Managing ACLs

The ACL API

ACL Overview

The Access Control List (ACL) filters packets passed from the AppNexus core switch into your VLAN. An ACL is made up of an ordered set of Access Control Entries (ACEs) that represent permit and deny statements applied to certain ports and incoming and destination IP addresses. For example, the below ACE permits TCP traffic from any IP address to the IP address 1.1.1.1:

```
permit tcp any host 1.1.1.1
```

Here is an example of an ACL made up of several ACEs. Note that the order of ACEs matters, because a core switch tests packets against ACEs one by one and stops checking after the first match. If no conditions match, the switch denies the packet.

```
remark - allow HTTP from world to instance LAX1:210
permit tcp any host 68.67.169.12 eq 80
remark - allow 40000-41000 ports from VLAN LAX1:2071 (subnet of 256 IPs)
permit udp 64.208.138.0 0.0.0.255 any range 40000 41000
remark - allow SSH from world
permit tcp any any eq 22
remark - allow all traffic (all source and destination ports) from 1.2.3.4
to the whole VLAN
permit tcp 1.2.3.4 any
```

Manage-vlan Tool

ACLs are set and modifed by customers using parameters in the `manage-vlan` CLI tool:

```
manage-vlan get-acl --vlan-id vlan_id [--file path] [--username]
 manage-vlan set-acl --vlan-id vlan_id (--file path | -) [--force]
[--username]
 manage-vlan append-acl --vlan-id vlan_id (--file path | -) [--username]
 manage-vlan validate-acl (--file path | -) [--username]
```

ACEs can be read either from a `--file` or via standard input.

- `manage-vlan get-acl`. This command lists the current ACL for your VLAN. If you specify the `--file` optional parameter, you can output the ACL is in the corresponding file.

Examples:

- `manage-vlan get-acl --vlan-id NYM1:2071 --username <USERNAME>`
- `manage-vlan get-acl --vlan-id NYM1:2071 --file nym1-vlan2071.acl --username <USERNAME>`
- `manage-vlan set-acl`. This command replaces the current ACL with a new one. If you attempt to erase the ACL completely, you will be prompted to enter "--force" as a precaution.
- `manage-vlan append-acl`. This command appends one or more new ACEs to the end of the current VLAN ACL.
- `manage-vlan validate-acl`. This command validates the syntax and semantics of ACEs without applying them to your VLAN

Examples:

- `manage-vlan validate-acl --file /path/to/file/acl.example`
- `cat /path/to/file/acl.example | manage-vlan validate-acl -`

ACL Syntax and Validation

ACLs must be in a specific format to be read by the API. We have chosen the Cisco format.

```
{permit | deny} protocol source [operator port] destination [operator port]
```

Example:

```
permit tcp any host 1.2.3.4
deny tcp any any
```

- Possible protocol values: ip, tcp, udp, gre, esp, ahp
- Source and destination may be specified in one of three ways:
  - A subnet: network address and network mask (note that cisco notation for "inverse masks" must be used) separated by a space. E.g. "171.69.198.0 0.0.0.255"
  - A single host. E.g. "host 1.2.3.4"
  - Any host, from 0.0.0.0 to 255.255.255.255. Use "any"

An operator and port combination specify the source or destination port when the ACE protocol is set to tcp or udp. Operators include: eq (equal), gt (greater than), lt (less than), and range (requires two ports numbers and represents inclusive range).

In addition to ACEs you can place remarks (comments) in ACLs. The remarks are needed usually for documenting the the ACL to make it easier to understand. For example:

```
remark - allow SSH from world
permit tcp any any eq 22
```

Note that in case you need to open **SNMP** to your instances/VLANs from beyond the AppNexus network, it is not enough to open 161 port via `manage-vlan`, as we also need to open an ACL on our border routers as well. Please open a Support ticket, requesting this to happen.

⚠ **NOTE**: The set-acl and append-acl commands will validate ACEs for syntactical correctness, but will not look at the overall ACL to see if it makes functional sense. Processing of ACLs stops when an ACE/rule is matched.