>
Published on *NetworkWorld.com Community* (http://www.networkworld.com/community)

The Future of Firewall Policies

By *Scott Hogg*
Created *Jan 19 2012 - 5:49pm*

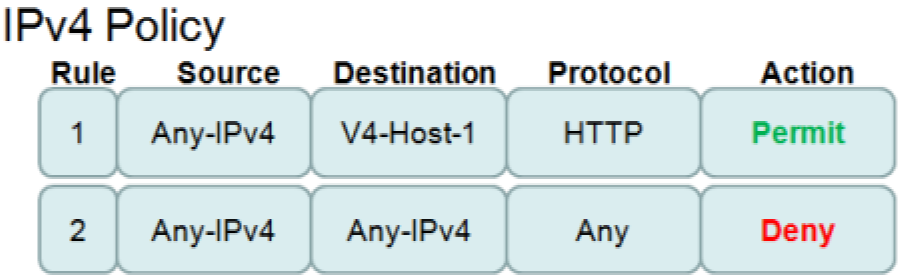
The majority of today's firewalls have only IPv4 source and destination address objects in their policies. However, many of our firewalls are IPv6-capable and allow security administrators to configure either IPv4 or IPv6 policies. Over time, maintaining two firewall policies will become burdensome. We will eventually want firewalls that use a consolidated policy that incorporates IPv4 [1] and IPv6 [2] together or firewalls that can be configured easily with objects that have both IPv4 and IPv6 characteristics.

Your perimeter firewall is on the critical path of your IPv6 deployment schedule. The firewall you use today like has some basic IPv6 packet filtering functionality. Most of the major firewall manufacturers have basic IPv6 packet-filtering capabilities. A smaller subset of those have Unified Threat Mitigation [3] (UTM), Intrusion Prevention System [4], application defenses, content filtering, and other features that are IPv6-enabled. If your current firewall does not have any IPv6 capabilities then it is likely that vendor is ignoring IPv6. If your current firewall vendor does not yet have IPv6 on their development roadmap, you will likely need to purchase a new firewall to gain IPv6 capabilities. When you are shopping for an IPv6-capable you should look for firewall vendors that support Extension Headers, intelligent fragmentation handling, Path MTU Discovery (PMTUD), and granular filtering of ICMPv6 [5] messages and multicast traffic.

Before you embark on adding IPv6 objects and rules to your firewall, it is important to consider how your current firewall handles packets behind the scenes. Most firewalls are "first-match" rule sets where the first rule in the policy that fully matches the packet determines the action. First-match policies are pretty easy to comprehend and debug. If no rule is matched then the packet falls through the policy to the last rule which typically implicitly drops the packet. Examples of this style are the Cisco ASA [6], Cisco Router Access-Lists, Juniper SRX [7]/SSG [8], Check Point [9], Fortinet [10], IPtables [11]/IP6tables, and ipfirewall [12] (ipfw/ip6fw). However, there are some firewalls that are "last-match" in nature. This can be potentially confusing because we typically think of every firewall policy ending with an implicit deny-all statement as the last rule. Examples of this type of "last-match" filtering are the OpenBSD Packet Filter [13] (pf), and IPFilter [14] (ipf). However, pf can be modified to operate in a more traditional "first-match" way with the "quick" keyword.

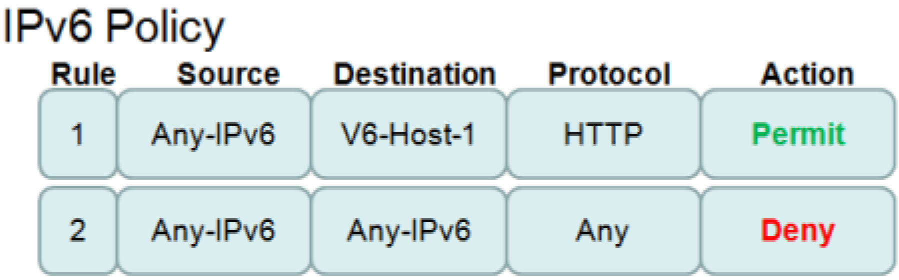
Regardless of whether your firewall is "first-match" or "last-match" you should consider how you will begin to add IPv6 rules to your firewall before you get too far along your transition. You would not want to start configuring your firewall one way and then realize that you have to completely rework the policy after 5 years of adding IPv6 objects and rules. Also, you should consider how you are going to name IPv6 host, network or group objects and how you will use those in your rulebase. You may even want to take this opportunity to rethink how you have named IPv4 objects in the past. You may consider adding some set of characters to the end of an IPv6 object name like "-v6" or "-6" to remind yourself that that object is an IPv6 object. Chances are you have not named your IPv4 objects with a "-v4" or "-4" at the end of them to remind yourself they are IPv4-only objects.

Most firewalls today have separate policies for IPv4 and IPv6. This is true of the Cisco ASA [6]. It uses separate "ip access-list" and "ipv6 access-list" commands to define the policy and then it uses "access-group" commands to apply the IPv4 or IPv6 access-lists to the appropriate interface and direction. The following picture shows what this logically looks like. You can see that there are two separate policies. The IPv4 objects are only used in the IPv4 policy and the IPv6 objects are only used in the IPv6 policy.



IPv4 Policy

Rule	Source	Destination	Protocol	Action
1	Any-IPv4	V4-Host-1	HTTP	Permit
2	Any-IPv4	Any-IPv4	Any	Deny

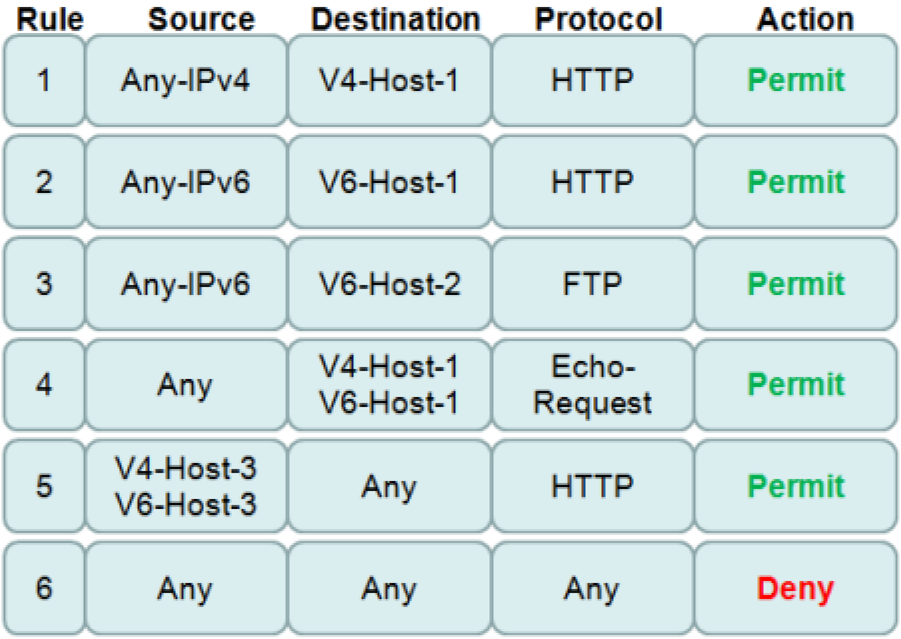


IPv6 Policy

Rule	Source	Destination	Protocol	Action
1	Any-IPv6	V6-Host-1	HTTP	Permit
2	Any-IPv6	Any-IPv6	Any	Deny

The advantage of this technique is that you can easily see how your IPv6 policy grows over time as your IPv6 deployment grows. You can easily see the permissions for a specific IPv6 host and make sure that you are only allowing the IPv6 access required for that host. Over time, the IPv6 policy will grow and be similar in size to the IPv4 policy. Each of the two policies are "first-match" in nature and so you can see the logic in the IPv6 -only or IPv4-only policy. However, the downside to this approach is that eventually you will have twice as much work to perform for any new addition to the environment. You will need to remember to make equal changes to the IPv4 and IPv6 policies. It could become difficult to troubleshoot if you added an IPv4 address object for a server but forgot to add the IPv6 address object for the server and now you do not know why IPv6 packets are being blocked.

There are some firewalls that have a single combined policy that contains both IPv4 and IPv6 objects in a combined list of rules. IP hosts or networks can be defined using either IPv4 or IPv6 addresses. These objects can be either hosts, networks, or groups of other objects. Check Point [9] and Palo Alto Networks [15] firewalls are examples of firewalls that use a single firewall policy for IPv4 and IPv6 rules. The picture below shows conceptually what this might look like.



Rule	Source	Destination	Protocol	Action
1	Any-IPv4	V4-Host-1	HTTP	Permit
2	Any-IPv6	V6-Host-1	HTTP	Permit
3	Any-IPv6	V6-Host-2	FTP	Permit
4	Any	V4-Host-1 V6-Host-1	Echo-Request	Permit
5	V4-Host-3 V6-Host-3	Any	HTTP	Permit
6	Any	Any	Any	Deny

You can see that there are IPv4 and IPv6 objects used in the same policy. Some lines like Rule #1 have IPv4-only objects and some lines like Rule #2 and Rule #3 have IPv6-only objects. You can also create rules like Rule #4 and Rule #5 that have combined IPv4 and IPv6 objects. The benefits of having a single combined policy is that it is easier to manage. For example, you can see how IPv6 can be added to existing rules if you create a new IPv6 object and add it to the same rules that the IPv4 object apply to. In this way you can make sure that you are permitting the same level of access to a system regardless of the IP version. Furthermore, you could create a group for "Host-1" that combines the "V4-Host-1" address and the "V6-Host-1" address. Then any rule that uses the group object "Host-1" would allow either IPv4 or IPv6 access to or from that host. However, you must be careful that adding an IPv6-address object to a group is not creating an overly-permissive policy.

You should also realize that even if you have a combined IPv4/IPv6 policy that behind the scenes are two different sets of packet handling logic; one for IPv4 and one for IPv6. It is obvious that you cannot have a rule match of an IPv6 packet against an rule with IPv4 addresses. However, for ASIC-based firewalls like Fortinet [10] and Palo Alto Networks [15], their chips can process IPv4 of IPv6 packets equally quickly.

Over time I envision that firewall manufacturers will move to a model where a single object will have both an IPv4 and IPv6 address. Imagine a host object that has two characteristics; an IPv4 address and an IPv6 address. Because we will all be operating dual-protocol networks for 10 or more years we need to think about what life will be like maintaining duplicate firewall policies. I believe that it will be easier to administer a firewall that has objects that have IPv4 and IPv6 addresses as attributes and then the rule will apply to that object regardless of IP version. Eventually firewalls will have full feature parity between IPv4 and IPv6, but today that is not the case.

As you are planning for your deployment of IPv6 you will need to think about how you will manage your IPv6-enabled firewall policies. Consider how your object naming convention will be adapted for IPv6 objects and how you will create rules for IPv6 hosts or networks. It is important to think this through before you have already created many rules that may not be organized the way you want. You will live with your choice for the next 30 years so make an educated decision.

Scott

Firewall & UTM IPv6 combined policy firewall policy first-match IPv6 last-match rule object separate policies

Source URL: http://www.networkworld.com/community/blog/future-firewall-policies

Links:
[1] http://en.wikipedia.org/wiki/IPv4
[2] http://en.wikipedia.org/wiki/Ipv6
[3] http://en.wikipedia.org/wiki/Unified_threat_management
[4] http://en.wikipedia.org/wiki/Intrusion_prevention_system
[5] http://tools.ietf.org/html/rfc4443
[6] http://www.cisco.com/go/asa
[7] http://www.juniper.net/us/en/products-services/security/srx-series/
[8] http://www.juniper.net/us/en/products-services/security/ssg-series/
[9] http://www.checkpoint.com/
[10] http://www.fortinet.com/
[11] http://www.netfilter.org/
[12] http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/firewalls-ipfw.html
[13] http://www.openbsd.org/faq/pf/
[14] http://www.freebsd.org/doc/handbook/firewalls-ipf.html
[15] http://www.paloaltonetworks.com/