Firewall Technologies

Firewall Technologies

- Firewall is a software/hardware system that acts as a barrier between
- trusted and untrusted networks

- Firewall Technologies
  - Packet Filtering
    - Stateless and hard to manage
  - Stateful Filtering
    - Session information is maintained in a connection/state table
    - Return traffic is automatically allowed
    - Capable of detecting certain attacks, e.g. DoS

Firewall Technologies

- Firewall Technologies
  - Application Layer Gateways
    - Designed to control certain type of traffic, eg. HTTP
    - Full protocol visibility („deep packet inspection”)
    - Examples : Cisco Web and Email Security Appliances
  - Transparent
    - Operates at Layer 2 („bump in a wire”)
  - Next Generation
    - A combination of an advanced stateful firewall and IPS
    - Prevents threats
    - Example : ASA with FirePOWER services

Access Lists

Access Lists

- Access List (ACL) is an example of a simple Packet Filtering technology
  - Mainly used to control traffic to/through the device
    - Often used by other features, such as Route Filtering, QoS or VPN
  - Consists of Access Control Entries (ACEs) processed sequentially in a top-down fashion
    - Each numbered entry consists of an action („permit" or „deny") and conditions
      - Source/Destination IP addresses, port numbers, TCP flags and more
    - When a match is found the evaluation stops
    - Ends with an implicit deny catch-all entry („deny any any")
  - Standard ACL (1-99 *or named) can only match source IPs
    - Useless for traffic filtering (primarily used for route filtering or VTY Access Control)
  - Extended ACL (100-199* or named) allows to match any supported condition

Access Lists

- Traffic-filtering ACLs must be applied to activate them
  - There can be only one ACL applied on an interface per direction („in" or „out")
    - An outbound ACL does not affect the traffic generated by the device
- IOS vs ASA
  - ACL uses wildcard (IOS) or regular (ASA) network masks
  - Inbound ACL controls transit (ASA) or to & through the box (IOS) traffic
- IPv6 Access Lists are always Extended
  - Work like IPv4 counterpart
  - Allow to match version 6 specific fields, like Extension Headers
  - Implicit deny does not affect Neighbor Discovery traffic but explicit does

Access Lists

- Configuration (IPv4)
  - On IOS define with **access-list *nr*** or **ip access-list [*nr*|*name*]**
  - On the ASA use **access-list [*nr*|*name*]**
  - Activate on an interface using **ip access-group** (IOS) or **access-group** (ASA)

- Configuration (IPv6)
  - Define with **ipv6 access-list *name***
  - Activate using **ipv6 traffic-filter** (IOS) or **access-group** (ASA)

Zone-Based Firewall

Zone-Based Firewall

- Zone-Based Firewall (ZFW) is the newest implementation of a stateful
- firewall on IOS
  - Much more granular and advanced than older CBAC
    - Many settings can be tuned, including application layer inspecion engines
  - Uses a concept of security zones, similar to the ASA
    - A zone consists of at least one physical/logical interface of the router
      - A pre-defined zone „self" is automatically associated with all router's ZFW interfaces
    - A pair of zones (aka „zone-pair") is used to define traffic to act on
      - Source zone is where the traffic originates from, destination is where it goes (direction does matter)

Zone-Based Firewall

- Default Traffic Processing
  - Intra-zone communication (source zone = destination zone) is allowed
  - Inter-zone traffic (source zone != destination zone) is blocked
    - Exception : traffic destined to/ sourced from zone „self" is allowed
  - Zone to no-zone (and vice versa) is always dropped
    - No-zone refers to an interface that was not assigned to any zone

- Intra-zone and inter-zone default traffic processing behavior can be
- changed by associating a zone-pair with a policy

Zone-Based Firewall

- Configuration
  - Traffic classification
  - Policy configuration
  - Policy activation

- Classification (class-map type inspect)
  - Condition/criteria types
    - Access-list (**match access-group**)
    - Protocol (**match protocol**)
    - Existing class (**match class-map**)
  - Condition/criteria processing logic (**match-all** vs **match-any**)

Zone-Based Firewall

- Policy Configuration (policy-map type inspect)
  - Classes are processed top-down like an ACL
    - An implicit class-default matches all remaining packets and by default drops them
  - Policy actions
    - Content filtering for HTTP[S] (**urlfilter**)
    - Drop (**drop**) or drop & log (**drop log**)
    - Rate-limit (**police**)
    - One-way allow (**pass**)
    - Stateful inspection (**inspect *parameter_map***)
      - Unless **match protocol** was used in a class, relies on PAM to find the inspection engine which results in unoptimized lookups

Zone-Based Firewall

- Port-to-Application Mapping (PAM)
  - A preconfigured database of applications/protocols and their default transport
    - For example HTTP -> TCP 80, IKE -> UDP 500
  - Existing entries can be updated with **[ip|ipv6] port-map [list *acl_nr*]**
    - Useful when non-standard ports are needed, e.g. **ip port-map http port tcp 8080**
    - The **list** argument is required to change system-defined mappings
  - New entries can be added for custom applications/protocols but their name must start with a prefix „**user-**", for example **ip port-map user-IKEv3**

Zone-Based Firewall

- Parameter Map
  - Controls common inspection options, such as timeouts or session parameters
  - The „default" map is used every time **inspect** is configured with no options
  - A custom map can be configured with **parameter-map type inspect**
    - Activated in a policy with **inspect** ***map_name***
    - All undefined settings are inherited from the „default" map
  - The „global" map allows to enable logging of packets dropped by the firewall due to reasons other than your policy **drop** action
- Parameter Maps other than „inspect" can be also configured
  - For example to control URL Filtering settings

Zone-Based Firewall

- Policy Activation
  - Create zones (**zone security**)
  - Define required zone-pairs (**zone-pair security**)
    - Attach your policy (**service-policy type inspect**)
  - Associate interfaces with zones (**zone-member security**)

- Application Layer inspection tuning
  - Create L7 class-map (**class-map type inspect [http|smtp|...]**)
  - Create L7 policy-map (**policy-map type inspect [http|smtp|...]**)
  - Nest L7 child policy in the L3/4 parent (**inspect** + **service-policy type inspect *L7polname***)

ASA Fundamentals

ASA Fundamentals

- Cisco ASA is an advanced next-generation firewall
  - Powerful stateful filtering and application-layer inspection capabilities
    - Session tracking, TCP Sequence Randomization, TCP Normalization and more
  - VPN gateway
    - IKEv1/IKEv2 L2L and IKEv1/IKEv2/SSL Remote Access
  - Next-generation IPS
    - ASA with FirePOWER, Advanced Malware Protection and Reputation URL Filtering
  - Virtualization
    - Contexts
  - High availability
    - Failover and Clustering

ASA Fundamentals

- ASA Models
  - The X-series of physical appliances (5506-X, 5508-X, 5512-X ... 5585-X)
    - Actual platform affects available bandwidth, inspection throughput, supported number of VPN peers and similar options
  - Virtualized platforms (ASAv)
    - ASAv5, ASAv10 and ASAv30
      - Delivers up to 100Mbps/1Gbps/2Gbps of througput, respectively
    - Does not support Clustering and multiple Contexts
    - Commonly used in Data Centers

ASA Fundamentals

- ASA Interfaces
  - Physical (**interface *physifname***)
    - Single port
  - Redundant (**interface redundant *nr***)
    - Two ports (active/standby)
  - EtherChannel (**interface port-channel *nr***)
    - Two or more ports (active/active)
  - Virtual (also known as Subinterfaces)
    - Traffic is logically separated at L2 by using VLAN tags
    - Corresponding switchport(s) must be configured as 802.1q trunk
    - Configure with **interface *name.nr***

ASA Fundamentals

- Interface Settings
  - IP address (**ip/ipv6 address [standby]**)
  - Security level (**security-level**)
    - Specifies how „trusted" a given interface is
    - Controls default filtering ASA's behavior
  - Interface name (**nameif**)
    - Default security level for „inside" is 100 and 0 for any other name
  - (Optional) VLAN tag (**vlan**)
    - Watch for Native VLAN and DTP
  - Activation (**no shut**)

ASA Fundamentals

- Default ASA Filtering Policy
  - Traffic originating on a higher security level interface (than the destination) is allowed
  - Traffic originating on a lower security level interface (than the destination) is blocked
    - Exceptions can be made with access-list
  - If two interfaces have the same security level, traffic is blocked
    - Change with **same-security-traffic permit inter-interface**
  - If incoming and outgoing interface is the same (Hairpinning/U-Turn), traffic is blocked
    - Change with **same-security-traffic permit intra-interface**
    - Useful in certain VPN scenarios
  - Traffic destined to the firewall (to-the-box) is dropped
    - Exceptions are ICMP to the local interface, DHCP and HTTPS to management port

ASA Fundamentals

- ASA routing is performed very similar to IOS
  - An exception is when packet matches an existing NAT translation
    - Then the translation slot itself determines egress interface, not a RIB lookup
      - Can be disabled by adding **route-lookup** to the NAT rule
  - In other cases longest match route from the RIB is used to find egress interface
    - Route recursion is performed for the next-hop(s) if necessary
  - Packet is switched, re-encapsulated and serialized onto the link

ASA Fundamentals

- Routing Configuration
  - Static route (**[ipv6] route** ***interface***)
    - Default route example : **route outside 0 0** ***next_hop_ip***
  - OSPFv2
    - Configure the process (**router ospf** ***process_id***)
    - Enable OSPF on interfaces (**network**)
  - OSPFv3
    - Configure the process (**ipv6 router ospf** ***process_id***)
    - Enable OSPF on interfaces (**ipv6 ospf** ***process_id*** **area** ***nr***)
  - EIGRP
    - Configure the process (**router eigrp** ***AS_nr***)
    - Enable EIGRP on interfaces (**network**)

ASA Management

ASA Management

- ASA can be managed through a console port or remotely

- Remote Management
  - In-band (any data interface)
  - Out-of-band (management port)
    - Does not allow traffic to go in/out the management network
    - By default **interface management** acts as a mangement port
    - Other interface can be selected with **management-only**
  - Supported methods include Telnet, SSH and HTTPS

ASA Management

- By default to-the-box traffic, including management packets, is blocked
  - Additional configuration is required so that the ASA starts listening for incoming packets
    - Management access is controled with **telnet**, **ssh** and **http**
    - Telnet can't be used to access the lowest security level interface (unless via VPN)

- Adaptive Security Device Manager (ASDM)
  - Java applet GUI for ASA configuration (connects over HTTPS)
  - Unless factory defaults are used, ASA must be configured for ASDM
    - Select an image (**asdm image**)
    - Enable HTTPS (**http server enable**) and allow access (**http**)
    - AAA is recommended for authentication (**aaa authentication http console**)

ASA Traffic Filtering

ASA Traffic Filtering

- The default ASA filtering policy can be changed with Access Lists
  - Commonly used to make exceptions to allow traffic from lower security level interfaces
  - Only extended ACLs can be used (**access-list** or **ipv6 access-list**)
  - Applied per-interface (**access-group ... interface**) or globally (**access-group ... global**)
  - Global ACL affects all incoming transit packets received on any interface
    - May affect the default „allow" for higher -> lower traffic
    - Explicit permits/denys of an interface ACL (if any) still take precedence

- Interface and Global ACLs are for transit traffic only
  - To-the-box traffic can be controlled by a Control Plane ACL
    - An ACL applied with **access-group ... control-plane**

ASA Traffic Filtering

- Objects
  - Reusable components acting as placeholders for certain values
    - IP addresses, subnets or ranges (**object network**)
    - Protocols and TCP/UDP port numbers (**object service**)
  - An object can only contain one element

- Object Groups
  - Like objects, but capable of storing multiple elements and/or other objects
  - Allow to group other data (e.g. icmp-types, users)
    - Configure with **object-group [protocol|network|icmp-type|service|user]**

ASA Network Address Translation (NAT)

Network Address Translation (NAT)

- NAT rewrites IP addresses (and possibly port numbers) in a packet
  - Typically to hide private IP addresses (RFC 1918)
  - Other applications include traffic redirection or overlapping subnet problems
  - Not a security tool

- NAT Types
  - Static (one-to-one, fixed pre-configured mapping)
  - Dynamic (one-to-one, new IP address is allocated dynamically from a pool)
  - PAT (many-to-one, source IP address and source port is changed dynamically)
  - Static PAT (many-to-one, address & port mapping is pre-configured)
  - Policy NAT (any condition-based translation)

ASA NAT

- ASA NAT implementation relies on two tables : Rules and XLATEs
  - NAT Rule describes the packet before and after the translation
    - When a translation occurs (original packet)
    - How it occurs (new/translated packet)
  - XLATEs
    - Stores current translations
    - Primarily used to de-translate (restore the original) packet

ASA NAT

- NAT on the ASA can be configured in Auto or Manual mode
  - Auto NAT is used to build simple translation rules
    - Supports source IP address translation only (with an optional source port)
    - Does not support Policy NAT or destination IP address translation
    - Configured within a network object (**object network**) with **nat**
  - Manual NAT is suited for complex translations of source/destination IP addresses and source/destination port numbers
    - Commonly used for Policy NAT or Twice NAT (source & destination IP translation)
    - Implemented through global configuration mode **nat**
      - Operates on objects and object-groups

ASA NAT

- Rule Processing
  - Overlapping translation rules are often configured
    - Each NAT rule is placed in one of three sections (sections are evaluated top-down)
      - Rules within each section are checked one by one, until first match is found
  - Section 1
    - Manual NAT rules, user-sequenced
  - Section 2
    - Auto NAT rules, sequenced dynamically based on ASA's internal algorithm
    - Prefers static rules over dynamic
  - Section 3
    - Manual NAT rules entered with „**after-auto**" option, user-sequenced

ASA Advanced

ASA Modes

- ASA Modes of Operation
  - Firewall Mode controls ASA's forwarding behavior
  - Context Mode enables/disables firewall virtualization
  - Both Modes affect features supported on the ASA

- Firewall Mode : Routed (default)
  - ASA acts as a L3 hop, each interface connects to a different L3 subnet
  - All regular features are supported
  - Might not be easy to insert a firewall into existing network

ASA Modes

- Firewall Mode : Transparent
  - ASA acts as a L2 switch, bridged interfaces are grouped and put into one L3 network
    - Multiple bridge-groups can be configured that cannot communicate to each other
  - Unsupported features : VPNs, dynamic routing protocols, multicast routing and QoS
  - Advantages
    - Can be easily placed into the network without having to re-address existing devices
    - Allows to control non-IP packets
  - Configure with **firewall transparent**
    - Configure a BVI interface for management (**interface bvi *nr***)
    - Associate interfaces with a bridge-group (**bridge-group *nr***)

ASA Modes

- Context Mode : Single (default)
  - No virtualization : one firewall and one policy
  - Supports all regular features

- Context Mode : Multiple
  - Enables virtualization
    - Multiple logical firewall instances can co-exist on a single physical unit
    - Each of the virtual firewalls is configured with a set interfaces and policy
  - Does not support certain VPN protocols and features, QoS, multicast routing and some routing protocols
  - Configure with **mode multiple** and then **context**

ASA High Availability

- Failover
  - Requires two physical firewalls
  - Works in one of two modes : Active-Standby or Active-Active
    - Active-Standby
      - Only the active unit forwards traffic
    - Active-Active
      - Both firewalls can actively forward traffic
      - Available in multiple context mode only
  - Stateful failover can be configured regardless of the failover mode

ASA High Availability

- Clustering
  - Combines multiple ASAs into a single unit
    - Results in increased throughput and redundancy
  - Upstream and downstream routers are responsible for traffic load-balancing
    - Spanned EtherChannel
    - Policy-Based Routing
    - Equal-Cost Multi-Path (ECMP) routing

Modular Policy Framework (MPF)

- MPF configuration rules control many of the ASA's features
  - Inspection engines, TCP Normalization, QoS and more
  - Work on traffic permited by the firewall policy (access rules, default policy)
  - Managed by MQC-like framework
    - Classification (**class-map**)
    - Policy Configuration (**policy-map**)
    - Policy Activation (**service-policy**)

Modular Policy Framework (MPF)

- Classification
  - All traffic (**match any**)
  - Access-list (**match access-list**)
  - TCP/UDP ports (**match [tcp|udp]**)
  - ToS (**match [dscp|precedence]**)
  - RTP (**match rtp**)
  - Tunnel Group (**match tunnel-group**)
  - Default Protocols (**match default-inspection-traffic**)
    - Special condition used in a Default MPF Policy to match multiple protocols in a single class

Modular Policy Framework (MPF)

- Policy Configuration
  - Classes are evaluated top-down and in certain cases more than one class can be a match
    - MPF Policies are processed in a complex way including internal ASA rules
    - Using one policy with non-overlapping classes results in MQC-like processing
  - Class-default match otherwise unclassified packets
- Policy Feature Types (Actions)
    - Inspection engines (**inspect *protocol [L7_policy_name]***)
    - Connection settings and Timeouts (**set connection**)
    - TCP Normalization and State Bypass (**set connection advanced-options**)
    - QoS (**police, priority, shape**)
    - Legacy IPS (**ips**) and FirePOWER (**sfr**)

Modular Policy Framework (MPF)

- Policy Activation
  - Per-interface (**service-policy interface**)
  - Globally (**service-policy global**)
    - Enables the policy for all interfaces
  - Interface-level policies take precedence for overlapping classes

- Default MPF Policy (global_policy)
  - Enabled globally
  - Has one class (inspection_default) matching default ports (**default-inspection-traffic**)
    - This special class allows to use multiple inspection engines
  - Enables inspection for several protocols, including DNS, FTP, TFTP, ESMTP and more

Modular Policy Framework (MPF)

- Inspection Overview
  - Generic Inspection
    - TCP and UDP packets are inspected by default, ICMP is not
  - Application Layer Inspection
    - Each supported protocol is inspected differently
      - FTP -> secondary channel opening
      - HTTP -> protocol conformance
      - IPv6 -> Extension Headers
    - The default behavior of many application-layer inspection engines can be tuned through L7 Policies (**policy-map type inspect *protocol***)