

Quick And Dirty Java Blocking Using A URI Resource Definition

The goal in this example is to block Java applets from entering the Internal Network but allowing unauthenticated access to the Internet for http from the internal network. First we must define a URI resource definition for what we intend to match. A URI resource is an extension of the rule, in other words it goes beyond the |Src|Dst|Svc| fields and gets more granular into the content of the service. We will be configuring a resource that will match all traffic by a wild card in both directions for http, but will block Java applets in both directions.

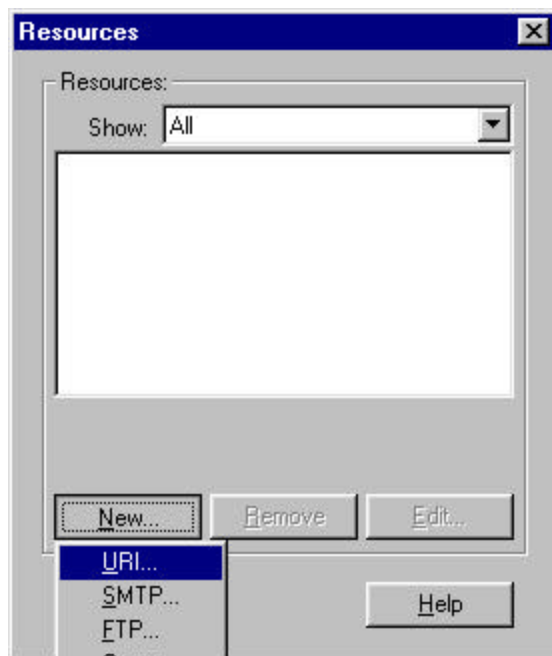
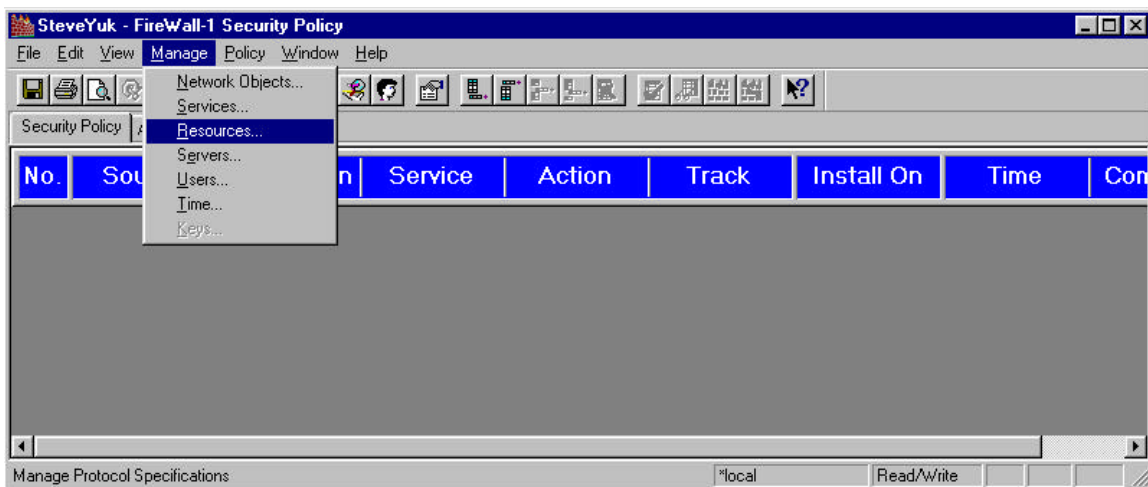
NOTE: You should be running Check Point Firewall-1 version v3.0B or higher.

NOTE: You must have the HTTP security servers installed with default options for the URI to work.

This also goes for creating URI definitions for FTP and SMTP as well.

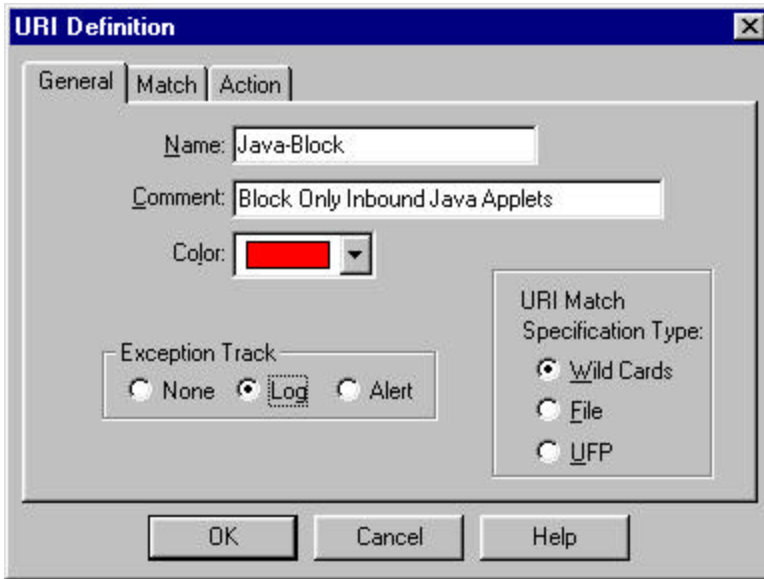
Define The resource Definition for http:

Select the *Resource* drop down menu from the *Manage* menu on the policy editor.



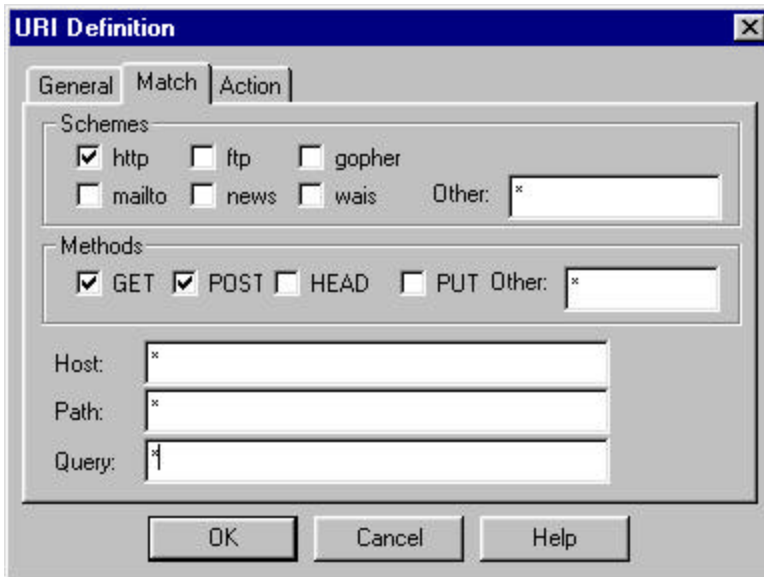
The Resources dialog box appears.

Click on *New* then *URI* to define our URI resource.



Fill in the **General** information for the URI Definition as shown. We will be using Wild Cards in this example because we want the Definition to apply to all traffic. If we were screening URL's we could import a file a specific URL's or utilize a UFP server such as WebSense.

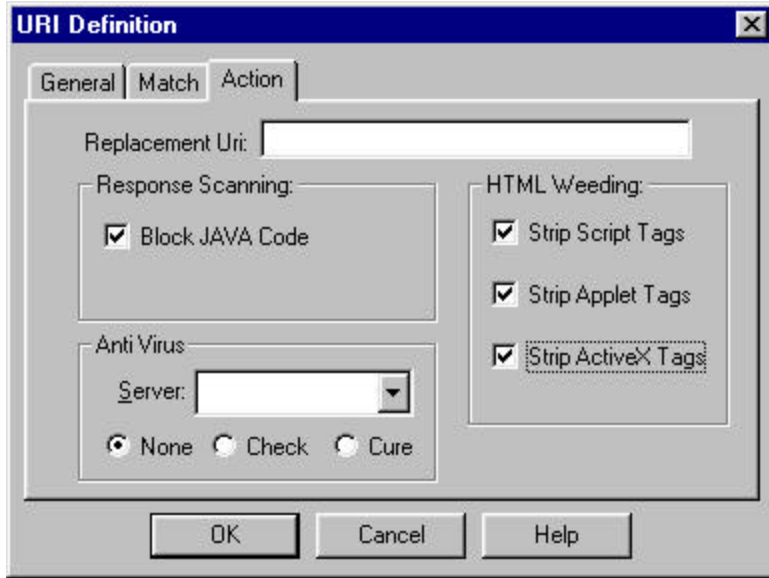
Next Click on the **Match** Tab.



Next we will define the **Match** criteria. For this example we will only select **http** as the allowed **Scheme** and **Get & Post** as our **Methods**. All other **Match** fields will be left as * to indicate a wild card matching anything.

This is where the resource becomes part of a rule. Where if we meet the Source and Destination in a rule, the Service must comply with what we outline in the URI as a **Match** and **Action**.

Next Click on **Action**.



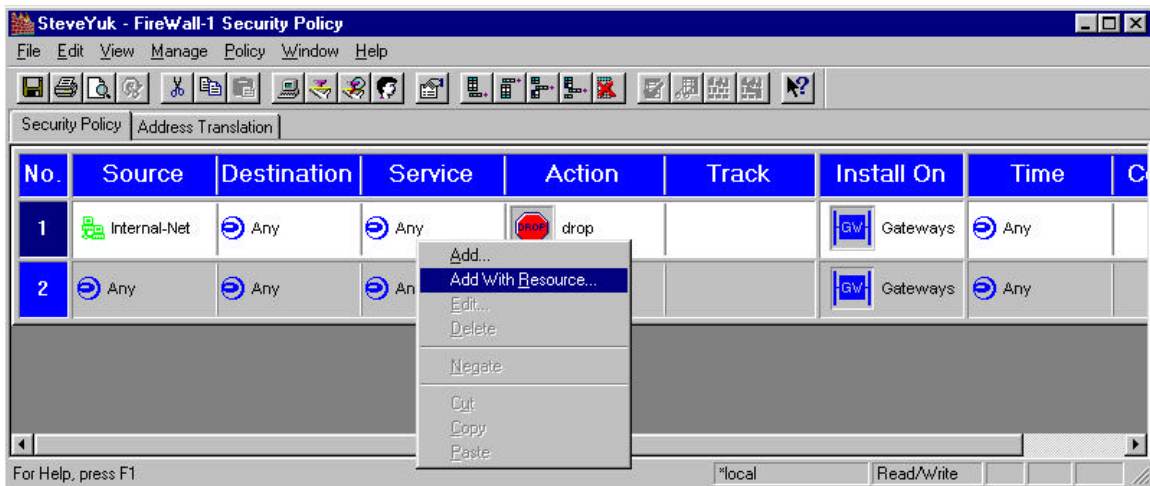
Next we must define our *Action* criteria. This is what the URI will do if all other criteria are met. Think of this as a rule base within a service.

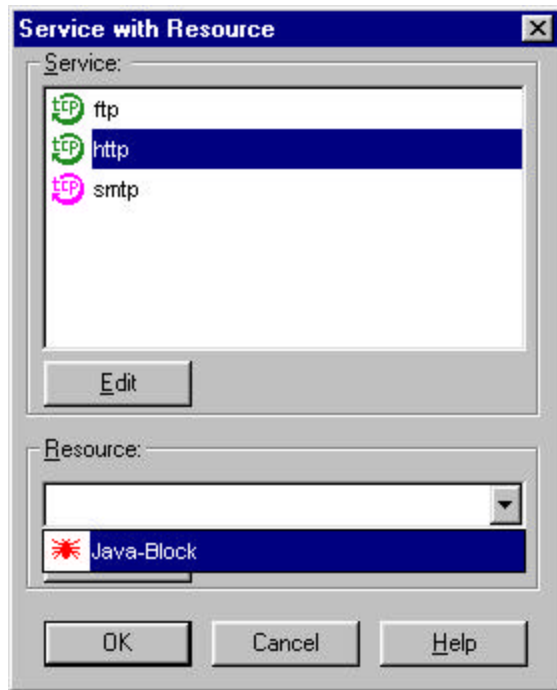
For this example we will select **Block JAVA Code** to block applets and we will also select to weed out the Tags within the page so that they are not displayed.

Click **OK**.

Next we must add a rule to the rule base to allow the internal users to access the Internet for HTTP, but we will block Java with out URI resource definition.

Add a rule to the rule base with the internal net as the source, to any destination and add the service with a resource as shown.

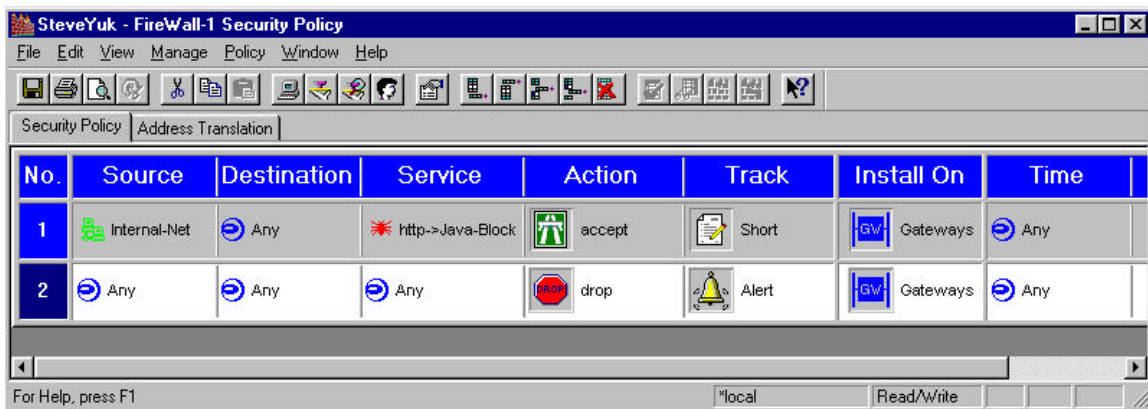




The Service with Resource dialog box appears. Select *http* and the resource *Java-Block*.

Click *OK*.

The completed rule base should look like this. Install the rule base and connect to a Java site such as www.sun.com and attempt to view a Java page. The page should be displayed omitting the Java Applet.



In the log viewer the http security server and URI resource provide detailed logging of the event in the *Info* field.

