Introduction to Cisco ASA

Andrew Ossipov
Technical Marketing Engineer
Cisco Security Business Group

Agenda

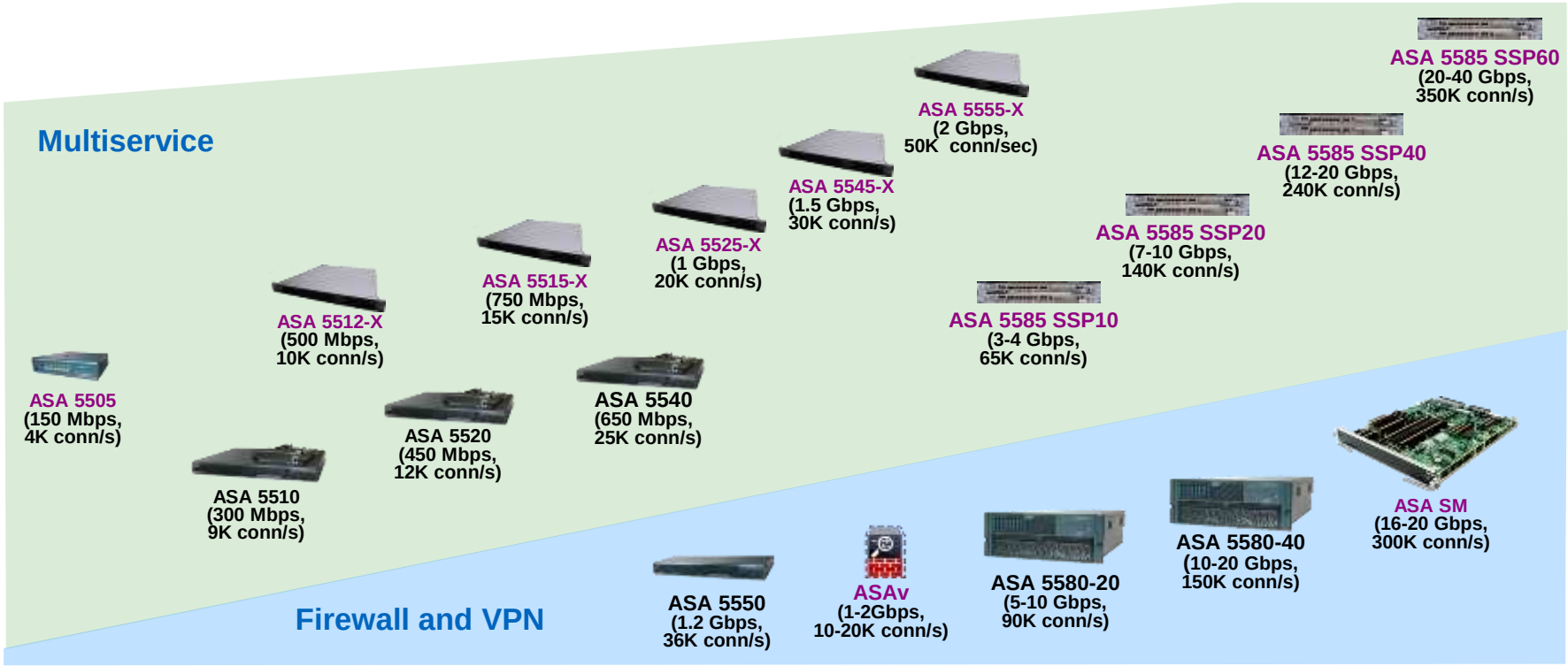
- ASA Hardware and Software
- Configuration Basics
- Network Address Translation (NAT)
- Access Control Lists (ACL)
- Packet Flow
- Troubleshooting Tools
- Troubleshooting Case Study
- Q&A
- Useful Links

ASA Hardware and Software

ASA Functionality

- Adaptive Security (more-than-an-)Appliance
- Stateful architecture is about **flows** or **connections**, not packets
  - Most effective with TCP, UDP, and ICMP
  - TCP is the main reason for deploying a stateful firewall
- A multitude of benefits beyond traditional security
  - High-speed NAT
  - Identity-based Access Control
  - High Availability and Scalability
  - Application Inspection
  - NGIPS with FirePOWER services

ASA Firewall Family



ASA5500-X Appliances

ASA5512-X
ASA5515-X
ASA5525-X

Hard Disk Drive (CX/SFR only)

Integrated Power Supply

USB 2.0 ports for external flash memory

Copper 1GE interfaces (6 or 8)

ASA5545-X
ASA5555-X

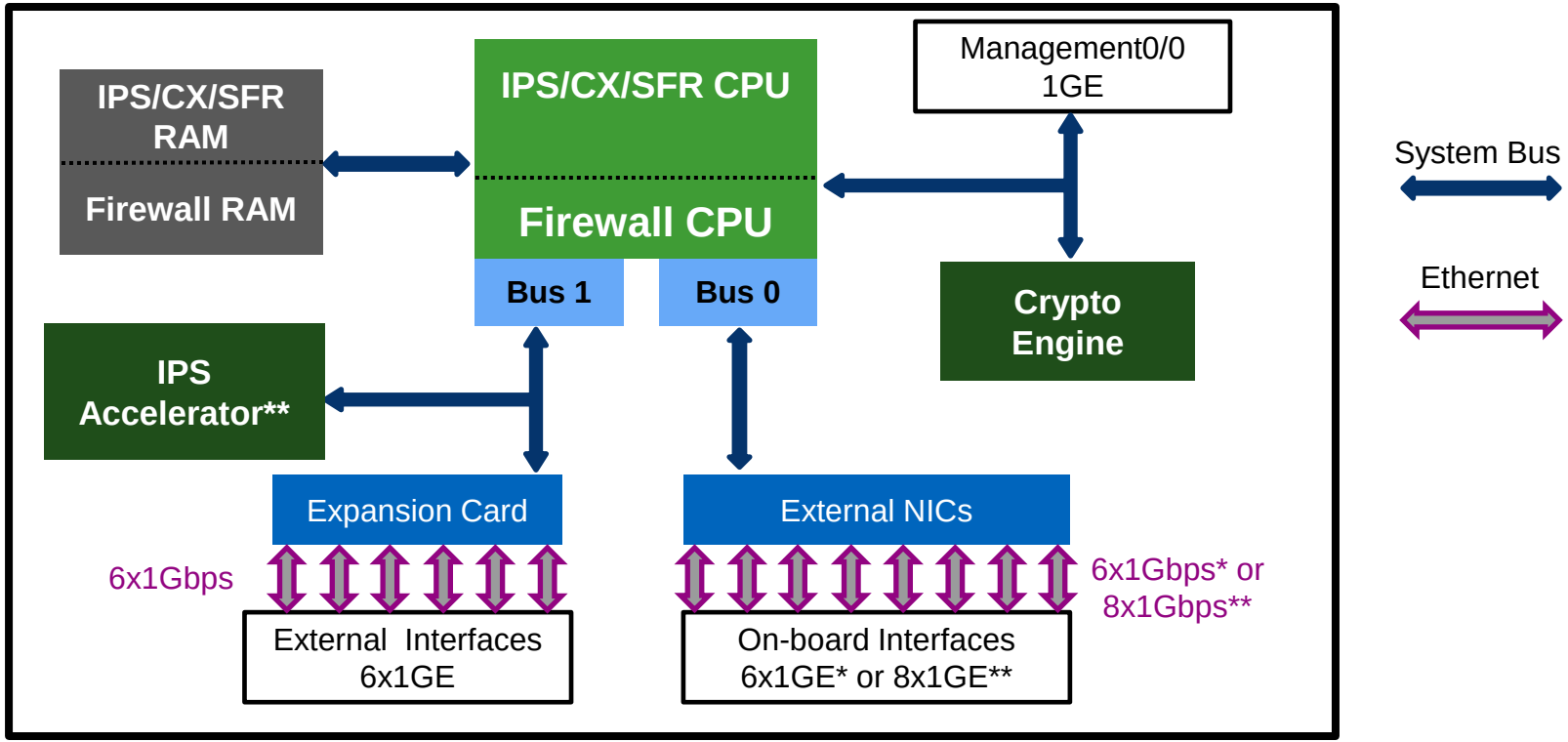
Redundant Hard Disk Drives (CX/SFR only)

Dual independent Power Supplies

1GE Copper or Fiber interface expansion slot (6)

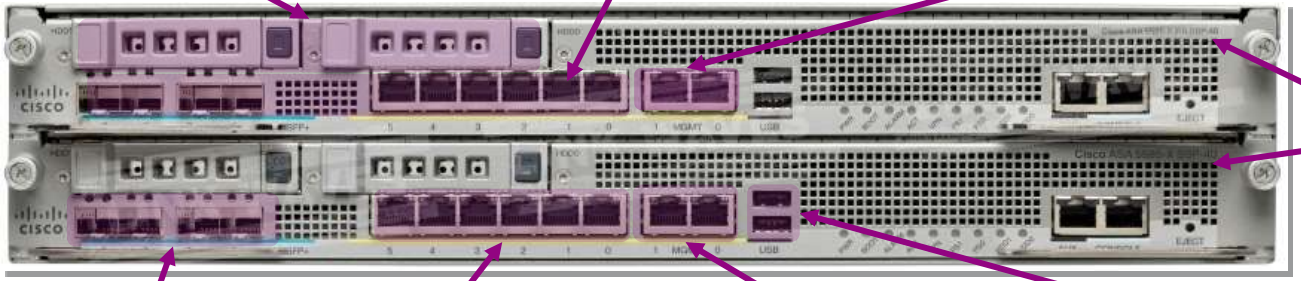
ASA/IPS Management interface

ASA 5500-X Block Diagram



*ASA5512-X and ASA5515-X
** ASA5525-X and higher

ASA5585-X Chassis and SSP



Hard Disk Drives (CX/SFR only)

Firewall expansion interfaces (IPS/CX/SFR)

CX/IPS/SFR Management 1GE interfaces

Dual independent Firewall SSPs or Firewall + IPS or Firewall + CX or Firewall + SFR

Fiber 10GE interfaces (2 or 4)

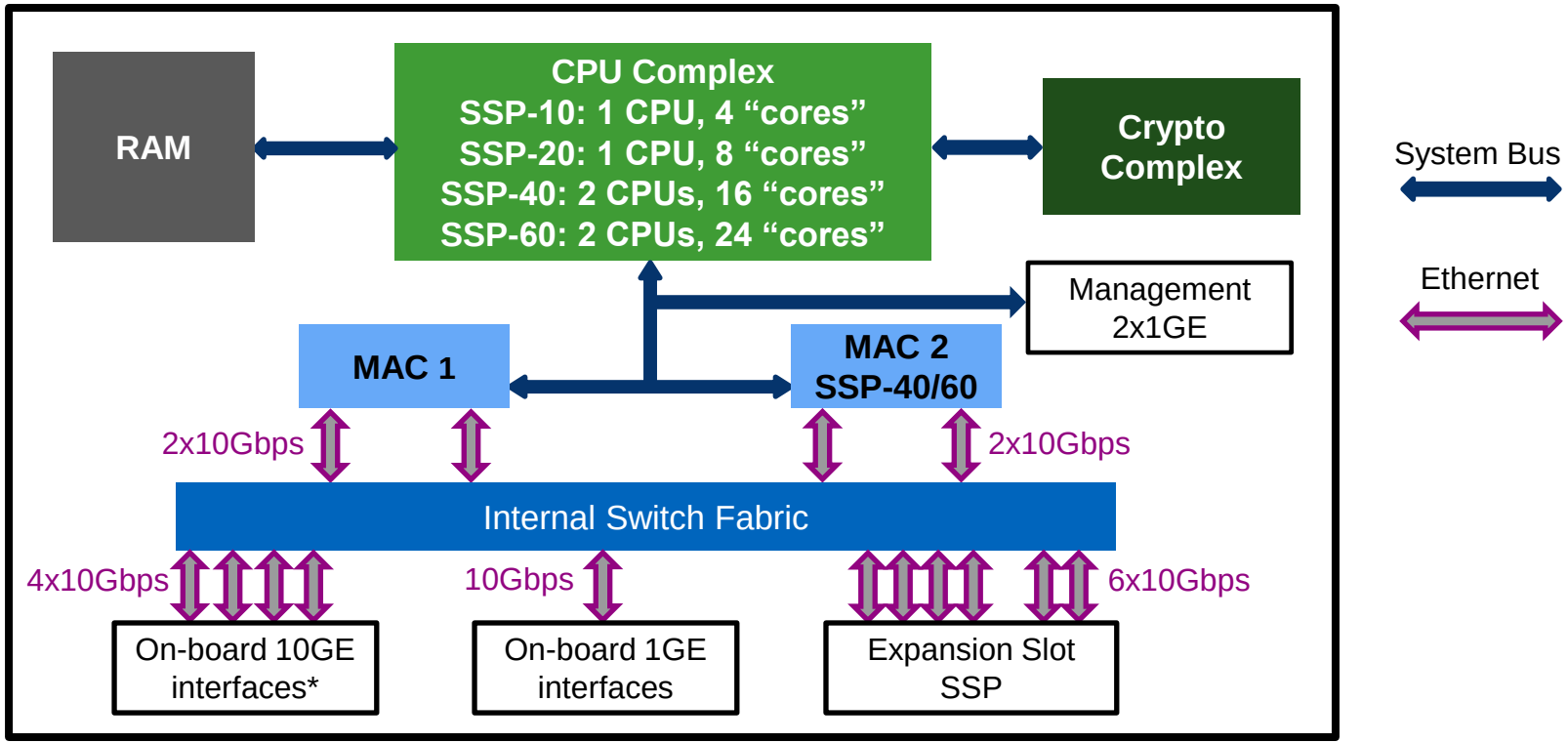
Copper 1GE interfaces (6 or 8)

ASA Management 1GE interfaces

Reserved USB ports

Dual independent Power Supplies or Power Supply + Fan

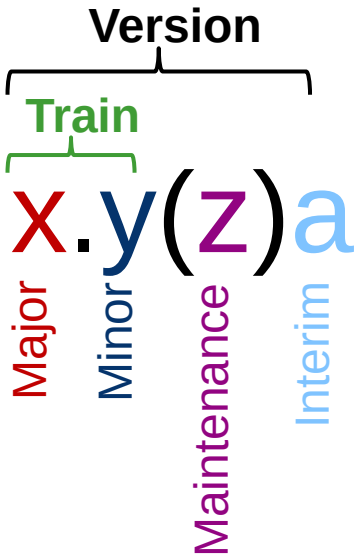
ASA5585-X Block Diagram



*2 on SSP-10/20 and 4 on SSP-40/60

ASA Software Builds

- New trains introduce new features
- Maintenance images undergo the most testing
  - Concentrate on bug fixes, avoid new features
- Interim images are usually internal
  - Limited testing, so only provided for specific problems
  - Some are posted on cisco.com after more testing




```
asa# show version
[…]
Cisco Adaptive Security Appliance Software Version 8.4(4)5
```

Version: 8.4(4)5
Train: 8.4
5th Interim of 8.4(4)

New Software Release Model

- Predictability and serviceability are the main drivers
- **Even** minor releases retain the original feature set for life
  - 9.**0**(x), 9.**2**(x), 9.**4**(x), …
  - Feature Stability train
  - Maintenance releases only deliver bug fixes
  - Longer life cycle
- **Odd** minor releases add features in maintenance images
  - 9.**1**(x), 9.**3**(x), 9.**5**(x), …
  - Feature Velocity train
  - Shorter life cycle to transform into the next even minor release

Configuration Basics

Interfaces

- Physical interfaces are externally visible Ethernet ports
- Virtual interfaces may share or bundle physical interfaces
  - IEEE 802.1Q VLAN trunks

```
interface GigabitEthernet0/1.10
 vlan 10
```

  - Less overhead with a Redundant interface, but more capacity with Etherchannel

```
interface Redundant 1
 member-interface GigabitEthernet0/0
 member-interface GigabitEthernet0/1
```

```
interface GigabitEthernet0/0
 channel-group 1 mode active

interface GigabitEthernet0/1
 channel-group 1 mode active

interface Port-Channel 1
```

Logical Interfaces

- All policies and operations use logical (named) interfaces

```
interface GigabitEthernet0/0.10
 vlan 10
 nameif outside
 security-level 0
 ip address 172.16.164.120 255.255.255.224 standby 172.16.164.121
```

- Security levels define inter-interface traffic policies
  - Most trusted security level is 100 (“inside”), least trusted is 0 (“outside”)
  - The only thing that matters is **relativity**
  - Traffic from higher- to lower-security interface is allowed by default
  - Traffic from lower- to higher-security interface requires an explicit policy
  - **same-security-traffic permit inter-interface**

Connection and Xlate Tables

- **Conn**(ection) table stores the state of every single active flow

```
asa# show conn
1 in use, 48 most used
TCP outside  172.16.164.216:5620 inside  192.168.1.150:50141, idle 0:00:00, bytes 0, flags saA
```

  - Every incoming packet is checked against the table
  - Biggest memory consumer (maximum count is limited by platform)

- **Xlate** table stores active NAT mappings

```
asa(config)# show xlate
1 in use, 14 most used
Flags: D - DNS, e - extended, I - identity, i - dynamic, r - portmap,
       s - static, T - twice, N - net-to-net
NAT from inside:192.168.1.150 to outside:172.16.164.116
    flags sT idle 0:00:01 timeout 0:00:00
```

  - Independent of the conn table
  - Maximum size is only limited by available memory

Packet Processing

- Interface Controller moves packets from Ethernet to memory
- Data Path CPU process checks all inbound packets **sequentially**
  - Multiple parallel threads on all modern platforms
  - Stateful checks are applied to every single packet
  - Fastpath, Slowpath, Control Plane
- New connection requests are directed to **Slowpath**
  - Access Control List check, NAT xlate creation, conn creation, logging
- Existing connections are processed in **Fastpath**
  - Bypass ACL check, find egress interface, apply NAT, transmit packet
- **Control Plane** performs Application Inspection and management

Security Policy Basics

- NAT policies define address translation
  - Independent source and destination NAT policy for every connection
  - Should **not** be used to control access
- ACLs permit or deny new **connections**
- NAT and ACLs are based on IP addresses and TCP/UDP ports
  - Logical abstraction is needed for better usability

Unified Objects

- Useful to refer to IP addresses, subnets, or ranges by name

```
object network INSIDE_NETWORK
 subnet 192.168.0.0 255.255.0.0
object network PAYROLL
 range 2001:DB8::10 2001:DB8::15
```

- Can be used with TCP, UDP, and ICMP as well

```
object service MSSQL_ADMIN
 service tcp destination eq 1434

object service ICMP_ADMIN_PROHIBITED
 service icmp unreachable 9

object service RTP_PORTS
 service udp source range 16384 32767 destination range 16384 32767
```

Match only the destination TCP port

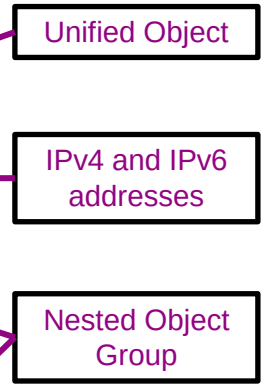
Both source and destination UDP ports must match

Object Groups

- Tie multiple **similar** objects into a single policy entity
  - Network addresses, transport protocols, identity attributes
  - Significantly simplify policy management

```
object-group network INSIDE_NETWORKS
 network-object object ACCOUNTING
 network-object 2001:DB8::/64
 network-object 192.168.2.0 255.255.255.0
 group-object BRANCH_NETWORKS
```

```
object-group network BRANCH_NETWORKS
 network-object 172.16.1.0 255.255.255.0
 network-object 172.16.2.0 255.255.255.0
```



Network Address Translation (NAT)

NAT Power of ASA

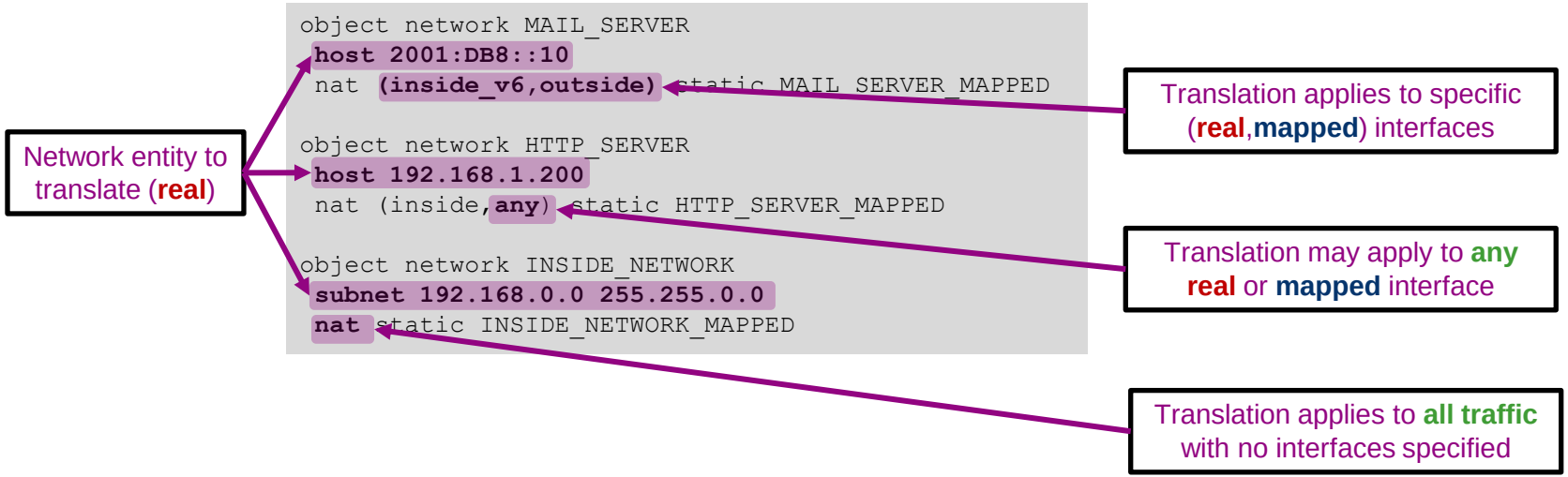
- Configured network IP is **real**, translated is **mapped**
- **Predictable** bidirectional one-to-one IP translation with **Static NAT**
  - **Static PAT** at TCP/UDP/ICMP level with limited IP space
- Outbound connectivity with many-to-few **Dynamic NAT**
  - One-to-one Dynamic NAT at IP level to preserve full range of ports
  - **Dynamic PAT** at TCP/UDP/ICMP level to maximize pool efficiency

Legacy NAT Challenges

- Complicated order of operation for many varieties of NAT
  - Identity, Exemption, Static, Dynamic, Policy, and so on
- Poor scalability of Policy NAT and performance impact on changes
- Inability to apply source and destination translation simultaneously
  - Possible through separate policies, but had to be creative
- **ASA 8.3** introduced new NAT paradigm
  - Just two varieties: **Network Object** and **Twice NAT**
  - Way more powerful, but may seem complicated at first

Network Object NAT

- Simplest form of defining translation policy for Unified Objects
  - Only **one** translation rule per object
  - Applies to **all** traffic to or from the object, use interfaces names to limit scope



Dynamic NAT and PAT

- Dynamic NAT maps hosts to a pool of IP addresses (one-to-one)
  - Temporal by nature and unpredictable mapping, so **not** for inbound connections
  - Once the mapped pool is exhausted, new translations fail (first come, first serve)

```
object network NAT_POOL
 range 198.51.100.100 198.51.100.199
object network INSIDE_NETWORK
 subnet 192.168.0.0 255.255.0.0
 nat (inside,outside) dynamic NAT_POOL
```

Mapped pool can be a Unified Object or Object Group

- Dynamic PAT can map multiple real hosts to one IP address
  - Interface PAT can be used to “back up” a dynamic NAT pool

```
object network INSIDE_NETWORK
 subnet 192.168.0.0 255.255.0.0
 nat (inside,outside) dynamic NAT_POOL interface
```

When NAT pool is exhausted, apply PAT using egress interface IP

Dynamic PAT Pools

- Multiple addresses in the mapped Object imply NAT, **not** PAT

```
object network PAT_ADDRESS
 host 198.51.100.90
object network INSIDE_NETWORK
 subnet 192.168.0.0 255.255.0.0
 nat (inside,outside) dynamic PAT_ADDRESS
```

Single address is required for PAT in this configuration (or specify the mapped IP in-line)

  - Old way to define a PAT pool is an Object Group with multiple single IP objects
- **ASA 8.4(3)** introduces an easier way to define PAT pools
  - By default, exhaust ports for each mapped address before moving to next one
  - With Round Robin, use next pool address to pick a port for each **new** real host

```
object network PAT_POOL
 range 198.51.100.200 198.51.100.240
object network INSIDE_NETWORK
 subnet 192.168.0.0 255.255.0.0
 nat (inside,outside) dynamic pat-pool PAT_POOL round-robin
```

Use **pat-pool** keyword to differentiate NAT and PAT pools

Move to next pool IP for each connection from a new host

Static NAT and PAT

- "Always on" one-to-one bidirectional IP mapping with Static NAT
  - Sizes of real and mapped objects must match

```
object network STATIC_MAPPING
 range 198.51.100.50 198.51.100.60
object network INSIDE_SERVERS
 range 192.168.1.35 192.168.1.45
 nat (inside,outside) static STATIC_MAPPING
```

192.168.1.35↔198.51.100.50
192.168.1.36↔198.51.100.51
…
192.168.1.45↔198.51.100.60

- Static PAT hairpins a certain port with limited mapped IP space
  - Also useful for changing default service ports for inbound connections

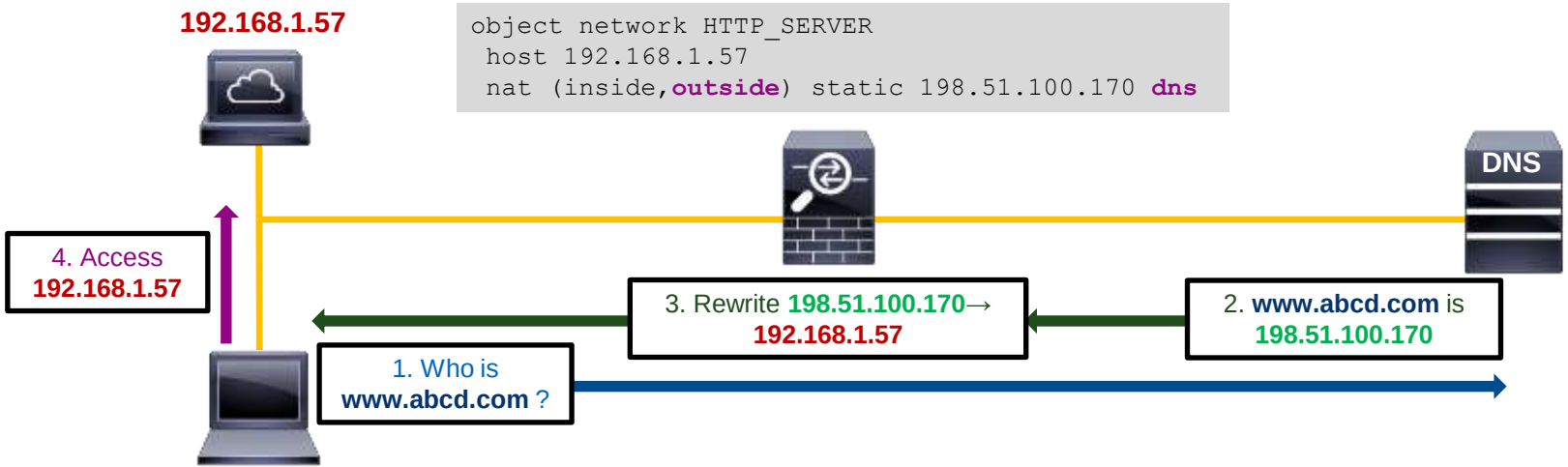
```
object network INSIDE_SMTP_SERVER
 host 192.168.1.56
 nat (inside,outside) static 198.51.100.75 service tcp 25 25

object network INSIDE_WEB_SERVER
 host 192.168.1.55
 nat (inside,outside) static 198.51.100.75 service tcp 80 8080
```

192.168.1.56/25↔198.51.100.75/25

192.168.1.55/80↔198.51.100.75/8080

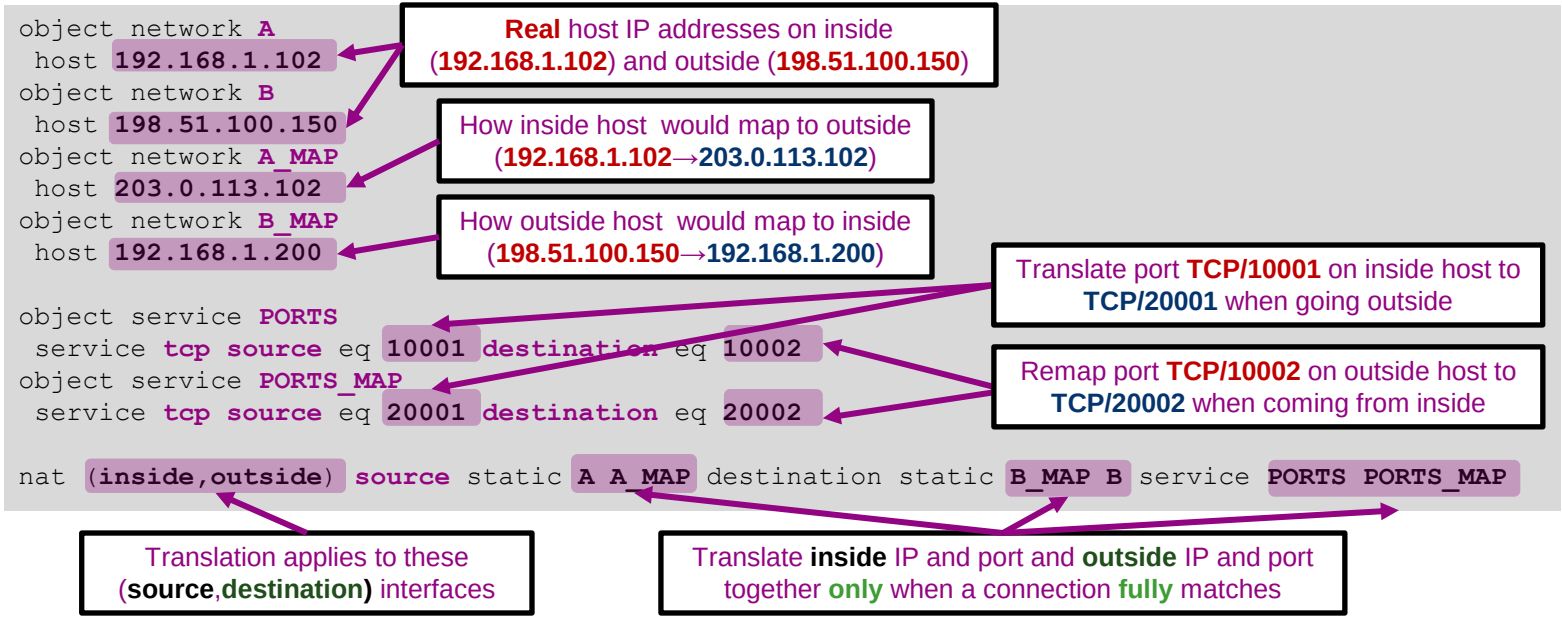
DNS Doctoring

- ASA can rewrite IP in transit DNS responses per NAT rules
  - Modifies A (IPv4) or AAAA (IPv6) record when crossing **mapped** interface
  - Should **only** be used with Static NAT



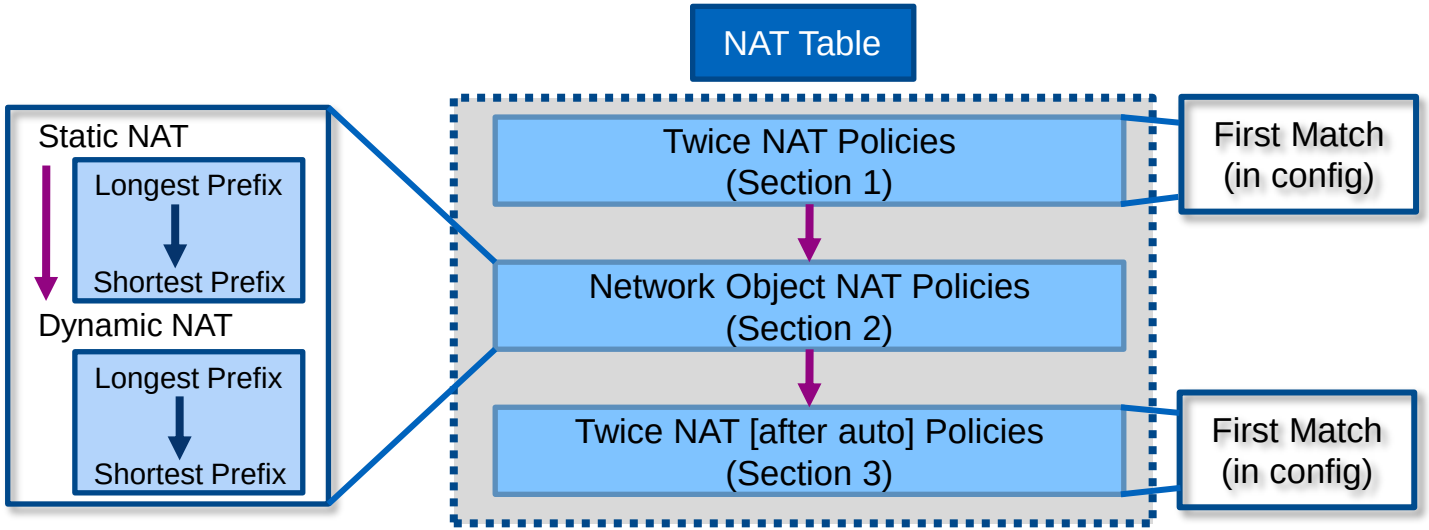
Twice NAT

- Match and translate packets on source and destination **together**
  - Similar to Network Object NAT, but cannot use in-line IP or DNS Doctoring
  - A dynamic translation can **only** pair with a static one



NAT Order of Operation In ASA 8.3+

- The ASA configuration is compiled into the NAT table
  - Twice NAT rules always match and translate both source and destination
  - Network object NAT translates destination **first**, then source (separate rules)
- The **show nat** command will display the NAT table in order



Access Control Lists (ACLs)

ACL Basics

- Extended ACLs must be used for most features
  - New connection establishment, traffic classification, and so on
  - Standard ACLs can be used by some features (Route Maps, Capture)
- All ACL checks apply to new connections **only**
  - **Real** IP addresses should be used in **ASA 8.3** and later
  - Per-interface ACLs apply at ingress or egress or both first
  - Global ACL applies ingress to all interfaces after per-interface ACL

Unified ACLs

- Legacy software used separate IPv4 and IPv6 interface ACLs

```
access-list INSIDE_IPV4 extended permit ip host 10.1.1.1 any
ipv6 access-list INSIDE_IPV6 permit ip host 2001:DB8:1 any
access-group INSIDE_IPV4 in interface inside
access-group INSIDE_IPV6 in interface inside
```

"Any" depends on the ACL type

- **ASA 9.0** uses a single ACL for all IPv4 and IPv6

```
access-list IN extended permit ip host 10.1.1.1 any4
access-list IN extended permit ip host 2001::1 any6
access-list IN extended permit ip host 10.1.1.1 host 2001:DB8::10
access-list IN extended permit ip any any
```

Any IPv4

Any IPv6

Mixed IPv4 and IPv6 (Need NAT)

Any IPv4 or IPv6

- Configuration migration from earlier releases
  - Dual interface ACLs are merged
  - Contextual **any** conversion applies to **ACLs only**

ACE Explosion with Object Groups

- All configured ACLs are expanded before programming

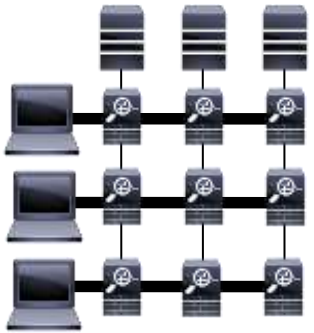
```
access-list IN permit tcp object-group INSIDE object-group DMZ_SERVERS object-group TCP_SERVICES
```

10 source IP addresses × 21 destination IP addresses × 33 TCP ports = **6930 rules**

- Nested Object Groups magnify the impact
  - Add a new source Object Group with 25 additional objects
  - Result: (10+25) x 21 x 33 = **24,255 rules** (ACEs)
- ACL Optimization prevents the Object Group expansion
  - Significant reduction in memory utilization, not so much on CPU

```
asa(config)# object-group-search access-control
```

- Cisco Security Manager (CSM) offers many ACL optimization tools



Identity Based Access Control

- Outbound access is difficult to control with IP-based ACLs
  - Inside users move around and IP addresses change all the time
- **ASA 8.4(2)+** can learn IP↔user mapping from Active Directory
  - ASA resolves user- and group-based ACL entries to IP addresses
  - External **AD Agent** or **Context Directory Agent** to interface between ASA and AD
  - Pushes LOCAL domain mappings for VPN and Cut-Through Proxy users
- Policy enforcement with Security Group Tags (SGTs) in **ASA 9.0+**
  - Devices map IP↔SGT with Security tag eXchange Protocol (SXP)
  - Unique 16-bit value assigned to a certain role (SG Name)
  - Authenticated and mapped at edge switch, enforced on ASA in transit
  - Abstraction from IP address or specific identity schemes

Packet Flow

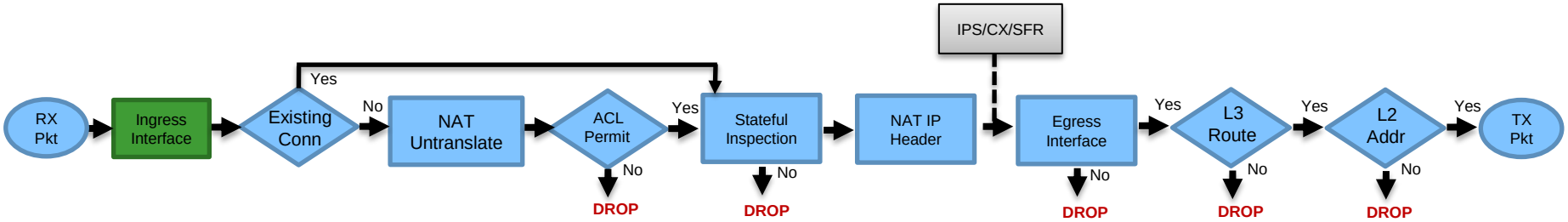
Understanding Packet Flow

- To effectively troubleshoot a connectivity problem, one must first understand the packet path through the network
- Attempt to isolate the problem down to a single device
- Then perform a systematic walk of the packet path through the device to determine where the problem could be
- For problems relating to the Cisco ASA, always
  - Determine the flow: Protocol, Source IP, Destination IP, Source Port, Destination Port
  - Determine the logical (named) interfaces through which the flow passes

```
TCP outside  172.16.164.216:5620 inside  192.168.1.150:50141, idle 0:00:00, bytes 0, flags saA
```

All firewall connectivity issues can be simplified to two interfaces (ingress and egress) and the policies tied to both

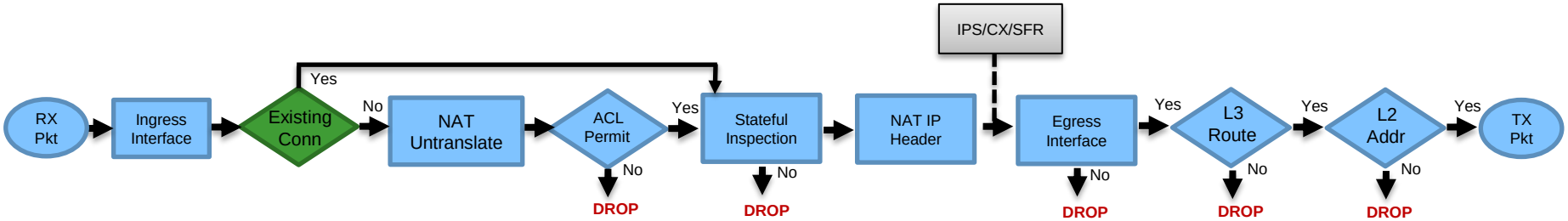
Packet Processing: Ingress Interface



- Packet arrives on ingress interface
- Input counters incremented by NIC and periodically retrieved by CPU
- Software input queue (RX ring) is an indicator of packet load
- **Overrun** counter indicates packet drops (usually packet bursts)

```
asa# show interface outside
Interface GigabitEthernet0/3 "outside", is up, line protocol is up
  Hardware is i82546GB rev03, BW 1000 Mbps, DLY 10 usec
        Auto-Duplex(Full-duplex), Auto-Speed(1000 Mbps)
        Input flow control is unsupported, output flow control is off
        MAC address 0026.0b31.36d5, MTU 1500
        IP address 148.167.254.24, subnet mask 255.255.255.128
        54365986 packets input, 19026041545 bytes, 0 no buffer
        Received 158602 broadcasts, 0 runts, 0 giants
        0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
[…]
        input queue (blocks free curr/low): hardware (255/230)
        output queue (blocks free curr/low): hardware (254/65)
```

Packet Processing: Locate Connection



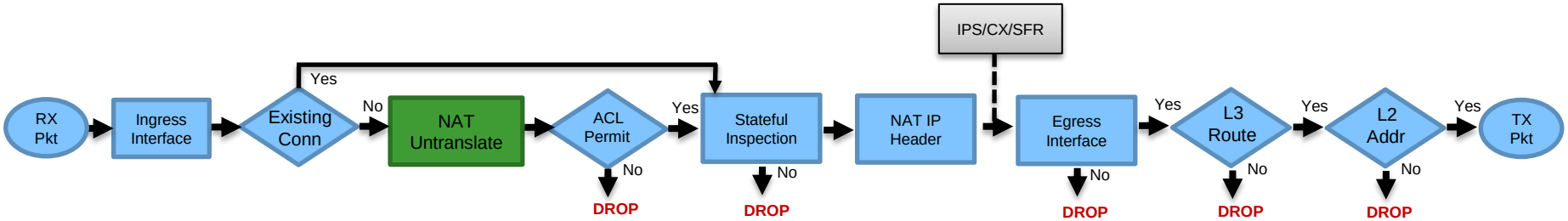
- Check first for existing connection in conn table
- If conn entry exists, bypass ACL check and process in Fastpath

```
asa# show conn
TCP out 198.133.219.25:80 in 10.1.1.9:11030 idle 0:00:04 Bytes 1293 flags UIO
```

- If no existing connection
  - TCP SYN or UDP packet, pass to ACL and other policy checks in Session Manager
  - TCP non-SYN packet, drop and log

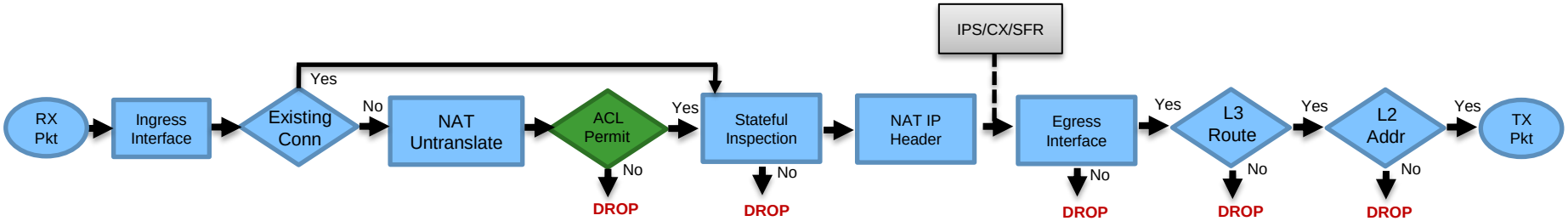
```
ASA-6-106015: Deny TCP (no connection) from 10.1.1.9/11031 to 198.133.219.25/80 flags PSH ACK  on
interface inside
```

Packet Processing: NAT Un-Translate



- Incoming packet is checked against NAT rules
- Packet is un-translated first, before ACL check
  - In **ASA 8.2** and below, incoming packet was subjected to ACL check prior to un-translation
- NAT rules can determine the egress interface at this stage

Packet Processing: ACL Check



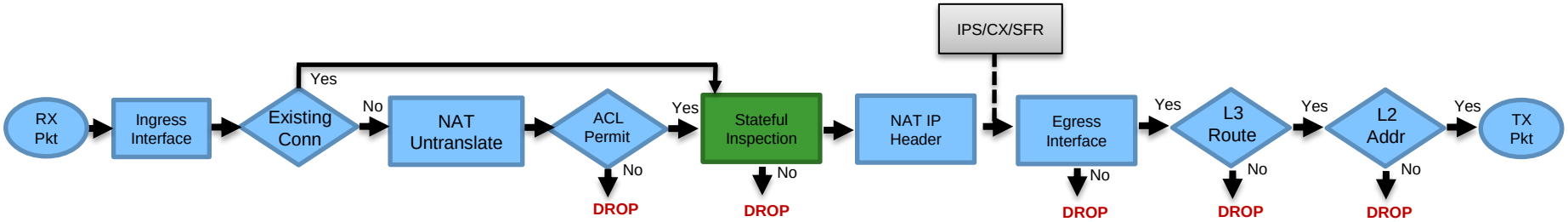
- First packet in flow is processed through ACL checks
- ACLs are **first configured** match
- First packet in flow matches ACE, incrementing hit count by one

```
asa# show access-list inside
access-list inside line 10 permit ip 10.1.1.0 255.255.255.0 any (hitcnt=1)
```

- Denied packets are dropped and logged

```
ASA-4-106023: Deny tcp src inside:10.1.1.9/11034 dst outside:198.133.219.25/80 by access-group "inside"
```

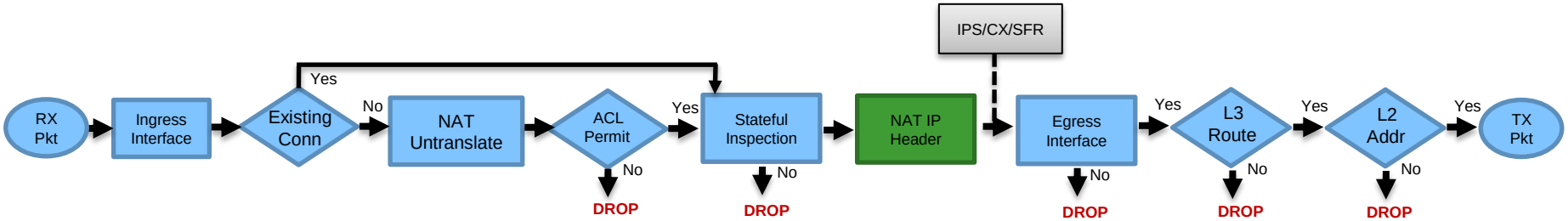
Packet Processing: Stateful Inspection



- Stateful inspection ensures protocol compliance at TCP/UDP/ICMP level
- (Optional) Customizable application inspection up to Layer 7 (FTP, SIP, and so on)
  - Rewrite embedded IP addresses, open up ACL pinholes for secondary connections
  - Additional security checks are applied to the application payload

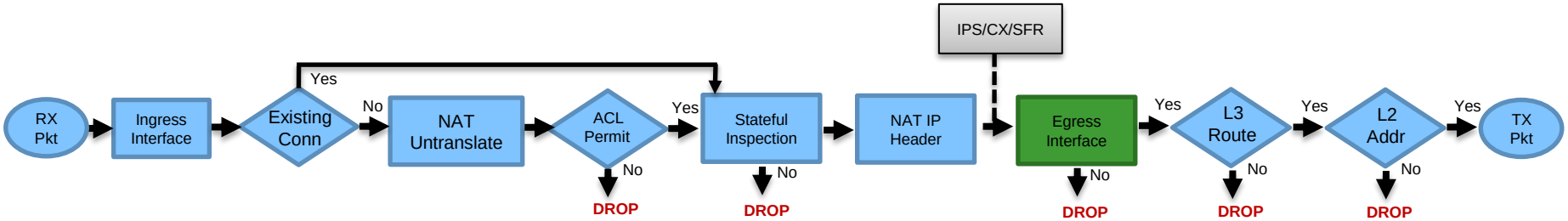
```
ASA-4-406002: FTP port command different address: 10.2.252.21(192.168.1.21) to
209.165.202.130 on interface inside

ASA-4-405104: H225 message received from outside_address/outside_port to
inside_address/inside_port before SETUP
```

Packet Processing: NAT IP Header

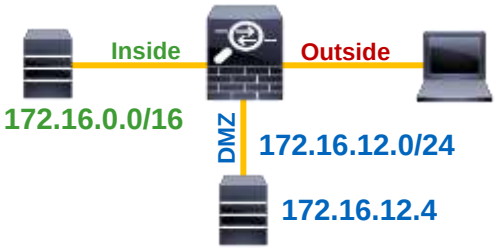


- Translate the source and destination IP addresses in the IP header
- Translate the port if performing PAT
- Update header checksums
- (Optional) Following the above, pass packet to IPS or SFR module
  - Real (pre-NAT) IP address information is supplied as meta data

Packet Processing: Egress Interface



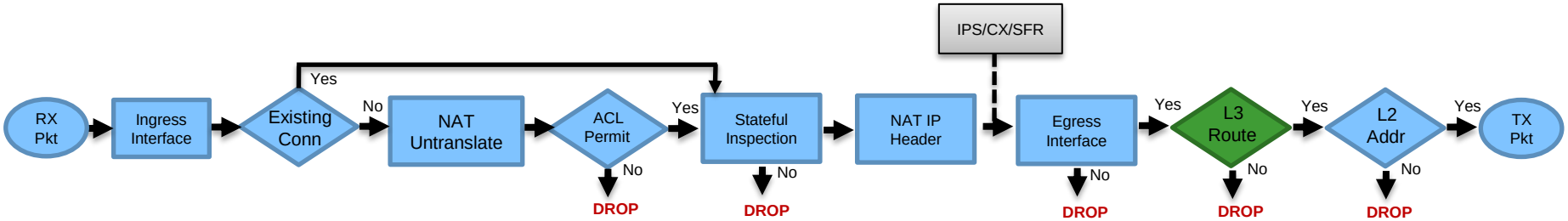
- Packet is **virtually** forwarded to egress interface (not forwarded to the Ethernet NIC yet)
- Egress interface is determined **first** by translation rules or existing conn entry, only **then** the routing table
- If NAT does not divert to the egress interface, the global routing table is consulted to determine egress interface



Packets received on **outside** and destined to **192.168.12.4** get routed to **172.16.12.4** on **inside** based on NAT configuration.

```
nat (inside,outside) source static 172.16.0.0-net 192.168.0.0-net
nat (dmz,outside) source static 172.16.12.0-net 192.168.12.0-net
```

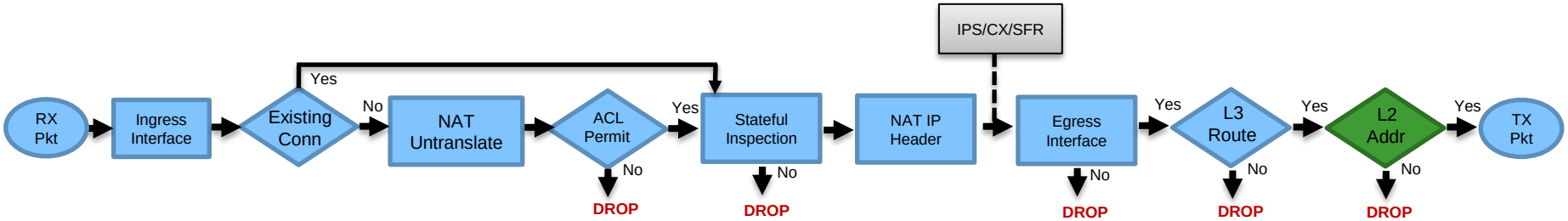
Packet Processing: L3 Route Lookup



- Once at egress interface, an interface route lookup is performed
- Only routes pointing out the egress interface are eligible
- Remember: NAT rule can forward the packet to the egress interface, even though the routing table may point to a different interface
  - If the destination is not routable out of the identified egress interface, the packet is dropped

```
%ASA-6-110003: Routing failed to locate next hop for TCP from inside:192.168.103.220/59138
to dmz:172.15.124.76/23
```

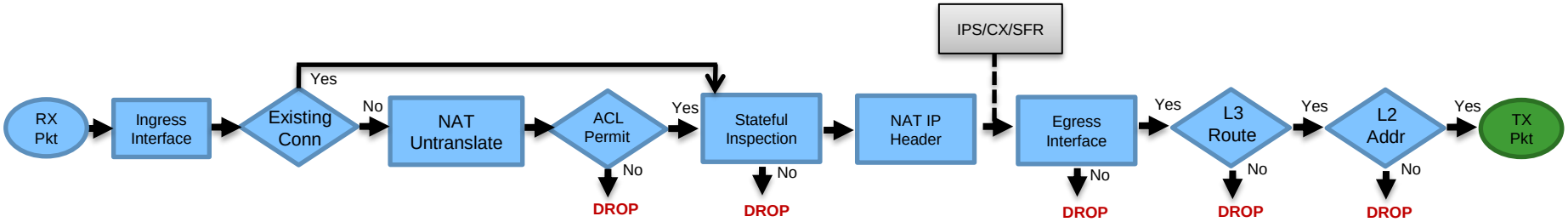
Packet Processing: L2 Address Lookup



- Once a Layer 3 route has been found, and next hop IP address identified, Layer 2 resolution is performed
  - Layer 2 rewrite of MAC header
- If Layer 2 resolution fails — **no** syslog
  - **show arp** will not display an entry for the L3 next hop
  - **debug arp** will indicate if we are not receiving an ARP reply

```
arp-req: generating request for 10.1.2.33 at interface outside
arp-req: request for 10.1.2.33 still  pending
```

Packet Processing: Transmit Packet



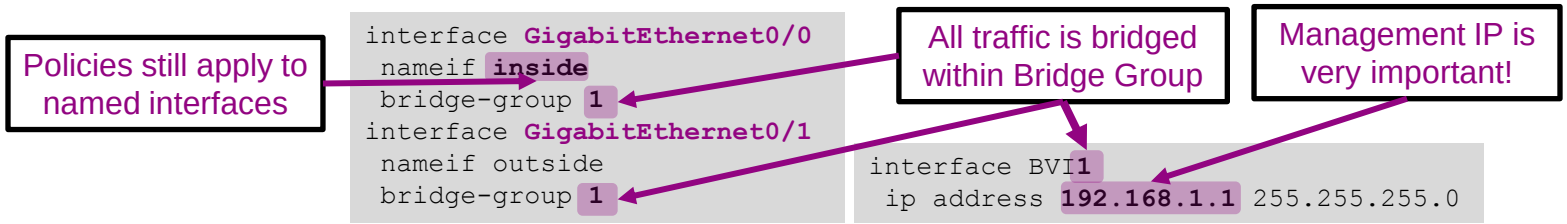
- Packet is transmitted on wire
- Interface counters will increment on interface
- **Underrun** counter indicates drops due to egress interface oversubscription
  - TX ring is full

```
asa# show interface outside
Interface GigabitEthernet0/1 "outside", is up, line protocol is up
  Hardware is i82574L rev00, BW 1000 Mbps, DLY 10 usec
          MAC address 503d.e59d.90ab, MTU 1500
          IP address 172.18.124.149, subnet mask 255.255.255.0
          …
          273399 packets output, 115316725 bytes, 80 underruns
          …
          input queue (blocks free curr/low): hardware (485/441)
          output queue (blocks free curr/low): hardware (463/0)
```

Firewall Deployment Modes

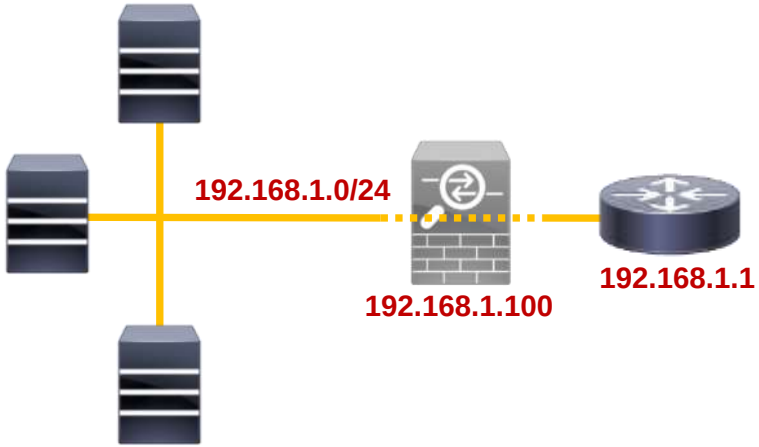
Routed and Transparent Firewall

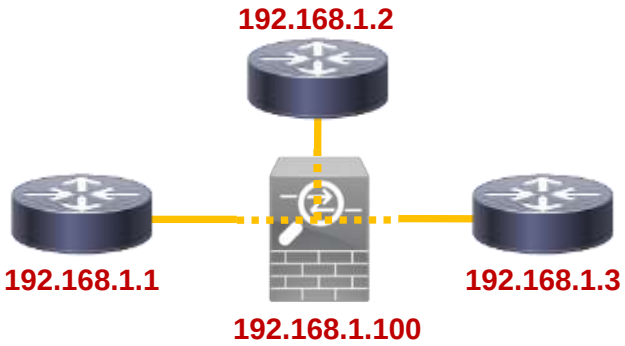
- Transparent Mode transforms ASA into a “bump in the wire”
  - Easy to drop into an existing network and pass non-IP traffic
  - **Pass-through** dynamic and multicast routing, no VPN support
  - Trunk interfaces are supported for VLAN bridging
  - Multiple Bridge Groups with up to 4 (sub)interfaces each in **ASA 8.4(1)+**
  - External router is **required** for inter-Bridge Group communication



Deploying Transparent Firewall

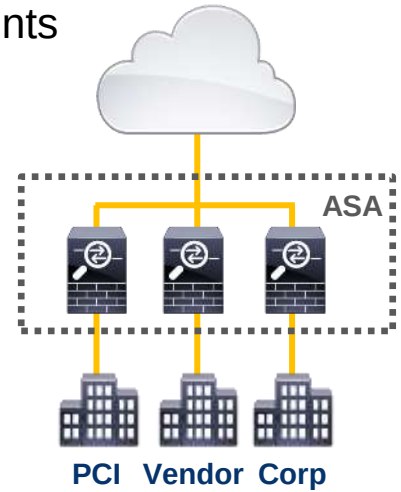
- All bridged segments and the **BVI** must be in the same IP subnet
  - **arp permit-nonconnected** is **required** with secondary subnets in **ASA 8.4(3)+**
- ASA typically separates hosts and their default gateway or routers
  - ASA **needs** routes to remote subnets for management, application inspection, NAT
  - Any external routes must be pointed **through** the ASA, never to the BVI IP





Multiple-Context Mode

- Multiple “virtual” firewalls operate concurrently on a **single** physical device
  - Each such firewall is called a **security context**, up to 250 contexts total
  - Independent interface sets, policies, routing tables, management, and so on
  - Inter-context routing via shared interfaces with virtual MAC addresses on ASA
  - Commonly deployed in VRF environments at intersection points
  - Mix routed and transparent contexts
- All contexts share **limited** hardware resources
  - Some resources can be limited on per-context basis
  - No VPN support except LAN-to-LAN tunnels in **ASA 9.0**



High Availability with Failover

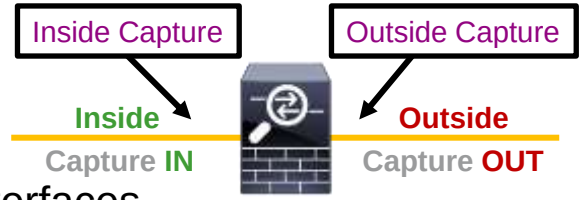
- A **pair** of identical ASA devices can be configured in Failover
  - Licensed features are aggregated except 3DES in **ASA 8.3+**
  - Data interface connections must be mirrored between the units **with** L2 adjacency
  - Primary and Secondary **designations** are statically assigned
  - Unit in Active **role** is processing all transit traffic, Standby takes over when needed
  - Virtual IP and MAC addresses on data interfaces move with the active unit
  - Centralized management from the active unit or context
- Optional Stateful failover “mirrors” stateful conn table between peers
  - Most connections survive a switchover seamlessly to the endpoints
  - Short-lived ICMP and HTTP connections are not replicated by default

High Scalability with Clustering

- Preserves all benefits of failover
- Combine identical ASA appliances in one traffic processing system
  - Up to 16 ASA 5585-X in **ASA 9.2(1)+** for 640Gbps UDP and 320Gbps EMIX
  - Up to 2 ASA5500-X (ASA5512-X requires Security Plus) in **ASA 9.1(4)+**
  - Centralized configuration mirrored to all members
- Stateless load-balancing via IP routing or Clustered Etherchannel with LACP
  - Cluster survives a single unit failure with minimal impact (much like failover)
  - All units should be connected to the same subnet on each logical interface

Troubleshooting Tools

Packet Capture



- In-line capability to record packets passing through ASA
- Two key steps in troubleshooting with captures
  - Apply capture under unique name to ingress and egress interfaces
  - Define the traffic that you want to capture, use pre-NAT “on the wire” information
  - Tcpdump-like format for displaying captured packets on the box

```
asa# capture OUT interface outside match ip any host 172.18.124.1
asa# capture IN interface inside match ip any host 172.18.124.1
asa# show capture IN

4 packets captured

   1: 10:51:26.139046       802.1Q vlan#10 P0 172.18.254.46 > 172.18.124.1: icmp: echo request
   2: 10:51:26.139503       802.1Q vlan#10 P0 172.18.124.1 > 172.18.254.46: icmp: echo reply
   3: 10:51:27.140739       802.1Q vlan#10 P0 172.18.254.46 > 172.18.124.1: icmp: echo request
   4: 10:51:27.141182       802.1Q vlan#10 P0 172.18.124.1 > 172.18.254.46: icmp: echo reply
4 packets shown
asa# no capture IN
```

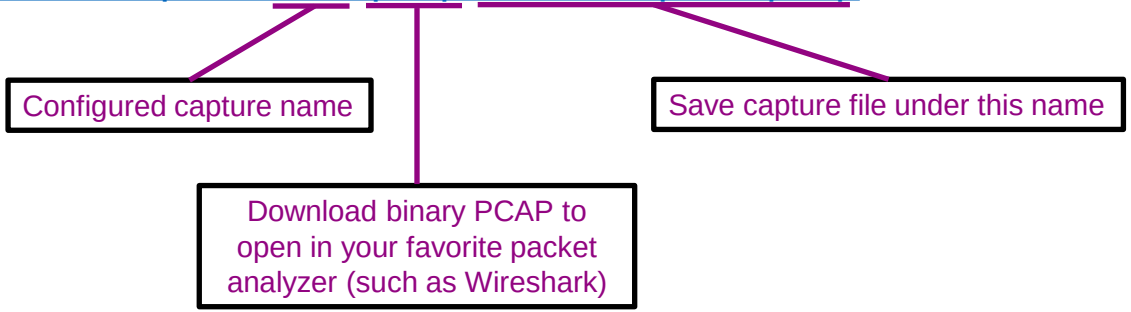
Unlike ACL, **match** covers both directions of the flow

Remember to remove the captures when done with troubleshooting

Packet Capture

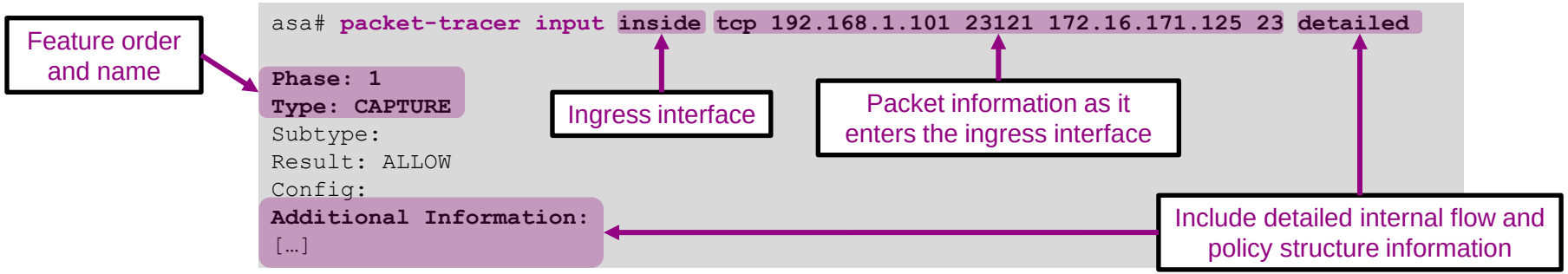
- Capture buffer maintained in RAM (512KB by default)
  - Stops capturing when full by default, **circular** option available
- Default recorded packet length is 1518 bytes
- Copy captures off via TFTP or retrieve through HTTPS with your web browser
  - Do this before removing the capture with **no capture**

https://x.x.x.x/admin/capture/OUT/pcap/outsidecapture.pcap



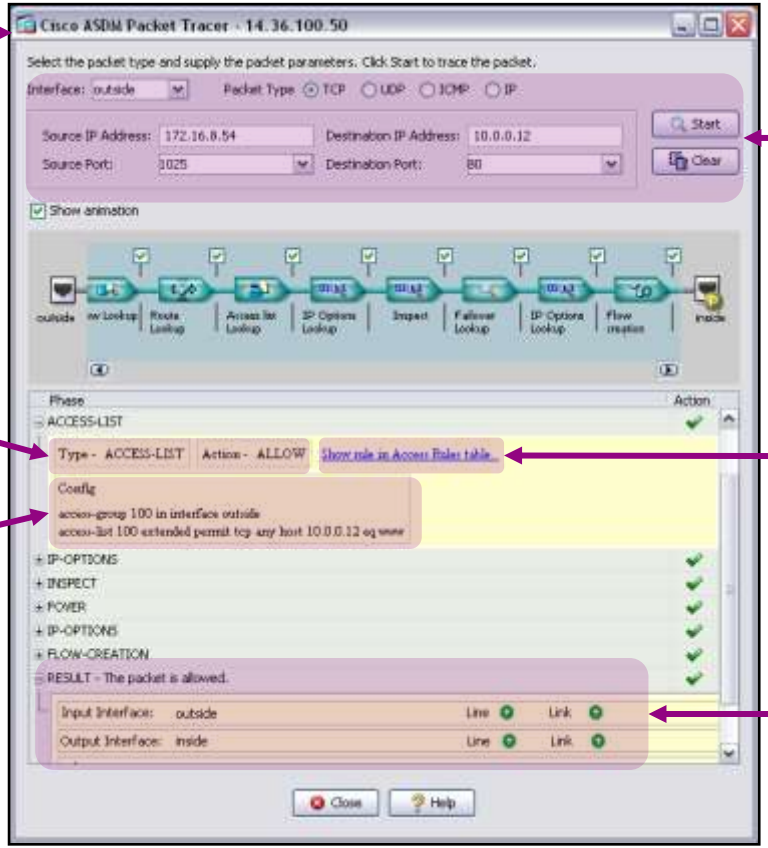
Packet Tracer

- Unique capability to record the path of a specially tagged packet through ASA
  - Best way to understand the packet path in the specific software version
- Inject a simulated packet to analyse the behaviour and validate configuration



Packet Tracer in ASDM

Launch from **Tools > Packet Tracer**



Define simulated packet

Feature type and resulting action

Direct link to edit policy

Associated configuration

Final outcome (allowed or dropped) and egress interface information

Packet Tracer: Tracing Captured Packet

- Enable packet tracer within an internal packet capture

```
asa# capture IN interface inside trace trace-count 20 match tcp any any eq
```

Trace inbound packets only

Traced packet count per capture (50 by default)

- Find the packet that you want to trace in the capture

```
asa# show capture inside
  68 packets captured
  1: 15:22:47.581116 10.1.1.2.31746 > 198.133.219.25.80: S
  2: 15:22:47.583465 198.133.219.25.80 > 10.1.1.2.31746: S ack
  3: 15:22:47.585052 10.1.1.2.31746 > 198.133.219.25.80: . ack
  4: 15:22:49.223728 10.1.1.2.31746 > 198.133.219.25.80: P ack
  5: 15:22:49.223758 198.133.219.25.80 > 10.1.1.2.31746: . Ack
     ...
```

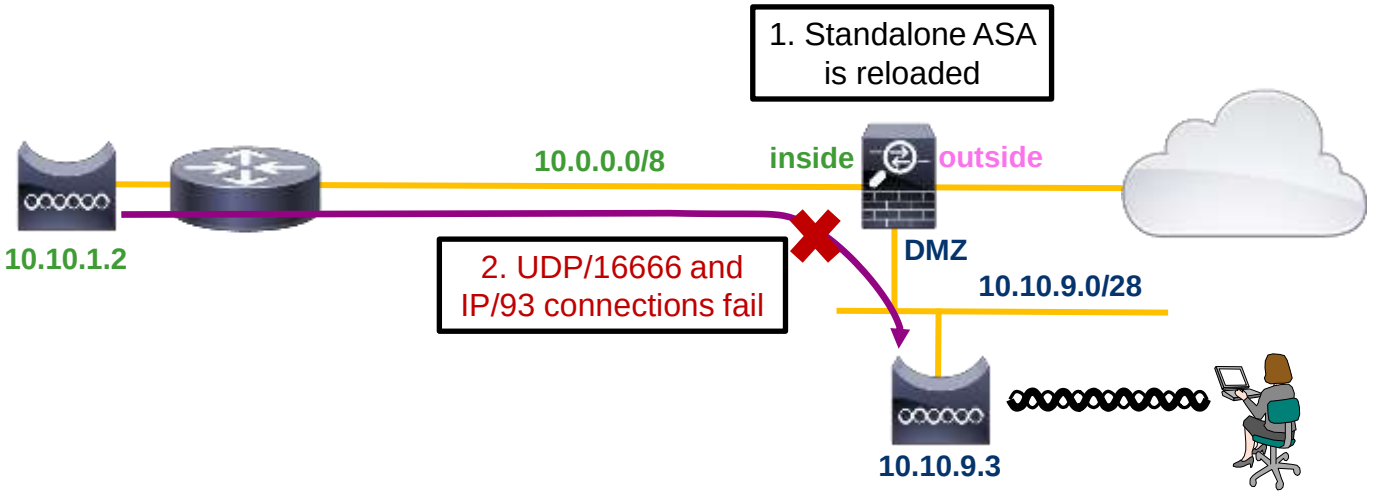
- Select that packet to show the tracer results

```
asa# show capture inside trace packet-number 4
```

Troubleshooting Case Study

Problem Summary

- After reloading the ASA, wireless mobility traffic (UDP and IP Protocol 93) from **inside** WLC to **DMZ** WLC fails
- Other traffic (TCP) recovers successfully
- The problem is mitigated by running **clear local-host** on the ASA



Checking Connection Table and Drops

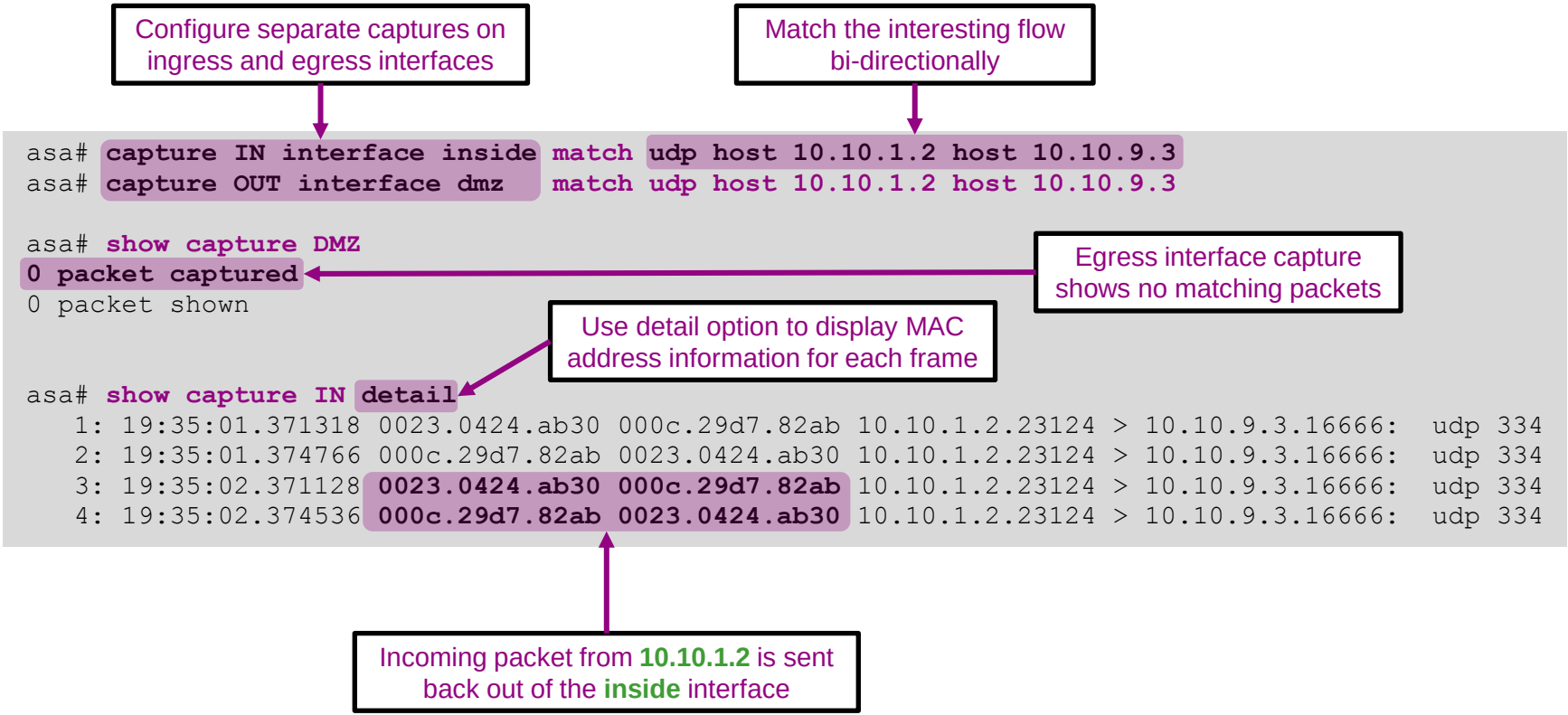
- Connections are built and passing traffic through the ASA

```
asa# show conn address 10.10.1.2

126 in use, 12654 most used
97 inside 10.10.9.3 inside 10.10.1.2, idle 0:00:00, bytes 32210
UDP inside 10.10.9.3:16666 inside 10.10.1.2:23124, idle 0:00:00, bytes 4338, flags -
97 inside 10.10.9.3 inside 10.10.1.2, idle 0:00:00, bytes 157240
```

- No packets dropped in ASP and no syslogs of interest

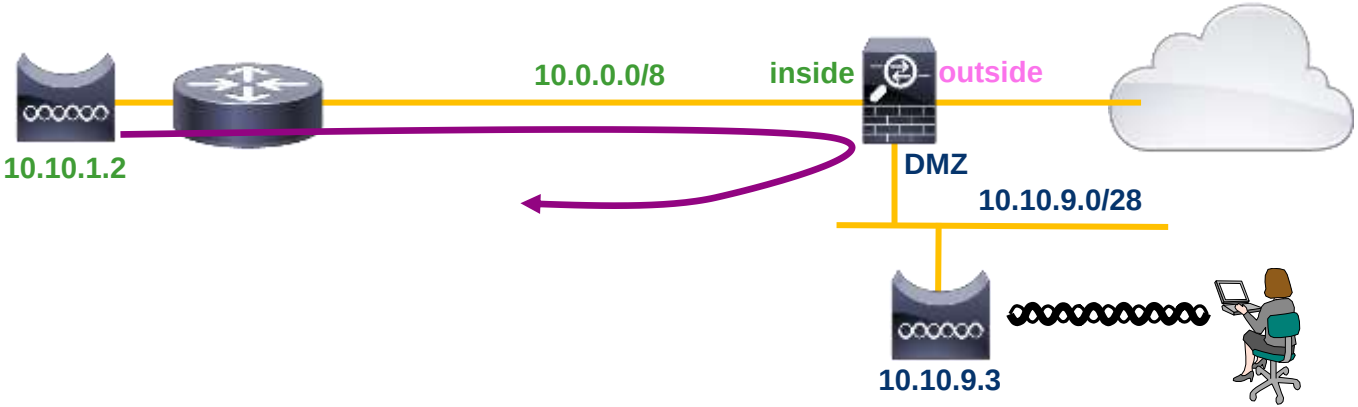
```
asa# capture asp type asp-drop all buffer 1000000
asa# show capture asp | include 10.10.1.2
asa#
asa# show log | include 10.10.1.2
```

Reviewing Packet Captures



U-Turn Connection

- Traffic is looping back out the inside interface towards the sender




```
asa# sh run | include same-security
same-security-traffic permit intra-interface
```

Allow connections to establish between two endpoints behind the same ASA interface (U-turn)

Checking Packet Tracer

- Packet Tracer shows that a **new** UDP flow will be correctly passed to **DMZ**

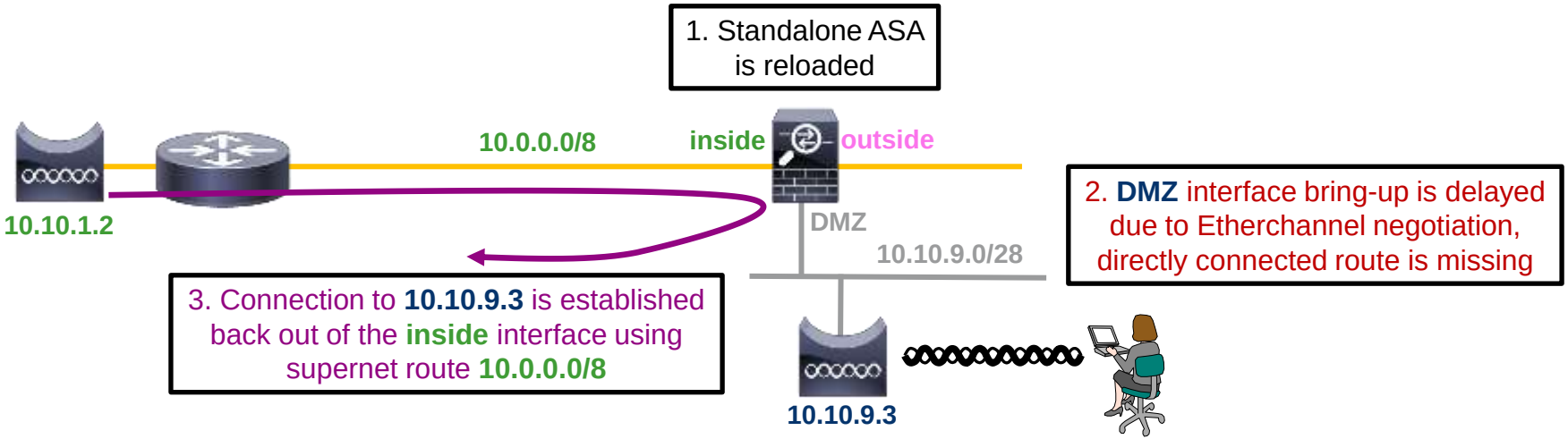
```
asa# packet-tracer input inside udp 10.10.1.2 23124 10.10.9.3 16666
[…]
Phase: 3
Type: ROUTE-LOOKUP
Subtype: input
Result: ALLOW
Config:
Additional Information:
in   10.10.0.0        255.255.0.0      dmz
[…]
Result:
input-interface: inside
input-status: up
input-line-status: up
output-interface: dmz
output-status: up
output-line-status: up
Action: allow
```

Correct routing prefix selected

Flow is allowed

Root Cause

- When conn entry was created, route lookup for **10.10.9.3** resolved to **inside**
- If DMZ interface was not up, the route to **10.10.9.0/28** was not present



Floating Connection Timeout

- The “bad” connection never times out since the UDP traffic is constantly flowing
  - TCP is stateless, so the connection would terminate and re-establish on its own
  - ASA needs to tear the original connection down when the corresponding route changes
  - **ASA 8.4(2)+** introduces **timeout floating-conn** to accomplish this goal

```
asa# show run timeout
timeout xlate 9:00:00
timeout pat-xlate 0:00:30
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp 0:00:02
timeout sunrpc 0:10:00 h323 0:05:00 h225 1:00:00 mgcp 0:05:00 mgcp-pat 0:05:00
timeout sip 0:30:00 sip_media 0:02:00 sip-invite 0:03:00 sip-disconnect 0:02:00
timeout sip-provisional-media 0:02:00 uauth 9:00:00 absolute uauth 0:01:00 inactivity
timeout tcp-proxy-reassembly 0:01:00
timeout floating-conn 0:00:00
asa#
asa# configure terminal
asa(config)# timeout floating-conn 0:01:00
```

Schedule the conn entry for termination in **1 minute** if a matching packet yields a different egress interface on route lookup

Q & A

Useful Links

- Cisco Live BRKSEC-3021 : Maximizing Firewall Performance
  - https://www.ciscolive.com/online/connect/sessionDetail.ww?SESSION_ID=78719&tclass=popup
- Cisco Live BRKSEC-3020 : Troubleshooting ASA Firewalls
  - https://www.ciscolive.com/online/connect/sessionDetail.ww?SESSION_ID=77832&tclass=popup
- Cisco Live BRKSEC-3032 : ASA Clustering Deep Dive
  - https://www.ciscolive.com/online/connect/sessionDetail.ww?SESSION_ID=78721&tclass=popup