you have just marked.



6. Click the + button to see a list of the actual files that can be reached from the object you selected originally. If the reference targets a frame in a frameset, then the index of the target frame appears at the top of the referenced files.

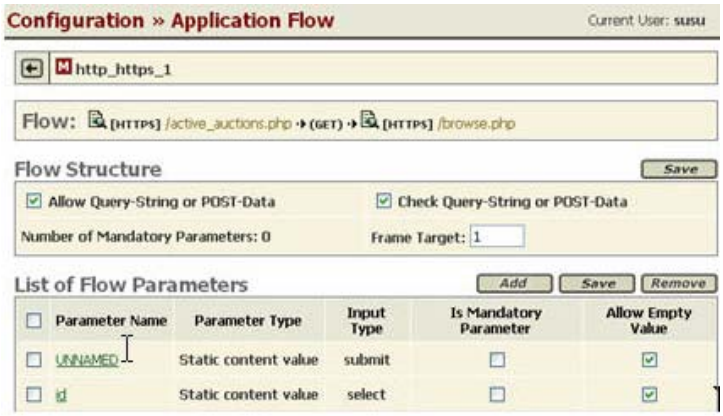


7. Click the "To Object" link to display the Flow window.

   Destination Objects are listed under the Frame Target Index into which they should be loaded by the application. Each entry specifies:

   - The method used to access the target object.
   - The number of known input parameters in ().
   - A protocol to request the target object.
   - Colorization of the targeted objects is used to differentiate between the Is Referrer flag settings (Brown=flag set to true, Green=flag set to false).

8. Click the Application Flow tab.
   You see a list of all flows.



The TrafficShield security application allows the user to view and edit the Query String and the POST Data. The flow parameters configuration is only accessible from these windows.

Flow structure

Allow Query-String or POST-Data

Check this box if a request that accesses the selected object via this specific flow may also carry a query string or POST data.

Check Query-String or POST-Data

Check this box to instruct TrafficShield security application to perform validity checks on the query string and the POST data. This relevant only if you already checked the Allow Query-String or POST-Data checkbox.

Number of Mandatory Parameters:

This number represents the number of parameters that must pass from the source to the destination object in this flow. This counter is updated automatically as additional parameters are marked as mandatory.

Frame Target:

This is the index of the HTML frame targeted by the flow. We do not recommend that you change this value unless you know that you want to specifically load this object into a specific frame.

◆ Note

The value 99 is a default frame index, which means that the target object is loaded into the same frame as where the referrer object is presented.

List of flow parameters

Checkboxes

The first column contains checkboxes used to mark the relevant entry.

Parameter Name

This column displays a list of the flow parameters.

Note

The Parameter Name "UNNAMED" is used for actual parameters on the flow that don't have a name.

Parameter Type

This field specifies the parameter type. See the parameter section below for details on the parameter types.

Input Type

This field defines the html input type of the parameter as it appears in the html source page.

Is Mandatory Parameter

Check this checkbox if this parameter must appear in the flow.

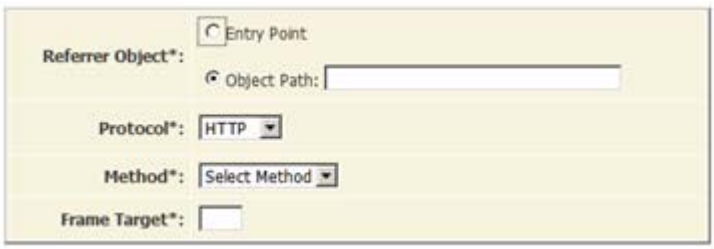
Allow Empty Value

Check this checkbox to allow the parameter to contain an empty value.

To manually add a flow:

This section explains how to add a new Application flow. Click OK after entering the new flow's information and click the Save button to save your changes.

1. Choose the Policy Management > Configuration > Web Objects tab.
2. Check the relevant object to which you want to add a new flow definition.
3. Click the Add button.
   The Add New Flow window opens:



Referrer Object

There are two possible referrer object types:

Entry Point

Choose this option if the object to which the flow should be added is an entry point.

Object Path

Choose this option and specify the referrer object path from which the target object should be accessed.

Protocol

Specify the protocol type by which the target object should be accessed.

Method

Choose the method by which the target object should be accessed.

Frame Target

This is the index of the HTML frame targeted by the flow. We do not recommend that you change this value unless you know that you want to specifically load this object into a specific frame.

◆ Note

The value 99 is a default frame index which means that the target object is loaded into the same frame as where the referrer object is presented.

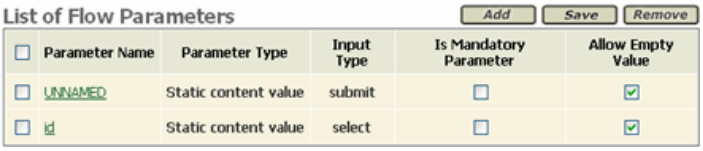
◆ Tip

In order to decide what to enter to the frame target index field, the html source page should be reviewed for frame set tags.

Defining the Flow parameters

This section describes the parameter properties and its configuration.

1. To access this window, choose the Policy Management > Configuration > Web Objects tab.
2. In the Web Objects window, choose the "target object."
3. From the list of Flows to Object, choose the "from object."
4. The Application Flow window appears and displays a List of Flow Parameters.



Checkboxes

The first column contains checkboxes used to mark the relevant entry

Parameter Name

Specify the name of the parameter as it appears in the request.

To view and edit the parameter properties, click the Parameter Name link. The Edit Parameter window appears.

Parameter Type

This field specifies the parameter type.

Input Type

This field defines the html input type of the parameter as it appears in the html source page.

Is Mandatory Parameter

Check this checkbox if this parameter must appear in the flow.

Allow Empty Value

Check this checkbox to allow the parameter to contain an empty value.

To add a new parameter to the flow:

1. Click the Add button in the List of Flow Parameters section in the Web Application tab.

◆ Note

The window contains two sections. In the top section, Add Parameter, the Parameter's general information is entered. The selected parameter type automatically changes the appearance and content of the bottom section. For example if you choose to add a parameter of a "static content value" type, the bottom section will display the Parameter Static Values screen.

Optional parameter types

Don't Check Value

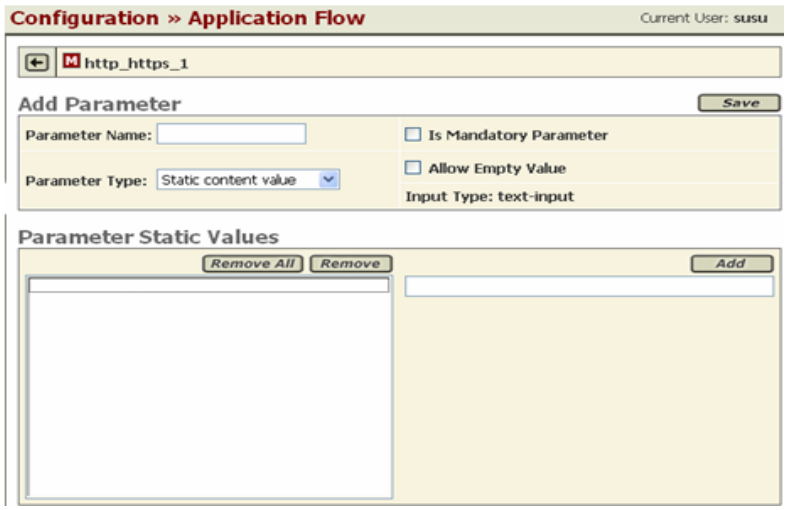
Select this option if you do not want TrafficShield security application to check the parameter value at all. If you choose this option, no bottom section appears in the window.

Note

A parameter defined as Don't Check Value must have a value in the request. The TrafficShield security application will not check its validity, but it will check its existence. To disable this functionality, check the Allow Empty Value box: this makes sure that empty parameters are also allowed.

Static Content Value

Select this option if users must select the value from a pre-defined list of values such as values found in a drop-down list or a list of values accessed via radio buttons. When this option is selected, the Parameter Static Values section appears.



To build a list of pre-defined values

1. In the box next to the Add button, enter a value.
2. Click Add. The value moves to the larger box.
3. Repeat this step to define all the values needed.

To remove a value from the list

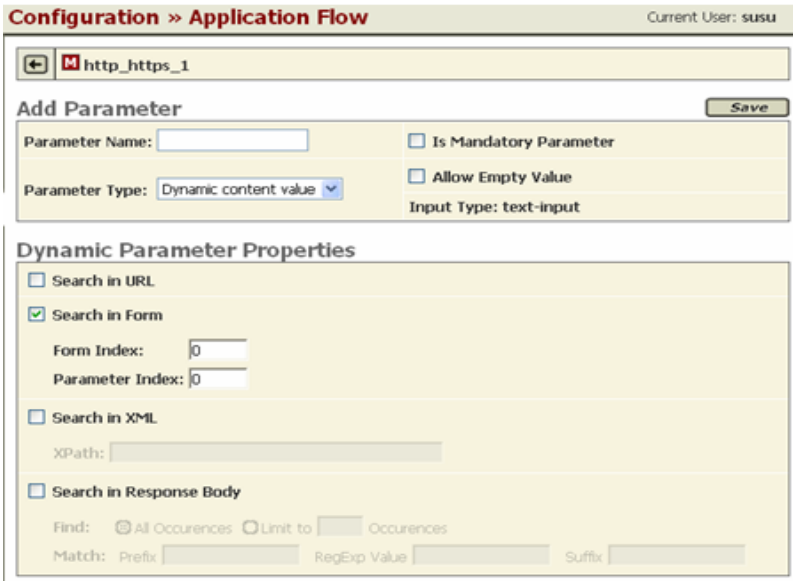
- Select the value and click the Remove button.
  The Remove All button clears the entire list.

Note

If the value list is empty for this parameter type, an illegal static parameter value violation is issued for any value received in this parameter in the request.

Dynamic content value

Use this option if the parameter value changes dynamically and the location of the value in the request cannot be foreseen. In this case, you instruct the TrafficShield security application to actually search for the value in the various sections of the request.



Enter the following information (you can run the search in one or more of the sections described below).

Search In URL

Check this box to instruct TrafficShield security application to search for the parameter value in the URL section of the request.

Search in Form

Check this box to instruct TrafficShield security application to search for the parameter value in one of the forms.

- In Form Index, specify the HTML index of the form that contains the parameter.

- In Parameter Index, specify the HTML index of the input parameter in the form that contains it.

Search in XML

Check this box to instruct TrafficShield security application to search for the parameter value in an XML block included in the request.

In the XPath box, specify the XML tag path (e.g., <products> <productPrices > <productSalesPrice >) where to look for the value.

Search in Response Body

Check this box to instruct the TrafficShield security application to search for the parameter value between two specific strings in the body of the request.

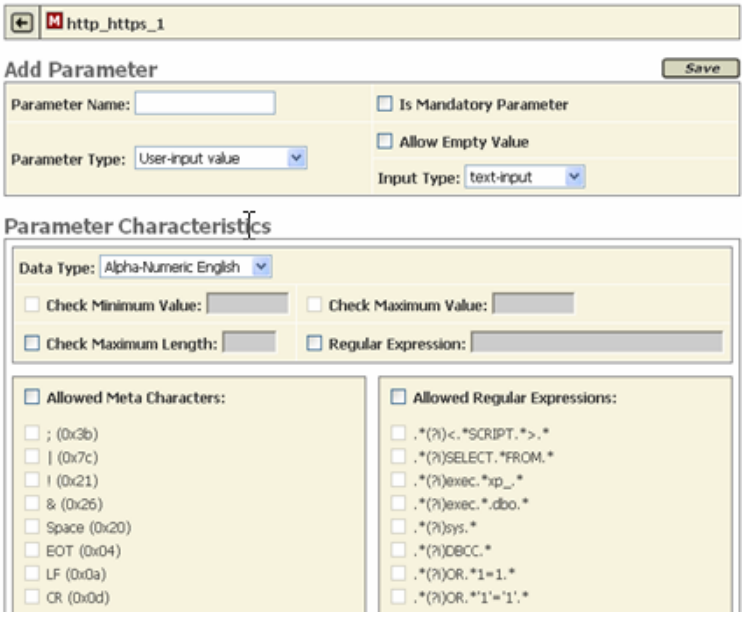
Enter the following information:

Item	Description
Find:	
All Occurrences	Select this option to search for all occurrences of the value.
Limit to… Occurrences	Select this option to search fore the first x occurrences of the value. Specify the number of occurrences to find.
Match	
Prefix	Enter the string that constitutes the starting point of the search in the request body.
RegExpValue	Enter a regular expression that describes the searched value (and parameter name, if necessary).
Suffix	Enter the string that constitutes the ending point of the search in the request body.

Parameter characteristics user input values

Select this option if the parameter accepts input from the user. For example it may be applied to html text area, input box, etc.

This option allows you to set the value's data type and to define the characters it may contain.



Data Type

Select the type of the parameter value. By selecting a type, you instruct the TrafficShield security application to consider as invalid any requests that contain data of a different type for this parameter.

Select	To limit the value to
Alpha-Numeric (language)	Any text consisting of letters, digits and the underscore character.
Integer	Whole numbers only (no decimals).
Decimal	Numbers only (including decimals).
E-mail	Text in e-mail address format only.
Phone	Text in telephone number format only.

Select the "Don't check" option if you do not want the TrafficShield security application to check the type of the parameter value.

Check Minimum Value

For numeric parameters of Integer/Decimal types, you can set a minimum value. A request that passes a parameter with a lower value is then considered illegal.

To set the minimum value, check the box and enter the value.

Check Maximum Value

For numeric parameters of Integer/Decimal types, you can set a maximum value. A request that passes a parameter with a higher value is then considered illegal.

To set the maximum value, check the box and enter the value.

Check Maximum Length

This attribute applies to all data types except the Don't check parameter type.

By setting a maximum length for parameters, you prevent unauthorized access via parameter values that have an unexpected length. For example, you can limit the length of an alpha-numeric value to 4 (characters) if it is never expected to contain more than 4 letters, and thus instruct the TrafficShield security application to consider as illegal any requests that contain a longer value.

To set the maximum length, check the checkbox and enter the maximum number of characters the value may contain.

Regular Expression

If the value is non-numeric, you can calculate it via a Regular Expression. To do so, check this checkbox and type the expression in the adjacent field.

This is a positive regular expression that defines what is legal.

Allowed Meta Characters

Use this section for characters defined as C (check) in the Character Sets table > Parameter values in the Administration tool. The TrafficShield security application will let through requests whose user input includes the characters marked here as valid. That is, C will be treated as Y (true). Please refer to the Installation and Configuration Manual for more details on Character Sets.

Allowed regular expressions

This is a list of regular expression designed to protect Web applications from common attacks via user input, like XSS, SQL injections, etc.

The user may allow a specific RegExp if normal input of the parameter is expected to contain a value that matches the RegExp.

Defining negative regular expression

The Negative Regular Expression tab contains a list of default and user-defined regular expressions. These regular expressions are meant to complete the security policy definitions.

The request/response content that matches at least one negative regular expression should be dropped.

Each regular expression may be modified to apply to one of the following parts of the request/response:

- Request URI
- Request key value pairs
- Request header values
- Server Response data (html body)

Character sets

The TrafficShield security application can be set to allow certain characters to appear in certain sections of a request. For example, you can allow letters, digits and the slash (/) in a path to an object but exclude the "@" character from it. Such exclusion causes TrafficShield security application to apply the Alarm/Blocking policy to the request that contains the excluded character.

Character sets can be defined for header values, object paths and user input (key value pairs).

For example, a path to an object may include the "/"character but not the name of a parameter. Therefore, a set should be defined for paths, which allows the "/" character, and another set should be defined for parameters, which excludes the "/" character.

In addition, you can define the valid character set for the data expected to be entered by the Web application users in a supported language. For example, if your application contains a form where users can type information in French, you can determine which characters are allowed when entering information in French; data entered in a form that contains characters not included in the French character set, as you have defined it, will activate the Alarm/Blocking mechanism.

Although the TrafficShield Application Firewall is shipped with default character sets for each such element, you can change them if you want. This section shows you how to enter such changes. When building a policy, you can further fine-tune the character set for input languages.

Note

The Character Sets are individual for each policy

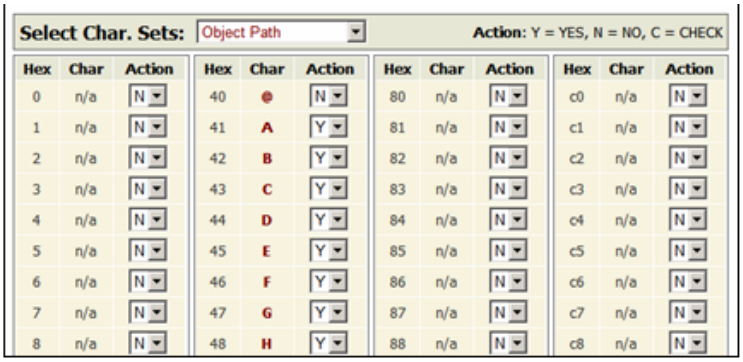
To build character sets

1. Click Policy Management > Configuration.
2. Click the Character Sets tab.
3. In Select Char. Set list, open the list and select the application element or input language for which you want to define a valid character set.

   The options are:

Option	Allows you to determine the characters allowed in
Object charset	The name of the web object.
Param Name	Parameter names.
HTTP Headers	The header section of an HTTP request.
Language names	User input in a specific language. For example, if your Web application supports French and you select User Input: French, data typed in by Web application users in form fields is verified against the French character set.

4. After selecting an option, TrafficShield security application displays an entire character set.



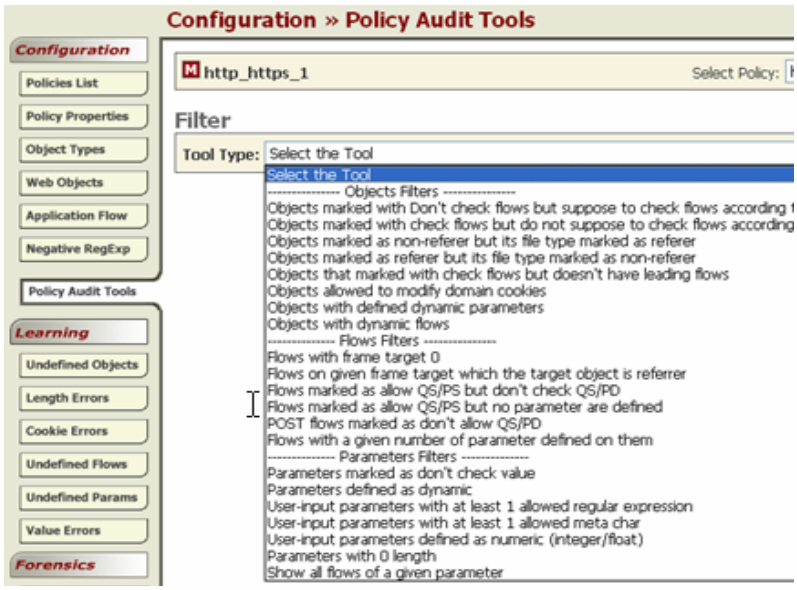
5. In the Action field of each character, select one of the following:

Action	Means
N	No. The character is invalid. An incoming request that contains this character will be blocked.
Y	Yes. The character is valid. An incoming request that contains this character will be let through.
C	Check, is equal to N, unless its explicitly defined as allowed in the Parameter Characteristics table under Application Flow (Policy Management tool). If the character is allowed there, then the request is valid. *C is not available for Header charset, Object charset, Parameter Name charset.*

6. To restore to default character set definitions, click **Restore Defaults** button.
7. Click **Save** to save the settings.

Policy audit tools

The Policy Audit tools analyze suspicious policy states. For example: Object without flows, Parameters with zero length, etc. Each report isolates a pre-defined state and assists the user in identifying conflicts & errors in the policy.



8

Monitoring

- Monitoring tools
- System monitoring area
- Security
- Reports on illegal requests
- Activity

Monitoring tools

Monitoring tools allow the network and policy administrators to monitor request traffic. This chapter explains how to use the TrafficShield security application monitoring tools to follow up on potential attacks and workload.

The monitoring tools described in this chapter are designed to help network and policy administrators examine both legal and potentially malicious traffic. The data collected by the Monitoring tool helps you identify overloaded units and make the necessary decisions on needed deployment changes.

All of the events tracked in Monitoring can also be captured in SNMP traps and exported to Syslog files. In addition, all the reports generated can be exported as HTML or PDF files. Contact your F5 Networks account representative for more details on these features.

To access the monitoring functions, click the Monitoring tab at the top of the TrafficShield security application.

This tool is divided into four areas which are explained in detail in this chapter:

- **System Monitoring area** monitors the TrafficShield security application units and their system status, for example; whether the unit is active or in standby mode. System logs can also be monitored from here.
- **Security Monitoring area** monitors the ongoing security statuses and events that occur on the TrafficShield security application units.
- **Reports area** generates reports and graphs on the ongoing attacks that have occurred on the TrafficShield security application units.
- **User Monitoring area** monitors the authorized users' activities on the TrafficShield security application units.

The filtering tools allow you to retrieve and focus on a set of events of particular interest to you. For example, you can focus on events that took place in the last hour, or events that involve requests that contained a specific text string.

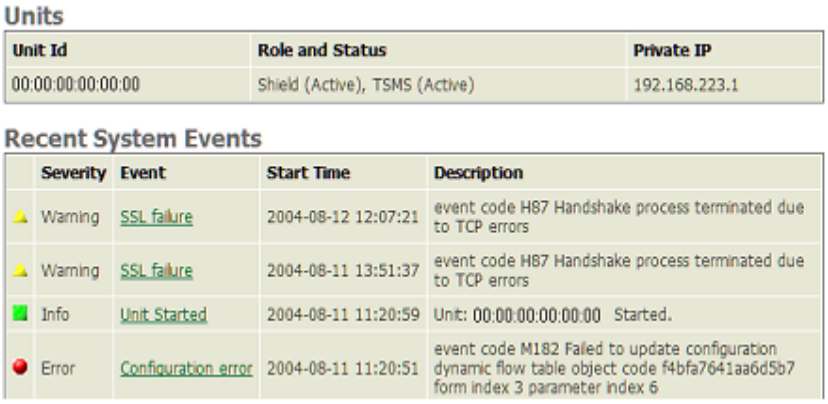
TrafficShield security application provides two filtering tools:

- The extensive filter
- The simple filter.

System monitoring area

Displaying the system status

Choose **Monitoring > System > Status** to open the Unit window and the Recent System Events window.



Units

Unit Id	Role and Status	Private IP
00:00:00:00:00:00	Shield (Active), TSMS (Active)	192.168.223.1

Recent System Events

	Severity	Event	Start Time	Description
⚠	Warning	SSL failure	2004-08-12 12:07:21	event code H87 Handshake process terminated due to TCP errors
⚠	Warning	SSL failure	2004-08-11 13:51:37	event code H87 Handshake process terminated due to TCP errors
■	Info	Unit Started	2004-08-11 11:20:59	Unit: 00:00:00:00:00:00 Started.
●	Error	Configuration error	2004-08-11 11:20:51	event code M182 Failed to update configuration dynamic flow table object code f4bfa7641aa6d5b7 form index 3 parameter index 6

Displaying the TrafficShield units' status

This window displays the current status of all the TrafficShield Units.

Unit Id

This is the MAC address of the relevant unit.

Role and Status

There are three possible roles:

Shield - This tool is responsible for blocking requests that violated the security definitions and alerting the user.

TSMS - TrafficShield Management Station, this tool is responsible for monitoring, configuring and managing the TrafficShield components and Graphical User Interface.

TSMS Backup - indicates whether the Hot Backup unit is active.

Possible statuses:

Active, None, Starting

Private IP

The unique IP address assigned to the TrafficShield security application unit.

Displaying the recent system events

The Recent System Events section lists the latest events that took place at the operating system level in the units or in the management station. The report can also refer to operating system events posted to the system log. Clicking an event displays more information about it in the Event Description box.

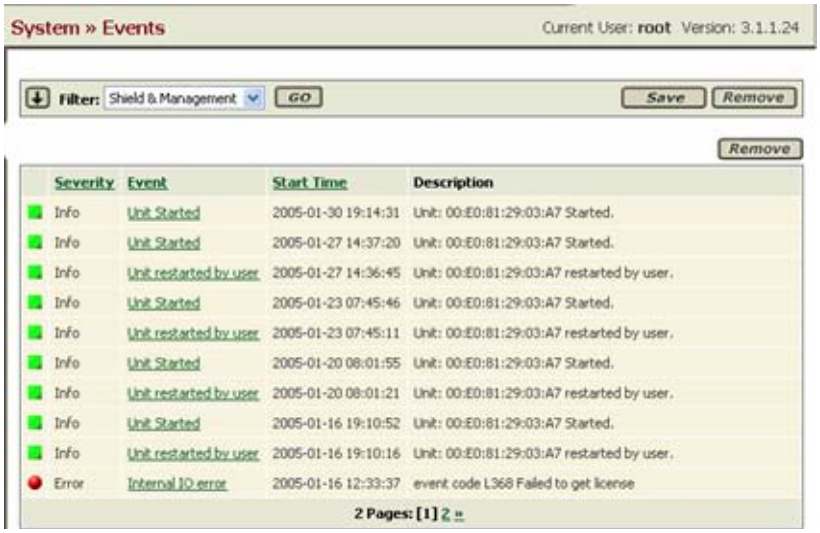
The Start Time and Last Time fields indicate the first and last time that this event occurred. The Count Field shows the number of times this event took place between the indicated times.

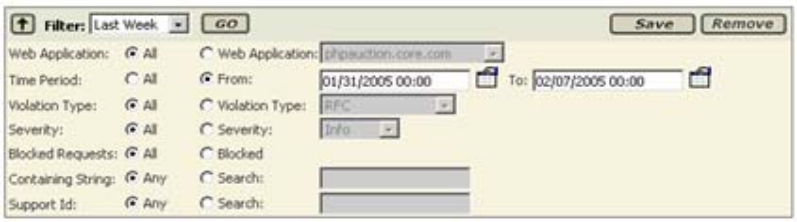
You can display the same report by selecting the Events tab in the System menu. This display includes a filtering tool that allows you to focus on certain events.

Events

This window is very similar to the System Status window, but instead of displaying the status of the TrafficShield security application's units, an advanced filter window is available.

1. Open the filtering tool by clicking the down-arrow icon displayed on the Filter row (you can close it by clicking the button again).





2. Select one or more filtering options.

The filtering options are those that have a radio button. For example, select the Severity radio button and then select a severity level to list only events of the selected severity.

You can select multiple filtering options to further limit the scope of the retrieval. For example, setting a period in From/To and selecting a severity, lists the events of the selected severity level that took place within the specified period.

To cancel the filter in a certain category, check the **All radio** button.

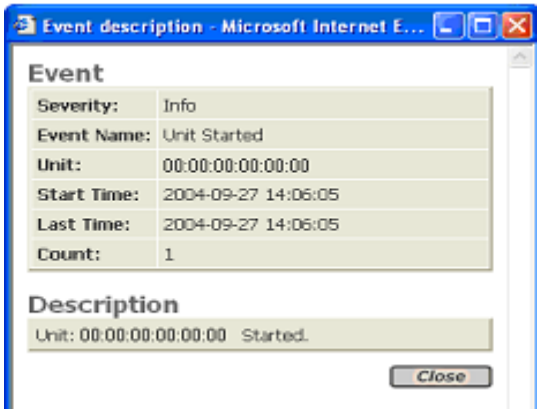
Criteria	Description
Filter	A predefined set of filtering parameters.
Type: Event Of	Filters the events that took place in the units, and events that have been posted to the operating system's log (system Log). Check the box that corresponds to the events you want to retrieve. You can select more than one option.
Name: Event	If you want to focus on a specific event, select the Event radio button and then select the event you want in the drop-down list.
Time Period: From/To	To retrieve events that took place in a certain period, select the From radio button. Then, use the icon in the From/To fields to select the start date/time and end date/time of the period. Note that you can select the time by clicking the time fields at the bottom of the calendar box.
Unit Units	If you want to focus on events that took place in a certain unit, select the Units radio button and then select the unit's ID.
Severity: Severity	To retrieve only events of a certain severity level, select the Severity radio button and then select a level from the drop-down list.
Containing String: Search	Use this option to pinpoint events whose message contains a certain text. Select the Search radio button and type the text.

Unit events

If you want to focus on events that took place in a certain unit, select the Units radio button and then select the unit's ID.

To display more information about the event

1. Click the event link. This displays a description of the event.



2. When you have read the event summary, click the Close button.
3. On the Events screen, click the Go button to activate the filter.
4. Click the Save button, after selecting the retrieval criteria, so you can re-use it whenever you want.
   This opens the following window.

5. Type a name for the selected criteria and click OK.
6. You can delete a criteria definition by selecting it in the Filter list and clicking the Remove button.

Security

Status

The Status tab in the Security menu shows a list of security violations that have occurred. There are two report types available. In Report Type, select:

- Violation Report, to display a list of violations.
- IP Report, to display the IP addresses that committed the violations.

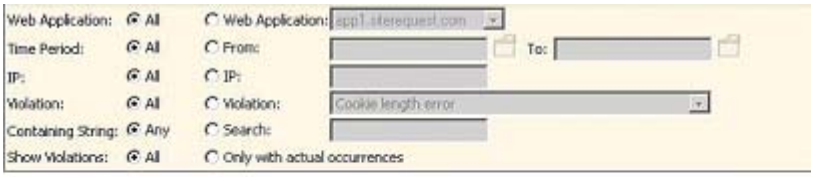
Both reports display the number of requests and the percentage of those requests that occurred from the total requests.

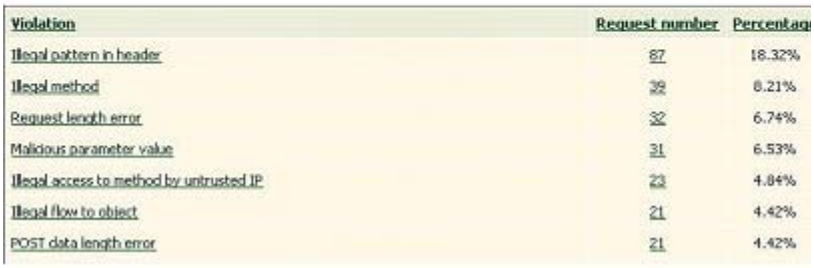
To define the filter criteria:

1. Open the filtering tool by clicking the down-arrow icon displayed on the Filter row (you can close it by clicking the button again).
2. Click the Go button to update the violation display using the latest filter criteria.
3. Click the Save button to save the changes made to the filter criteria, thus creating a customized filter.
4. Use the Remove button to remove customized filters.

   Note: *It is not possible to delete the built in filters.*

5. The filter criteria are displayed in the top part of the window while the filtered violation list is displayed in the bottom part of the window.

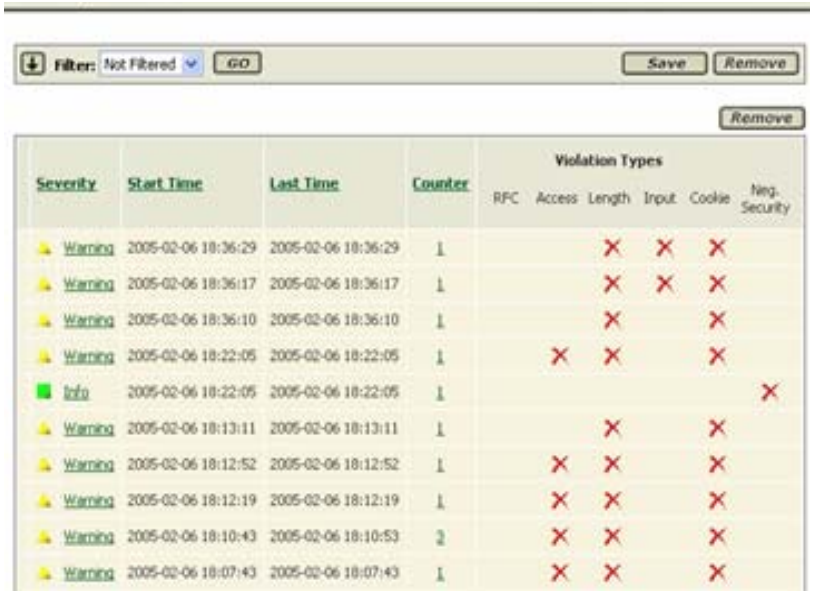




Filter	A predefined set of filtering parameters
Web Application	To focus on events relating to one of the protected Web applications, select the Web Application radio button and then select the Web application from the drop-down list.
Time Period From/To	To retrieve events that took place in a certain period, select the From radio button. Then, use the icon in the From/To fields to select the start date/time and end date/time of the period. Note that you can select the time by clicking the time fields at the bottom of the calendar box.
IP	To retrieve events originating from an IP address, select the IP radio button and then enter the address in the adjacent box.
Violations	To list the events that were registered as a result of a specific attack type, select the Violation radio button and then select the standard attack name from the drop-down list.
Containing String: Search	Use this option to pinpoint events whose message contains a certain text. Select the Search radio button and type the text.

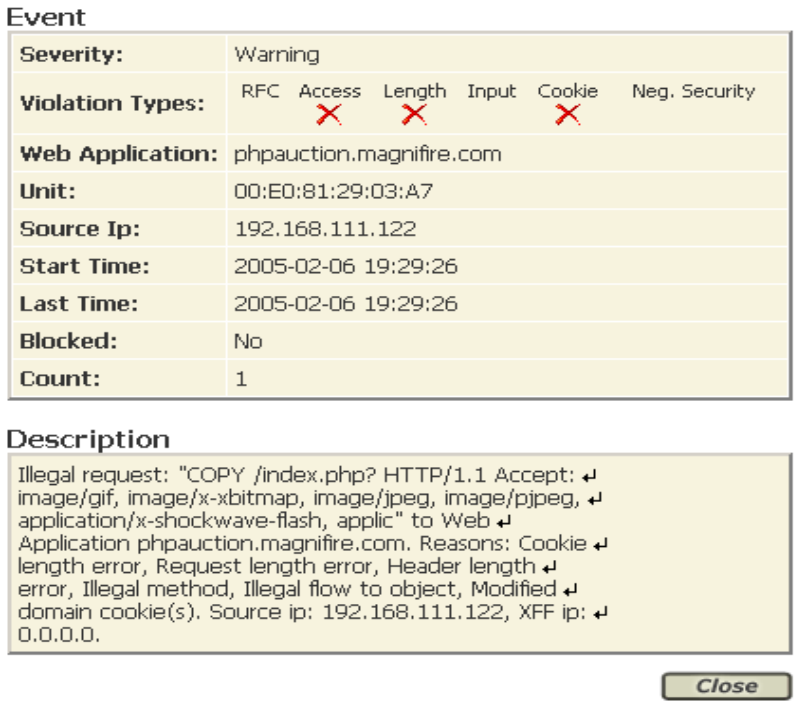
Displaying the events

The Security-Events tab lists the events relating to requests that do not comply with the blocking parameters. For example, you can see a list of events relating to requests that committed a length or a cookie violation.



Events that have been blocked are marked with the ("stop") icon.

To display more information about the event, click the severity link. This displays a description of the event.



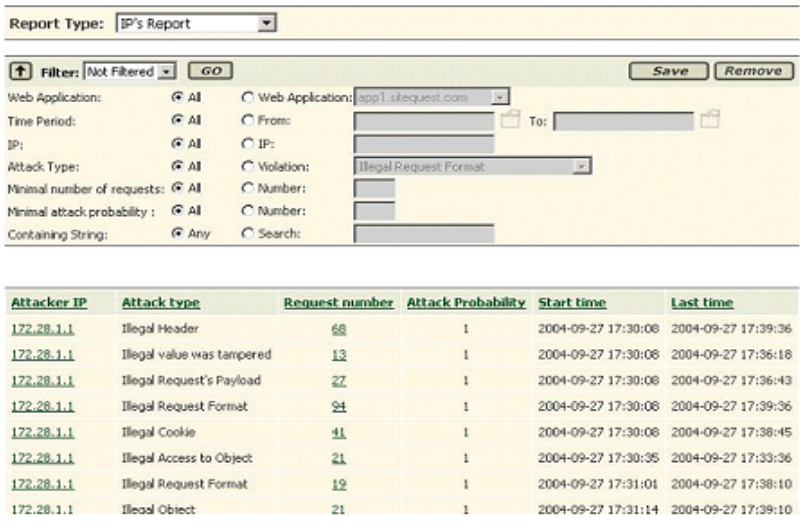
Reports on illegal requests

Attacks report

This report provides a more global view on a number of illegal requests of a given type.

When sent at a high frequency, these illegal requests are considered as a clear intention to cause a specific damage. For example, the TrafficShield security application detects such attack types as “buffer overflow,” "parameter value tempering," "forceful browsing," and more. The Reports-Attacks tab puts together such sets of illegal requests.

1. Open the filtering tool by clicking the down-arrow icon displayed on the Filter row (you can close it by clicking the button again).



2. Use the **Go** button to update the attack display using the latest filter criteria.

3. Use the **Save** button to save the changes made to the filter criteria, thus creating a customized filter.
4. Use the Remove button to remove customized filters.
5. The columns displayed are:
   - **Request Number**
     The Request Number column indicates the number of requests of the specific attack type. Click a number to display the requests.
   - **Attack Probability**
     The TrafficShield security application calculates and suggests a probability that the certain set of requests already launched an attack.
   - **Start Time**
     This is the first time this attack was noted.
   - **Last Time**
     This is the last time this attack was noted.

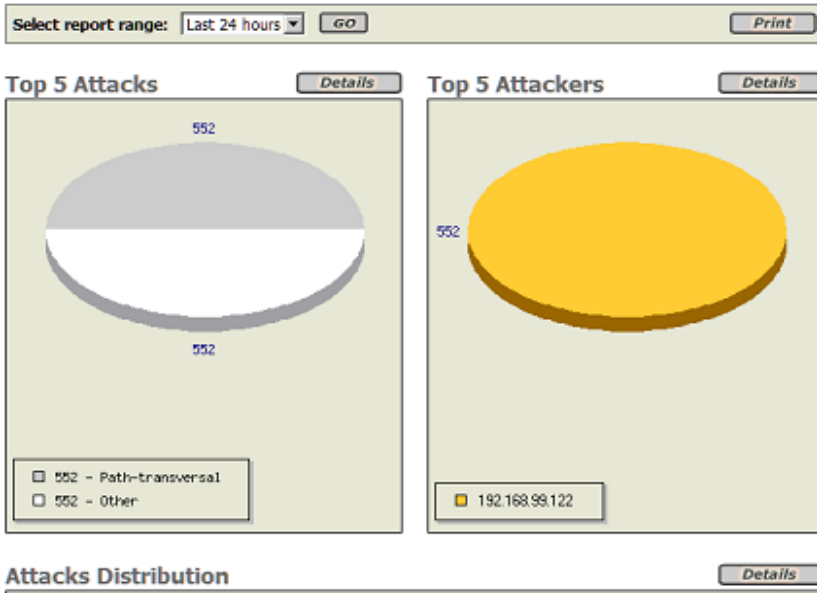
   The options in the Report Type section are as follows:

Criteria	Description
Filter	A predefined set of filtering parameters
Web Application	To focus on events relating to one of the protected Web applications, select the Web Application radio button and then select the Web application from the drop-down list.
Time Period From/To	To retrieve events that took place in a certain period, select the From radio button. Then, use the icon in the From/To fields to select the start date/time and end date/time of the period. Note that you can select the time by clicking the time fields at the bottom of the calendar box.
IP	To retrieve events originating from an IP address, select the IP radio button and then enter the address in the adjacent box.
Attack Type	Select an attack type. This applies, especially, to the Attacks Report that groups together requests that have the characteristics of a standard attack type. You can use it in conjunction with "Minimal number of requests".
Minimal number of requests	Use this parameter to list attacks that included at least a specified number of requests that characterize standard attack types.

Criteria	Description
Minimal attack probability	This is a sorting option that displays the attacks from the lowest probability.
Containing String	Use this option to pinpoint events whose message contains a certain text. Select the Search radio button and type the text.

Executive report

The report is displayed by selecting the Reports > Executive tab. It graphically displays the attack statistics.



This report contains the same type of information as in the Attacks report, only it retrieves the five most frequent attacks or attackers (IP). The Details button functions like the links in the Attacks report, listing attacks or IP addresses.

The Attacks Distribution section displays the attack types over time. The Details button displays the same information in textual format.

Activity

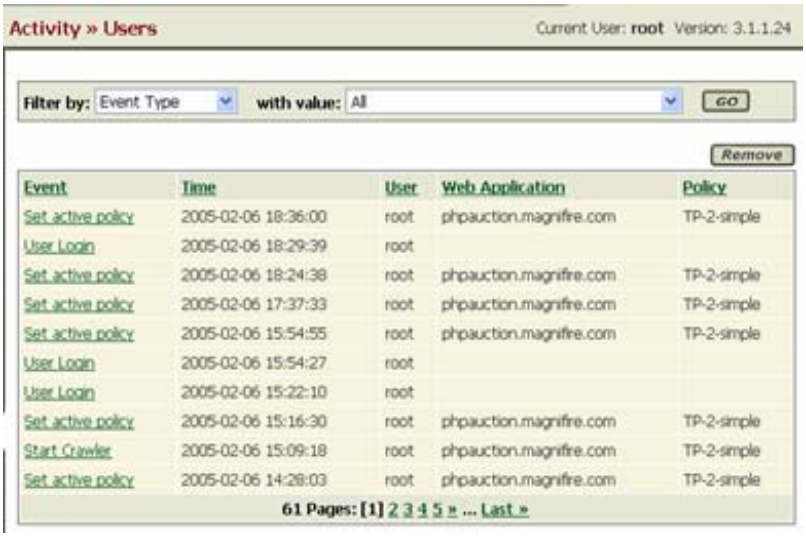
User activity consists of operations such as logging on to TSMS or adding a new policy.

Users

You can use the monitoring tool to examine the user activities that took place in the system.

To monitor user activities:

1. On the top menu, click the **Monitoring** button.
2. In the **Activity** section of the navigation pane, select the Users tab.



3. In **Filter By**- By Value you select the events to display.
   For example, in Filter By, select Policy, and in With Value, select the name of a policy, and click the Show button to list the user activities that took place in relation with the indicated policy.
   Another example: Select Duration-Last Hour returns events posted in the last 60 minutes.
   The **Remove** button deletes all of the listed events.
4. To list the events that meet the criteria, click the **Go** button.

Glossary

ARP

Address Request Protocol: (a networking protocol). A method for finding a host's IP address from its Ethernet address. The sender broadcasts an ARP packet containing the IP address of another host and waits for it (or some other host) to send back its Ethernet address. Each host maintains a cache of address translations to reduce delay and loading. ARP allows the IP address to be independent of the Ethernet address, but it only works if all hosts support it.

ARP is defined in RFC 826.

The alternative for hosts that do not do ARP is constant mapping.

Check Object

Indicates whether TrafficShield security application should check the Object requested in the HTTP/HTTPS request against the list of its known objects before it forwards the request to the server. In case it doesn't find the requested object in the list, it generates a violation that, based on the blocking policy, can cause the request to be blocked

Cookie

A packet of information sent by an HTTP server to a World-Wide Web browser and then sent back by the browser each time it accesses that server. Cookies can contain any arbitrary information the server chooses and are used to maintain state between otherwise stateless HTTP transactions. Typically this is used to authenticate or identify a registered user of a Web application without requiring them to sign in again every time they access that Web application. Other uses are maintaining a "shopping basket" of goods you have selected to purchase during a session at a Web application, Web application personalization (presenting different pages to different users), and tracking a particular user's access to a Web application.

DELETE

An HTTP request type that requests to delete a resource on the web server.

Domain Name

A series of alphanumeric strings separated by periods, such as **www.siterequest.com**, that is an address of a computer network connection, and that identifies the owner of the address.

Dynamic Parameter

A dynamic parameter is a parameter in a request where the set of legal values this parameter can have is changing dynamically, and usually depends of the user session. For example, in a banking application the account number is a dynamic parameter, since each user has its own set of legal account numbers that this parameter can have. This set of legal account numbers is dynamically generated by the server and embedded in the web

page sent to user. TrafficShield security application extracts this list of legal values from the web page that is sent to the user, and uses them to verify that the value sent in the request for the dynamic parameter is legal.

Dynamic Value

See *dynamic parameter*

Entry Point

A web page that could be the first requested page in the Web application: an end-user could get to the Entry Point by typing a URL in the browser window, opening a favorites menu, be linked from a different Web application or e-mail client. The end user could also get to the Entry Point by clicking a back button of the browser.

Flow

The defined access path for a browser to get from one object to another specific object.

GET

A type of HTTP request that does not have a content body

Learning

A process of making a policy more accurate by verifying how the policy complies with the traffic requests, and if there are discrepancies between the policy and the traffic requests, then translating these discrepancies into a suggestion for modifying the policy. The learning phase also enables the system administrator to verify that the policy is not generating any false positives before turning on the blocking feature. The learning process can be used to fine-tune any policy component such as requests length, parameters, and values. In case new objects are added in the Web application, TrafficShield security application can learn those objects and their flows using the learning engine.

Length-Cookie

The length of the cookie.

Length-Post Data

The length of the Data that comes with a POST request.

Length-Query String

The length of the Query string.

Length-Request

See *Request Length*.

Length-URI

The length of the URI in characters.

Meta character

A character or a sequence of characters that has a special meaning (<SCRIPT >, \ , SELECT, INSERT, ; ,`, <).

Method

The HTTP/HTTPS request method, e.g. GET, POST, HEAD, PUT, and DELETE.

Non Existent Object

The flow did not match the defined flows.

Object

A file or a script that generates web pages on the web server that can be requested by a user,

Object is Allowed to modify domain Cookie

In case an Object (i.e., a web page) includes a JavaScript/java applet/flash as part of the client-side and can change a domain cookie value, the object should by defined as "Object is allowed to modify Cookie."

Path Traversal

An HTTP Attack that uses patterns like ../../ to get access to files not intended to viewed above the WWW root, or in order to cross directories on the server.

Policy

A set of rules that enables TrafficShield security application to understand if a request is valid.

POST

A type of HTTP request, in which a query is put into a content body and possibly compressed or encoded.

PUT

An HTTP request type that requests a content change on the web server.

Query String

Part of an HTTP request that specifies a list of parameters and values into a CGI script. For instance:

http://www.siterequest.com/index.cgi?param1=value1&param2=value2

Anything that comes after the question mark in the example above is a query string.

Referrer

A web page that requests other objects An HTML page could request picture files and other html objects to be downloaded, but pictures cannot cause other objects to be downloaded. For example, HTML, asp, php pages are usually Referrers, while gif and jpeg images are not.

Regular Expression

Used by UNIX utilities such as grep, sed and awk, and by editors such as vi and Emacs. A regular expression (regexp) is a sequence of characters which provides the user with a powerful, flexible and efficient test processing tool.

For more details on how to write regular expressions please refer to the many books written on this subject; for example: Mastering Regular Expressions, by Jeffrey E.F. Frieldl, Published by O'Reilly & Associates, Inc.

Request Length

The total Length of the HTTP request (in characters) which includes the request line, all headers, cookies, and post data.

Server IP

The IP address of the Web Server that TrafficShield security application is protecting (usually this is an internal IP address).

Service IP

The external IP address on which TrafficShield security application is listening for http requests. (Usually this is the IP address that the DNS A record of the Web Server is mapped to.)

Shield Unit

The on-line enforcing mechanism responsible for TCP session termination, requests parsing, and analyzing.

Static Parameter

A parameter in the request where its values are chosen from a known set of values: Name of a Country, Yes/No, etc.

Static Value

See *static parameter*.

Target Frame

The frame to which the object is loaded.

Undefined Flow

The flow did not match the defined flows.

Undefined Object

The object did not match any objects on the list of allowed objects.

URI

Part of the URL that specifies the name of the object requested: in **http://www.siterequest.com/index.html**, **index.html** is the URI.