

# Effective Solutions for Firewall Rule Cleanup

## Using Athena FirePAC

### Abstract

Firewall rules that provide access to a wide array of services in a large network, while at the same time securing the critical assets from attacks, tend to become very large in size and redundant in functionality. As rule bases become large, administrators become hesitant to modify existing rules and instead add new rules for fear of causing an adverse impact on existing service availability. Over time, rule bases become very bloated, requiring not only more effort in making changes but also having an adverse impact on the firewall performance. It is therefore essential to clean up the rule base and reduce its size. This paper presents some techniques to cleaning up the rule base *along with an effective solution that addresses these automatically for you using Athena FirePAC for Firewall Rule Cleanup.*

### Introduction

Firewalls protecting enterprise networks with an already complex web of inter-connections will inevitably grow more complex because of the need to add rules in order to provide network access and protect against attacks. Ideally, rules would be added to the firewall in an organized manner. Furthermore, rules would be organized and enhanced to suit specific business purposes. Unfortunately, that is not reality. Firewall administrators change; as new people transition into the role, rules are added in an ad hoc manner without realizing that the new rules are redundant and not needed in the first place. Moreover, as the rule bases become large, firewall administrators become hesitant to modify existing rules and instead add new rules for fear of causing an adverse impact on existing service availability. This makes the problem even worse and the job of administrators very difficult if they have to address issues raised during firewall or PCI audits. Here are some of the things that administrators should pay attention to to address the rule bases from becoming large and redundant:

- New generalized rules are added that replace a number of more specific rules that already exist in the firewall. This typically happens when specific rules are added initially to the firewall to allow services to specific hosts or subnets and then more general rules are added when the business scope expands to other networks or services or much larger subnet or services (sometimes “any” network or service). When this happens, the previous specific rules become redundant and need to be cleaned up.
- New rules are added without realizing that one or more rules preceding or succeeding the new rule already handle the functionality being addressed by the new rule. Depending on where the rule is added, the new rule might never be triggered. This happens when there are multiple rules in the rule base that each cover portions of the new rule and together completely cover the new rule. As a general practice, before adding new rules, existing rules should be queried to see if they can be modified to satisfy the change request. Change requests do not happen in a vacuum, they are made to serve a business purpose that probably already exists and is being enhanced.
- New rules are added as a special case of one or more subsequent rules to exhibit special behavior (often temporarily). These special cases include enabling or disabling logging only for specific hosts or services instead of the much large networks or services being handled by the subsequent rules, performing application inspection for specific services involving specific assets, tracking quality of service attributes, and requiring user authentication for specific services or assets. Sometimes special cases are created at the beginning of the ruleset for the most used traffic to increase firewall performance. Some of these rules are temporary in nature, sometimes added to track usage or do some testing; however these are not cleaned up even when the reason for adding these in the first place is no longer relevant.
- Rules become stale when the business reason for adding the rules goes away. These rules are not used anymore but remain in the firewall impacting maintenance and firewall performance. These rules can be identified and cleaned up by tracking their usage in the firewall either by logging these rules or by looking at rule usage hit counts. Attention must be paid to those rules that might be used only during a specific time of the year and will not be in the firewall logs unless logs are captured and analyzed for a sufficient time period. Either way, rules should be removed after confirming with the business owner of the rules.

## Identifying redundant rules that are never triggered

The redundant rules that can be safely removed without affecting the firewall behavior are rules that are never triggered because preceding rules cover them and match first. Identifying redundant rules that are completely covered by one rule is a little easier than finding rules that are covered by more than one preceding rule(s). In the first case, you can compare the two overlapping rules that are in question side by side and verify if the first rule covers the second rule completely. In the second case, each of the preceding rules will each cover only portion of the redundant rule but together they cover the redundant rule completely. This requires a more detailed analysis of all the overlaps that exist between the rules.

Here is the process for identifying these redundant rules:

1. Understand how rules are organized into access lists, what each rule is addressing and the sequence in which the rules are evaluated. This generally means understanding how rules are organized in a sequence: access lists in Cisco firewalls, Zone to Zone policies in Juniper NetScreen or the flat rule base in Check Point so that the redundancy analysis is limited to rules within the access list instead of the whole. You need to understand the source, destination and service elements used in each of the rule(s). If object groups are used or if multiple objects are used, you need to understand the complete expanded combinations in the rule.
2. Find out rule by rule in a given access list the rules that overlap with each other on all the Source, Destination and Service elements of the rule. For each overlap found for a rule,
  - a. If the overlap with another rule higher up in the rule sequence of the access list is such that it completely covers this rule, then this rule is never triggered and can be removed.
  - b. If there is no single rule that completely covers this rule, but there are multiple rules higher up in the rule sequence that together cover this rule, then this rule can be removed.

*Cisco PIX Examples: Rule on line 134 is covered by rule on line 85. Similarly rules on lines 180 and 181 are covered by rule on line 140. Rules on lines 134, 180 and 181 can be safely removed without affecting firewall behavior. The address object group inet\_servers contains web\_servers and mail\_servers as members. Similarly the service object group inet\_svcs contains web\_svcs and mail\_svcs as members.*

```
85  access-list acl_inside permit tcp any any eq ftp
134 access-list acl_inside permit tcp host 172.16.0.15 any eq ftp
140 access-list acl_outside permit tcp any object_group inet_servers object_group inet_svcs
180 access-list acl_outside permit tcp any object_group web_servers object_group web_svcs
181 access-list acl_outside permit tcp any object_group mail_servers object_group mail_svcs
```

## Identifying redundant rules that are covered by succeeding rules

Identifying redundant rules that are covered by one or more succeeding rules is a little more difficult and attention needs to be paid to the rule options before removing them. Some of these rules might have been added as a special case of succeeding rules for special processing; for example to enable or disable logging, perform application inspection, user or other forms of authentication, quality of service attributes, etc. It may be that some of the special cases were needed for a temporary time period only but never cleaned up. So all these cases should be reviewed and any special cases that no longer require the special processing should be removed.

Here is the process for finding and removing these rules.

1. Find out rule by rule in a given access list, the rules that overlap with each other on all the Source, Destination and Service elements of the rule.
2. For each overlap found for a given rule, the rule can be removed
  - a. If there is a rule below in the rule sequence that is the first rule that completely covers this rule and have the same rule action and rule options.
  - b. If there are one or more rules in the access-list that together cover this rule and have the same rule action and rule options. If any of the rules that overlap this rule have a different action, then you cannot remove this rule.

*Juniper NetScreen Examples: Rule with policy id 11 is covered by policy id 34 and can be removed. Policy id 87 is covered by policy id 90, however policy id 90 has user authentication, so policy id 87 cannot be removed unless user authentication is not required in policy id 90.*

```
set policy id 11 from "Trust" to "DMZ" "Host_Int_172.16.0.24" "Host_Dmz_192.168.1.4" "HTTP" permit log
set policy id 11
set service "HTTPS"
exit
```

```
set policy id 34 from "Trust" to "DMZ" "Any" "Net_Dmz_192.168.1.0_m24" "HTTP" permit log
set policy id 34
set service "HTTPS"
exit
```

```
set policy id 87 from "Trust" to "DMZ" "Host_10.3.1.22" "Net_Dmz_192.168.1.120" "SQL-SVCS" permit log
exit
set policy id 90 from "Trust" to "DMZ" "Host_10.3.1.0_m24" "Net_Dmz_192.168.1.0_m24" "Any" permit auth
server "Int_Auth_Server" user-group "intranet" log
set policy id 90
exit
```

## Identifying unused rules

Identifying rules that become stale because the business purpose for that rule went away might not be straight forward because of absent documentation on the firewall rules and the transitioning of the firewall administrator and/or the business owner of the rule. Identifying these rules then requires analysis of usage data in the form of firewall log data or access list hit count. Depending on the firewall, these rules need to be enabled with the log option first before collecting the usage data from the firewall. Once the usage data is collected for a reasonable time period, then the usage data can be analyzed to find rules that have zero hit count. These rules should then be disabled with appropriate documentation and can be removed after monitoring for any service availability complaints.

For Check Point and Juniper NetScreen firewalls, the firewall logs are needed for determining rule usage. Any rule which has a tracking option enabled and is not found in the firewall log data can be deemed as unused. For Checkpoint firewalls, Rule UID in the firewall log data is can be used to identify the used rules in the firewall rule base. For NetScreen firewalls, policy ids in the firewall syslog data can be used to identify the used rules in the firewall rule base.

For Cisco PIX/ASA/FWSM firewalls, access list hit counts can be used for determining rule usage. Cisco firewall logs do not contain ACE line numbers information, so it is easier to analyze the access list hit counts tracked by the firewall. Any rule which has a hit count of zero is considered as unused. The access list hit counts can be obtained by using the command "show access-list". The access list hit counts are reset when the firewall is restarted. The hit counts can also be reset explicitly using the command "clear access-list [id] counters".

## Identifying unused objects

There are two kinds of unused objects in the firewalls. Objects that are not used in any rules either directly or indirectly through an object group membership are somewhat easier to identify. Most of the firewalls will not let you delete an object that is still being used in the firewall. So you can safely delete in most cases, any object you want to delete. However identifying objects that are referred in rules but are not really used, requires usage data analysis. Unlike the unused rule analysis above, this requires matching the Source, Destination addresses and the services for each record in the firewall logs against the objects and object group members used in the rules. This analysis will help in cleaning up not only unused objects in a rule but also unused members of an object group across all the rules.

## How does Athena FirePAC help?

Athena FirePAC automates the cleanup and optimization of firewall configurations. Using FirePAC, administrators can isolate more rules for removal than any other solution. FirePAC performs a thorough analysis of the rule overlaps and dependencies to identify every possible rule relationship. Whether your firewall has a few hundred or thousands of rules, FirePAC guarantees you will reduce your maintenance burden by at least 10-30%. Specifically, Athena FirePAC allows you to identify:

1. All rules that can be safely removed without affecting firewall behavior. These include rules that are covered by one or more preceding or succeeding rules.
2. Additional candidates for removal based on special processing options in the rules.
3. The most used and unused rules from firewall logs or access list hit count data.
4. All unused objects that are not referred in any rules.
5. Optimized rule order, which takes the rule order dependencies into account to move the most used rules as far as possible up the rule set.

See an example FirePAC rule cleanup and optimization report here at:

<http://www.athenasecurity.net/pdf/example-pix-02-FirewallCleanup.pdf>

## Who is Athena Security?

Over 300 companies use Athena products to clean up the firewall rules and reduce the risks to critical hosts by eliminating the vulnerabilities, non-compliances and errors in firewall infrastructure. Athena FirePAC is an affordable, easy to use firewall analysis tool for large or small enterprises. It confirms that each firewall is configured to behave correctly. FirePAC performs safe, offline analysis on the rule base to predict how data flows through the firewall to reach critical hosts. Install it on your desktop in seconds, and generate reports that reveal exactly how your firewall is working. See more at <http://www.athenasecurity.net>