



Dynamic distributed programmable firewall

Dušan Gabrijelčič, dusan@e5.ijs.si

Motivation

- Policy “deny all, permit allowed” do not cover all cases any more => keep static rules but dynamic are also needed,
- Perimeter security is no longer sufficient, defense in depth is needed => distributed firewalls on perimeter and in the core are needed to be able to cope with threats,
- A way to configure them and maintain their configuration dynamically is needed => flexible access to control firewall capabilities and accountability of response actions, programmability,
- Multiple devices already in the network with similar/different firewall capabilities => use them, extend when necessary,
- Threats varies with time => requires extensibility of the firewall capabilities,
- Consistent, understandable management => policy driven approach.

Firewall Element Architecture

- Elements of the distributed architecture: System Manager, Firewall Elements, Firewall Devices, Hardware Classifiers,
- Firewall capabilities: anything that can be used as response mechanism (drop, pass, resource limit, redirect, log, modify, protect, select, etc. the network traffic),
- Main tasks:
 - Export firewall capabilities of different firewall devices and provide response mechanisms accessible in programmable way,
 - Extensibility of firewall devices capabilities via module environment,
 - Policy based management.

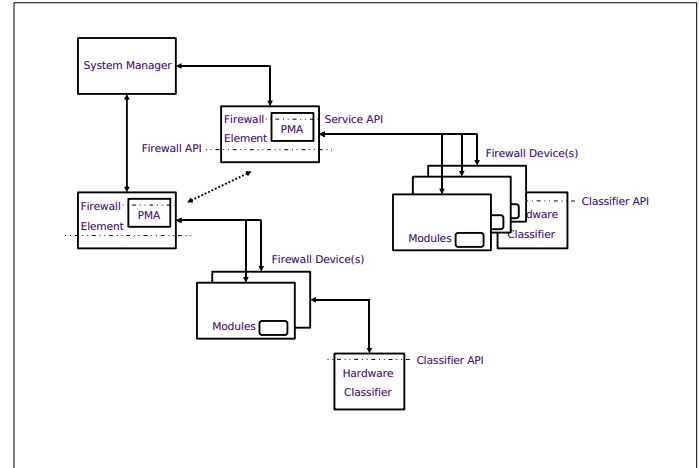
Programmable Interfaces

- Slim, simple design, extensibility,
- Firewall API methods:
 - attach and detach a device,
 - append, insert, modify, remove a rule,
 - create, select, pwg, flush and remove a group,
 - copy, move, find, ls, count rule or group, has capability.
- Service API methods:
 - load, remove, activate, disable, list policies,
 - add, remove, list modules.

Key benefits

- Programming interface to access firewall devices firewall capabilities,
 - Dynamic configuration,
 - Fine grain control of the configuration,
 - Easy to combine and use with high level programming language,
- Possibility to control different devices via same interface, use right device for right task,
- Extensibility of firewall capabilities via module loading,
- It is easy to support additional devices (abstract/device layer, OO paradigm).

Outline:



Requirements

- Dynamic configuration through programmable interfaces (append, insert, delete modify, search, copy, etc. the response actions),
- Logical grouping of the responses,
- Support for multiple firewall device and access protocol types,
- Response verification,
- Loading of arbitrary modules,
- Policy language and policy engine,
- Safety, security and performance.

Implementation and results

- Implemented in Java, abstract and device level, remote interface exported over RMI, can be any other,
- Support five different devices: Linux, Cisco router and firewall, two different hardware platforms, accessed through unified classifier API, support both integrated and in-line classifiers,
- Performance of the rule manipulation varies from few hundreds (shell) to few tens (ssh) per second,
- Modules are managed via Oscar OSGI framework,
- PMA implementation is based on Ponder2 policy language and engine (ICL).

Open Issues

- Distributed aspect needs to be further developed (control, responses, logging, testing, debugging),
- Support more devices and modules,
- Export firewall devices monitoring information,
- More responses, composed responses?