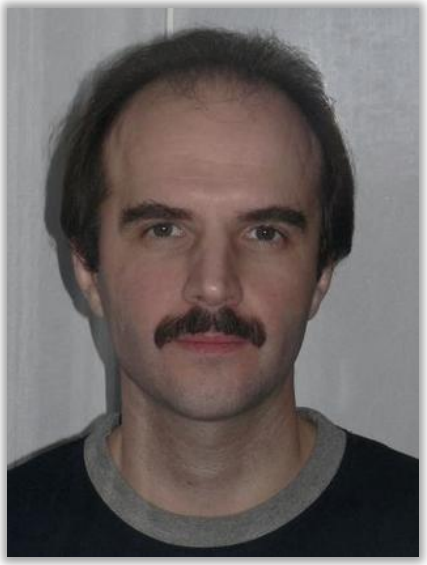
Using packet-tracer, capture and other Cisco ASA tools for network troubleshooting

Oleg Tipisov
Customer Support Engineer, Cisco TAC

Jan, 2014. Revision 1.0
Cisco Public

Cisco Support Community – Expert Series Webcasts in Russian

Сегодня на семинаре эксперт Cisco TAC **Олег Типисов** приведет примеры использования различных возможностей и диагностических средств Cisco ASA для решения проблем, возникающих при передаче трафика.



Олег Типисов

Инженер центра технической поддержки Cisco TAC в Москве

Спасибо, что посетили наш семинар сегодня

Сегодняшняя презентация включает опросы аудитории

Пожалуйста, участвуйте!

Скачать презентацию вы можете по ссылке:

https://supportforums.cisco.com/docs/DOC-39468

Присылайте Ваши вопросы!

Используйте Q&A панель, чтобы послать вопрос. Наши эксперты ответят на них

Introduction

- Which pill would you choose, red or blue?

Agenda

- ASA Software Architecture
- Packet Tracer
- Packet Capture
- TCP Ping
- Case Study: Infected Local Host
- Resource Management
- Monitoring Resource Utilization
- Conclusion

ASA Software Architecture

Terminology

- Data Path (DP) – Process “thru-the-box” packets
- Control Point (CP) – Handle “to-the-box” packets and also the console
- CP handles all of the configuration and management as well as some network protocols like ARP and routing
- Most of the code-base is CP related, but most of the cycles on the box are in the DP
- Data Path is a separate thread within the ASA process
- Data Path is “flow based”, with a distinct flow-creation path (the slow-path) and a fast-path for packets on existing flows
- The combination of the fast-path and the slow-path is known as Accelerated Security Path

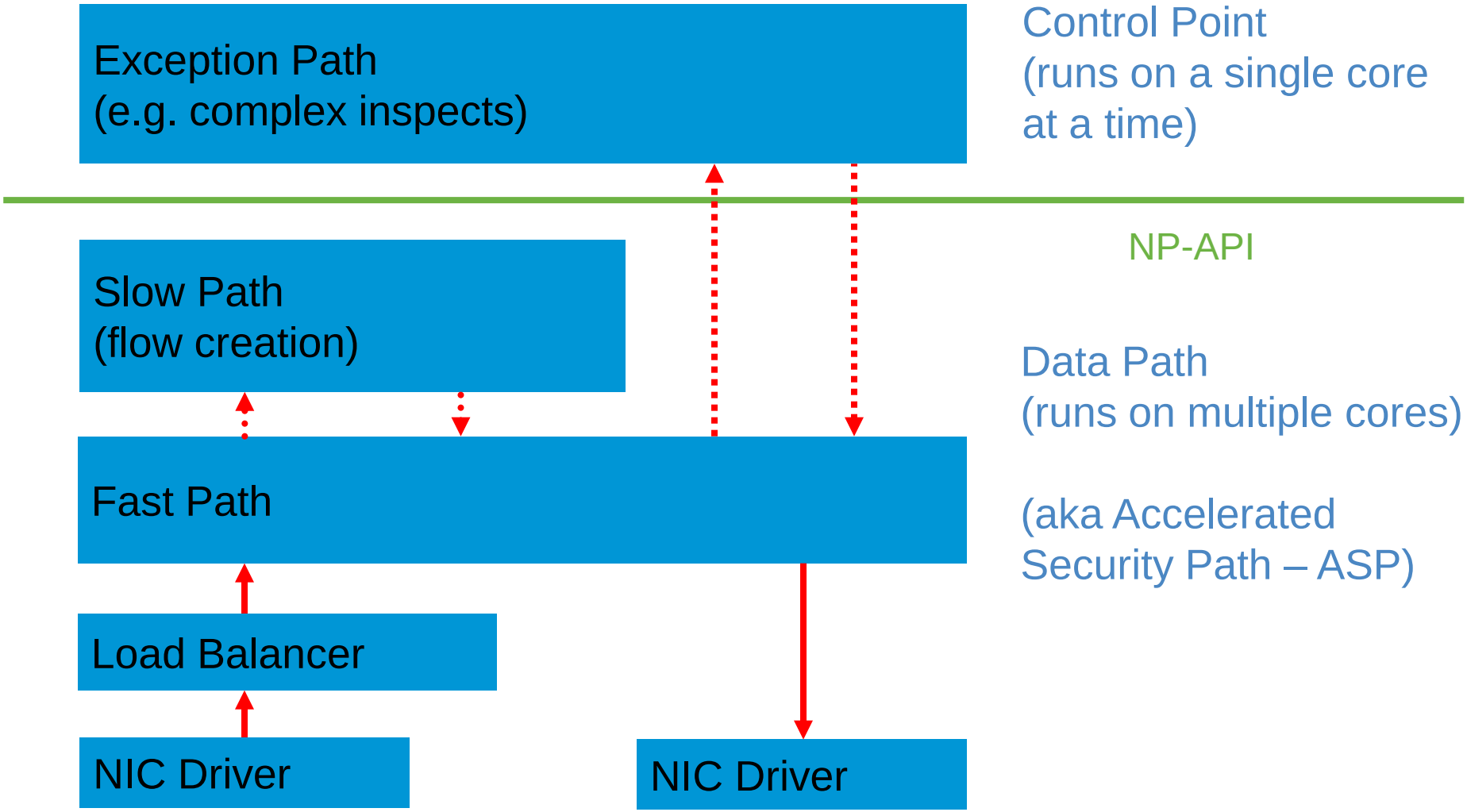
History

- In the 6.x days the Data Path was mixed in with the Control Point code, and there was no separation between the flow-setup path and the fast-path
- The FWSM team took the 6.0 code, and forked development creating a separate Data Path that ran on IBM network processors (NP), while still using the rest of the PIX code as the Control Point
- For 7.0 (ASA), we merged with the FWSM code creating a new Data Path that emulated their NP code, which we called the SoftNP

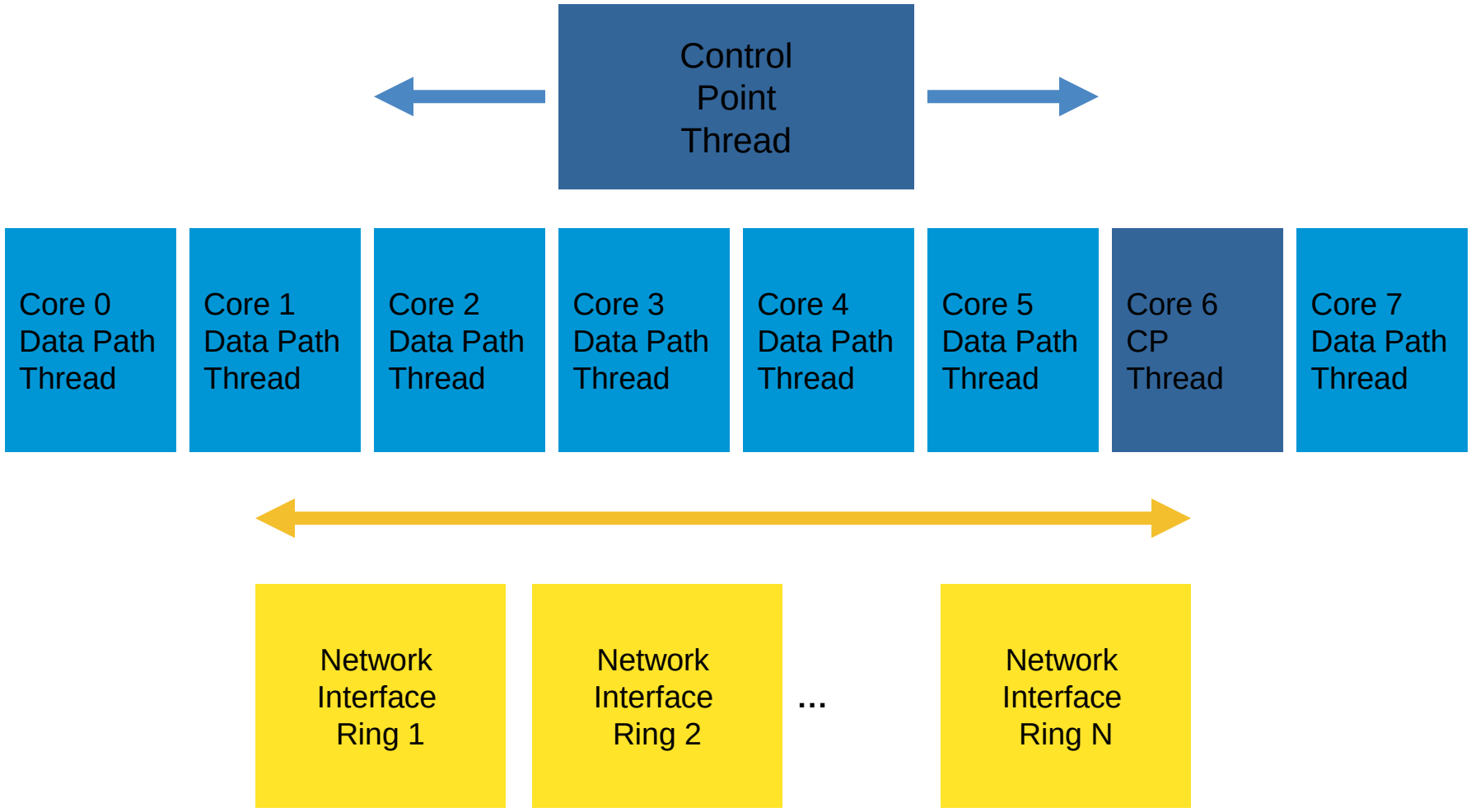
History

- SoftNP was designed as a portable 32-bit / 64-bit Data Path with a well defined API (NP-API)
- Control Point code should not directly access SoftNP data structures and vice-versa. All communication should be done through the NP-API
- Designed from the ground up to be high-performance and scalable
- Many features were added over the old 6.x Data Path (e.g. virtual firewalls, IPv6, transparent firewalls, etc.)

ASA Software Architecture



SMP Architecture



SMP Platforms

- The first SMP platform was ASA5580 (ASA5580-20, ASA5580-40)
- All modern ASA platforms are SMP platforms
  - ASA5512-X, ASA5515-X, ASA5525-X, ASA5545-X, ASA5555-X
  - ASA5585 SSP-10, SSP-20, SSP-40, SSP-60
- All of them run -smp- image (e.g. asa913-2-smp-k8.bin)
- But only one CPU core is available to ASA software on low-end devices and the rest is dedicated to IPS or CX software module
  - ASA5512-X – ASA5545-X
- Example:

```
Hardware:   ASA5545, 12288 MB RAM, CPU Lynnfield 2660 MHz, 1 CPU (8 cores)
            ASA: 6144 MB RAM, 1 CPU (1 core)
```

For your reference

SMP Platforms (from “show version”)

Model	CPU	ASA Cores
5512-X	Clarkdale 2793 MHz 1 CPU (2 cores)	1 CPU (1 core)
5515-X	Clarkdale 3059 MHz, 1 CPU (4 cores)	1 CPU (1 core)
5525-X	Lynnfield 2393 MHz, 1 CPU (4 cores)	1 CPU (1 core)
5545-X	Lynnfield 2660 MHz, 1 CPU (8 cores)	1 CPU (1 core)
5555-X	Lynnfield 2792 MHz, 1 CPU (8 cores)	1 CPU (2 cores)
5580-20	AMD Opteron 2600 MHz, 2 CPUs (4 cores)	ALL
5580-40	AMD Opteron 2600 MHz, 4 CPUs (8 cores)	ALL
5585 SSP-10	Xeon 5500 series 2000 MHz, 1 CPU (4 cores)	ALL
5585 SSP-20	Xeon 5500 series 2133 MHz, 1 CPU (8 cores)	ALL
5585 SSP-40	Xeon 5500 series 2133 MHz, 2 CPUs (16 cores)	ALL
5585 SSP-60	Xeon 5600 series 2400 MHz, 2 CPUs (24 cores)	ALL

Note: Cores are not really “cores”. They’re “threads”. For example, SSP-20 runs on Xeon L5518 4C/8T

ASA Troubleshooting Tools

Packet Tracer

Packet Tracer

- A packet can be traced by
  - Defining packet characteristics via ASA CLI
  - Capturing packets using trace option
- In both cases a packet, tagged with the trace option, is injected into the specified interface and processed in data path
- This packet is real and can be captured on the interface by the “capture” tool
- Each “action” taken on the packet is recorded
- When the packet reaches egress interface, or is dropped, it is punted to the control plane
- The control plane reads and displays the actions taken on the packet, along with the associated lines in the configuration

Tracing Packets from CLI

- Packet tracer is useful for both configuration testing and troubleshooting of packet forwarding issues
- It is a primary tool to test ACLs and NAT configuration
- Packet tracer is not a traffic generator: packet payload is empty, only basic packet characteristics can be defined
- “detailed” option can be used to display ASP classification rules for the packet

```
packet-tracer input <interface> {tcp | udp | icmp | rawip} <source>
<destination> [detailed | xml]
```

Packet Tracer Example

- In this example we will trace the packet from outside host 195.1.1.1 to SMTP server 192.0.2.1 (172.16.1.2) in DMZ
- Relevant parts of configuration are shown below

```
object network obj-172.16.1.2
 host 172.16.1.2

object service SMTP
 service tcp source eq smtp

nat (dmz,outside) source static obj-172.16.1.2 interface service SMTP SMTP

access-list outside_in extended permit tcp any host 172.16.1.2 eq smtp
access-group outside_in in interface outside

access-list ips_for_dmz extended permit tcp any host 172.16.1.2 eq smtp

class-map ips_for_dmz
 match access-list ips_for_dmz

policy-map outside_policy
 class ips_for_dmz
  ips inline fail-open

service-policy outside_policy interface outside
```

Packet Tracer Example

- Capture was configured to verify that packet-tracer generates real packet

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 1
Type: CAPTURE
Subtype:
Result: ALLOW
Config:
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7ffe60593830, priority=13, domain=capture, deny=false
        hits=1, user_data=0x7ffe6056b090, cs_id=0x0, l3_type=0x0
        src mac=0000.0000.0000, mask=0000.0000.0000
        dst mac=0000.0000.0000, mask=0000.0000.0000
        input_ifc=outside, output_ifc=any
```

show asp table classify interface outside domain capture

show capture

Information displayed here is not correct

Packet Tracer Example

- "show capture" can be used to display the packet

```
ASA/C1# show capture
capture cap-out type raw-data interface outside [Capturing - 74 bytes]
  match tcp host 195.1.1.1 host 192.0.2.1

ASA/C1# show capture cap-out detail

1 packet captured

   1: 12:59:15.179586 0000.0000.0000 503d.e59d.8997 0x8100 Length: 58
      802.1Q vlan#76 P0 195.1.1.1.1234 > 192.0.2.1.25: S [tcp sum ok]
914040549:914040549(0) win 8192 (ttl 255, id 35455)
```

Packet Tracer Example

- MAC ACL is used by default in routed firewall mode to allow only IPv4, IPv6 and ARP traffic

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 2
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7ffe5f45b930, priority=1, domain=permit, deny=false
        hits=61, user_data=0x0, cs_id=0x0, l3_type=0x8
        src mac=0000.0000.0000, mask=0000.0000.0000
        dst mac=0000.0000.0000, mask=0100.0000.0000
        input_ifc=outside, output_ifc=any
```

show asp table classify interface outside domain permit [hits]

Bytes are swapped here:
0x08 is 0x0800 = IPv4
0x608 is 0x0806 = ARP
0xdd86 is 0x86dd = IPv6

This rule is about IPv4 unicast

Packet Tracer Example

- UN-NAT changes destination IP from 192.0.2.1 to 172.16.1.2 and diverts the packet to DMZ interface ignoring routing table

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 3
Type: UN-NAT
Subtype: static
Result: ALLOW
Config:
nat (dmz,outside) source static obj-172.16.1.2 interface service SMTP SMTP
Additional Information:
NAT divert to egress interface dmz
Untranslate 192.0.2.1/25 to 172.16.1.2/25
```

show nat [detail]

show nat divert-table

- NAT on ASA is overloaded with functions: NAT, “policy-based” routing (NAT divert), security (NAT RPF check). In case of a conflict between NAT divert and routing you would see:

```
%ASA-6-110003: Routing failed to locate next hop for TCP from <nameif>:<IP>/<port> to
<nameif>:<IP>/<port>
```

Packet Tracer Example

- IPv4 access-list is required to allow traffic from outside to DMZ
- Note that real IP (172.16.1.2) should be specified in outside ACL in 8.3+, because UN-NAT is performed before ACL check

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 4
Type: ACCESS-LIST
Subtype: log
Result: ALLOW
Config:
access-group outside_in in interface outside
access-list outside_in extended permit tcp any host 172.16.1.2 eq smtp
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7ffe6055b390, priority=13, domain=permit, deny=false
        hits=0, user_data=0x7ffe559729c0, cs_id=0x0, use_real_addr, flags=0x0, protocol=6
        src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0
        dst ip/id=172.16.1.2, mask=255.255.255.255, port=25, tag=0, dscp=0x0
        input_ifc=outside, output_ifc=any
```

show access-list

show asp table classify interface outside domain permit [hits]

Packet Tracer Example

- We have a rule in NAT ASP classification table, but source IP is not changed by NAT in this case

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 5
Type: NAT
Subtype:
Result: ALLOW
Config:
nat (dmz,outside) source static obj-172.16.1.2 interface service SMTP SMTP
Additional Information:
Static translate 195.1.1.1/1234 to 195.1.1.1/1234
 Forward Flow based lookup yields rule:
 in  id=0x7ffe6056c980, priority=6, domain=nat, deny=false
        hits=14, user_data=0x7ffe60541160, cs_id=0x0, flags=0x0, protocol=6
        src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0
        dst ip/id=192.0.2.1, mask=255.255.255.255, port=25, tag=0, dscp=0x0
        input_ifc=outside, output_ifc=dmz
```

show nat [detail]

show asp table classify domain nat

Packet Tracer Example

- Per-session PAT is a new 9.0 feature created for clustering
- It can also be very useful to improve PAT scalability in non-clustering configurations (more on this later)
- It applies to dynamic PAT only

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 6
Type: NAT
Subtype: per-session
Result: ALLOW
Config:
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7ffe5ff158b0, priority=0, domain=nat-per-session, deny=false
        hits=27, user_data=0x0, cs_id=0x0, reverse, use_real_addr, flags=0x0, protocol=6
        src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0
        dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0, dscp=0x0
        input_ifc=any, output_ifc=any
```

show asp table classify domain nat-per-session

show run all xlate

"false" means that per-session PAT is enabled for the flow:

xlate per-session permit tcp any4 any4

Packet Tracer Example

- RSVP and IGMP packets with IP options are allowed by default
- We hit implicit deny rule here, but our packet doesn't have any IP options and is allowed to go
- Use "show run all policy-map" to learn more

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 7
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7ffe5f4617e0, priority=0, domain=inspect-ip-options, deny=true
        hits=797, user_data=0x0, cs_id=0x0, reverse, flags=0x0, protocol=0
        src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0
        dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0, dscp=0x0
        input_ifc=outside, output_ifc=any
```

show service-policy inspect ip-options

show asp table classify domain inspect-ip-options

Packet Tracer Example

- SMTP inspection is enabled by default
- Use “show run all policy-map _default_esmtp_map” to learn more

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 8
Type: INSPECT
Subtype: inspect-smtp
Result: ALLOW
Config:
class-map inspection_default
 match default-inspection-traffic
policy-map global_policy
 class inspection_default
  inspect esmtp _default_esmtp_map
service-policy global_policy global
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7ffe5ff26d40, priority=70, domain=inspect-smtp, deny=false
        hits=14, user_data=0x7ffe6019d910, cs_id=0x0, use_real_addr, flags=0x0, protocol=6
        src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0
        dst ip/id=0.0.0.0, mask=0.0.0.0, port=25, tag=0, dscp=0x0
        input_ifc=outside, output_ifc=any
```

show service-policy inspect esmtp

show asp table classify domain inspect-smtp

Packet Tracer Example

- IPS can be configured in inline or promiscuous mode
- Traffic, generated by packet-tracer, is not sent to IPS for analysis
- Real traffic is processed in a very specific way (more on this later)
- IPS rules should use real IP 172.16.1.2 since 8.3

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 9
Type: IDS
Subtype:
Result: ALLOW
Config:
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7ffe6056f6a0, priority=51, domain=ids, deny=false
        hits=1, user_data=0x7ffe605d78a0, cs_id=0x0, use_real_addr, flags=0x0, protocol=6
        src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0
        dst ip/id=172.16.1.2, mask=255.255.255.255, port=25, tag=0, dscp=0x0
        input_ifc=outside, output_ifc=any
```

show service-policy [ips]

show asp table classify domain ids

Packet Tracer Example

- NAT RPF check verifies that forward and reverse traffic hits the same NAT rule

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 10
Type: NAT
Subtype: rpf-check
Result: ALLOW
Config:
nat (dmz,outside) source static obj-172.16.1.2 interface service SMTP SMTP
Additional Information:
 Forward Flow based lookup yields rule:
 out id=0x7ffe6056cd30, priority=6, domain=nat-reverse, deny=false
        hits=14, user_data=0x7ffe591ccec0, cs_id=0x0, use_real_addr, flags=0x0, protocol=6
        src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0
        dst ip/id=172.16.1.2, mask=255.255.255.255, port=25, tag=0, dscp=0x0
        input_ifc=outside, output_ifc=dmz
```

show nat [detail]

show asp table classify domain nat-reverse

- It's a security mechanism that prevents outside hosts to reach inside host directly, bypassing NAT (remember that ACLs in 8.3 and above are configured to permit traffic to real IP)

Packet Tracer Example

- NAT RPF is a very complicated subject
- ASA behavior has changed several times
- The following syslog message can be produced if RPF check fails

```
%ASA-5-305013: Asymmetric NAT rules matched for forward and reverse flows; Connection for
icmp src outside:195.1.1.1 dst inside:10.1.1.2 (type 8, code 0) denied due to NAT reverse
path failure
```

Packet Tracer Example

- Flow is created and immediately torn down, syslog messages are generated
- Note that real ESMTP traffic would be punted to CP for inspection

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Phase: 13
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 2227, packet dispatched to next module
Module information for forward flow ...
snp_fp_tracer_drop
snp_fp_inspect_ip_options
snp_fp_tcp_normalizer
snp_fp_punt <inspect_esmtp>
snp_fp_translate
snp_ids
snp_fp_tcp_normalizer
snp_fp_adjacency
snp_fp_fragment
snp_ifc_stat
```

show conn [long]

show xlate

Packet Tracer Example

- Information about reverse flow is also displayed
- Reverse flow steps #11, #12 were omitted for brevity

```
Module information for reverse flow ...
snp_fp_tracer_drop
snp_fp_inspect_ip_options
snp_fp_tcp_normalizer
snp_fp_translate
snp_fp_punt <inspect_esmtp>
snp_ids
snp_fp_tcp_normalizer
snp_fp_adjacency
snp_fp_fragment
snp_ifc_stat
```

Packet Tracer Example

- Final result is shown here

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 25 detail

Result:
input-interface: outside
input-status: up
input-line-status: up
output-interface: dmz
output-status: up
output-line-status: up
Action: allow
```

- Syslog messages:

```
%ASA-6-302013: Built inbound TCP connection 2227 for outside:195.1.1.1/1234
(195.1.1.1/1234) to dmz:172.16.1.2/25 (192.0.2.1/25)

%ASA-6-302014: Teardown TCP connection 2227 for outside:195.1.1.1/1234 to
dmz:172.16.1.2/25 duration 0:00:00 bytes 0 Free the flow created as result of packet
injection
```

Packet Tracer Limitations and Restrictions

- Packet, generated by packet-tracer, is not sent to either ASA L7 inspection engines or IPS module, “show service-policy” stats not updated
- Packet tracer cannot be used to do VPN tracing in outside to inside direction
- Packet tracer is not supported in transparent mode
- Not all packet processing steps are shown (e.g. normalizer)
- Packet tracer cannot be run from system context to test multicontext classifier, but it can be run from user contexts to test context security policy

Packet Tracer and “acl-drop” Drop Code

- Sometimes it could be difficult to diagnose a problem looking just at the packet-tracer output. Simple example:

```
ASA/C1# packet-tracer input outside tcp 195.1.1.1 1234 192.0.2.1 139 detail

Phase: 3
Type: ROUTE-LOOKUP
Subtype: input
Result: ALLOW
Config:
Additional Information:
in   192.0.2.1        255.255.255.255 identity

Phase: 5
Type: ACCESS-LIST
Result: DROP
Implicit Rule
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7ffe5f45ca40, priority=0, domain=permit, deny=true
        hits=11, user_data=0x9, cs_id=0x0, use_real_addr, flags=0x1000, protocol=0
        src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0
        dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=0, dscp=0x0
        input_ifc=outside, output_ifc=any

Drop-reason: (acl-drop) Flow is denied by configured rule
```

Here we connect from outside to ASA outside IP 192.0.2.1, but PAT for port TCP/139 is not configured.

Packet is routed to “identity” interface, which is ASA itself…

…and is dropped by implicit drop rule. This is fine, but the problem is that “acl-drop” code is a generic drop code used in many different situations.

Packet Tracer and “acl-drop” Drop Code

- Sometimes syslog messages can help differentiate between different drop reasons. In this case we see:

```
%ASA-7-710005: TCP request discarded from 195.1.1.1/1234 to outside:192.0.2.1/139
```

- And adding “permit ip any any” line into interface ACL doesn’t change behavior, since interface ACL doesn’t control access to ASA itself
- Should we try to connect to port TCP/21 (which has valid PAT rule configured) we would see…

```
%ASA-4-106023: Deny tcp src outside:195.1.1.1/1234 dst inside:172.16.1.3/21 by access-group "outside_in"
```

- …if such traffic is dropped by implicit deny rule in interface ACL
- ASP drop code would be the same: acl-drop

Packet Capture

Packet Capture

- Capture types

```
ASA/C1# capture cap-out type ?

  asp-drop    Capture packets dropped with a particular reason
  isakmp      Capture encrypted and decrypted ISAKMP payloads
  raw-data    Capture inbound and outbound packets on one or more interfaces
  tls-proxy   Capture decrypted inbound and outbound data from TLS Proxy on one or more
interfaces
```

- ISAKMP capture can be useful for IPSec troubleshooting. ASA adds decrypted IKEv1 or IKEv2 packets to the capture and they can be decoded in Wireshark (this is beyond the scope of this presentation)
- ASP drop capture can be used to capture dropped packets
- The default type is “raw-data”, which allows capturing on ASA Ethernet interfaces, ASA-IPS control-plane interface, etc.

Packet Capture

- Traffic can be captured both before and after it passes through the firewall; one capture on the inside interface, one capture on the outside interface
- Ingress packets are captured before any packet processing has been done on them
- Egress packets are captured after all processing (excluding L2 source MAC rewrite)
- “nameif” needs to be configured on a interface to capture on it (which means you cannot capture on Ethernet interfaces in system context in multiple context mode)
- Do not use capture on heavily-loaded SMP systems! There is a risk of high CPU!

Packet Capture

```
capture cap-out [buffer <bytes>] [circular-buffer] [packet-length <bytes>] [headers-only]
interface <name> [real-time] {access-list <name> | match <capture-filter>}
```

- Capture buffer saved in RAM (default size is 512 KB)
- Default is to stop capturing when the buffer is full
- Default packet length is 1518 bytes
- Jumbo frames are supported (up to 9216 bytes)
- “real-time” option can be used to display captured packets on the screen in real time (not recommended)

Packet Capture

```
capture cap-out [buffer <bytes>] [circular-buffer] [packet-length <bytes>] [headers-only]
interface <name> [real-time] {access-list <name> | match <capture-filter> …}
```

- Capture is bidirectional if capture filter is configured in the command (“match …”)
- Capture is unidirectional if an ACL is used as a capture filter
- Don’t use “any” in ACL in 9.0 or newer versions as “any” means “any4 + any6”. In 9.0 we implemented “Unified ACLs for IPv4 and IPv6”. The diagnostics will be:

```
ERROR: Capture doesn't support access-list <CAPTURE1> containing mixed policies
```

- The “match” keyword can be used up to three times in the capture command

Packet Capture

- Verify if packets are captured:

```
ASA/C1# show capture
```

- Display packets (similar to tcpdump)

```
ASA/C1# show capture cap-out [detail | dump] …
```

- Several other options are available, but they are rarely used

Packet Capture

- Copy capture off the box via TFTP or FTP
- It can also be saved to disk0: if the server is unavailable
- In multiple context mode “copy” command is available in system context only (it uses admin context IP for copying)
- In single mode use:

```
ASA# copy /pcap capture:cap-out tftp://<IP>/cap-out.pcap
```

- In multiple context mode use:

```
ASA# copy /pcap capture:C1/cap-out tftp://<IP>/cap-out.pcap
```

Packet Capture

- Clear capture buffer to restart capture:

```
ASA/C1# clear capture cap-out
```

- Don’t forget to turn capture off when done:

```
ASA/C1# no capture cap-out
```

Packet Capture

- Contrary to popular belief (and documentation) this tool can be used to capture ARP packets

```
ASA/C1# capture cap1 int outside ethernet-type arp

ASA/C1# show capture cap1

2 packets captured

   1: 20:16:50.476156       802.1Q vlan#76 P0 arp who-has 192.0.2.2 tell 192.0.2.1
   2: 20:16:50.476690       802.1Q vlan#76 P0 arp reply 192.0.2.2 is-at 0:13:7f:3d:bd:1
```

- "show arp" and "show arp statistics" can also be used for ARP troubleshooting

```
ASA/C1# show arp statistics
        Number of ARP entries in ASA: 1

        Dropped blocks in ARP: 0
        Maximum Queued blocks: 3
        Queued blocks: 0
        Interface collision ARPs Received: 0
        ARP-defense Gratuitous ARPS sent: 0
        Total ARP retries: 14
        Unresolved hosts: 0
        Maximum Unresolved hosts: 1
```

Tracing Captured Packet

- Capture tool can record what actions were taken on a data packet in Accelerated Security Path
- This information is stored in trace buffers when the packet is processed
- The maximum number of buffers is 1000, the default is 50 for each capture
- “detail” option can be used to record more information about packet processing
- Unlike packet-tracer, this tool can be used to trace SYN and non-SYN packets, although information collected about non-SYN packets is very limited

Tracing Captured Packet Example

- Create a capture using the trace option

```
ASA/C1# capture cap-out trace detail trace-count 10 interface outside match tcp any host
192.0.2.1 eq 25
```

- Send traffic and verify that packets are captured

```
ASA/C1# show capture
capture cap-out type raw-data trace detail trace-count 10 interface outside [Capturing -
152 bytes]
  match tcp any host 192.0.2.1 eq smtp
```

- Display captured packets

```
ASA/C1# show capture cap-out

2 packets captured

   1: 21:21:30.236300       802.1Q vlan#76 P0 195.1.1.1.58135 > 192.0.2.1.25: S
2825729494:2825729494(0) win 4128 <mss 1460>
   2: 21:21:30.236926       802.1Q vlan#76 P0 192.0.2.1.25 > 195.1.1.1.58135: R 0:0(0) ack
2825729495 win 0
2 packets shown
```

Tracing Captured Packet Example

- Display information collected during packet processing
- The output looks the same as packet-tracer output

```
ASA/C1# show capture cap-out trace packet-number 1

2 packets captured

   1: 21:21:30.236300       802.1Q vlan#76 P0 195.1.1.1.58135 > 192.0.2.1.25: S
2825729494:2825729494(0) win 4128 <mss 1460>

Phase: 1
Type: CAPTURE
…
Phase: 2
Type: ACCESS-LIST
…
Phase: 8
Type: INSPECT
…
Phase: 9
Type: IDS
…
Phase: 13
Type: FLOW-CREATION
```

Note that counters increment in:

show service-policy inspect esmtp
show service-policy ips

This makes sense. Unlike packet-tracer, this is a real traffic and hence all ASP processing is real.

Tracing Captured Packet Example

- In this test, traffic was sent to IPS and dropped by a custom signature there, but the trace result is still “allow”:

```
Phase: 9
Type: IDS
Subtype:
Result: ALLOW
…
Result:
output-interface: dmz
output-status: up
output-line-status: up
Action: allow
```

```
signature: description=Drop all ESMTP traffic id=60000 created=20000101 type=other
version=custom

 actions:
    deniedPacket: true
    deniedFlow: true
    tcpOneWayResetSent: true
```

Tracing Captured Packet Example

- This is expected
- ASA only shares the copy of a packet with the inline IPS and requests the module to indicate its action (such as deny packet or reset TCP connection)
- While the module is coming up with the decision, all other packet processing tasks continue normally
- Before sending the packet out, the IPS decision is considered among other things
- Packet Tracing does not support this deferred decision logic, so the IDS phase will always be an "allow“, although the packet can be dropped
- What this means is that capture on ingress and egress ASA interfaces usually needs to be configured to make sure that traffic is either dropped or passed through the ASA

ASA – IPS Interactions

```
IPS: deny-connection-inline
ASA: %ASA-4-420002: IPS requested to drop TCP packet from … to …
ASA: %ASA-6-302014: Teardown TCP connection … Flow terminated by IPS
```

```
IPS: deny-packet-inline
ASA: %ASA-4-420002: IPS requested to drop ICMP packet from … to …
```

```
IPS: reset-tcp-connection
ASA: %ASA-4-420003: IPS requested to reset TCP connection from … to …
ASA: %ASA-6-302014: Teardown TCP connection … Flow reset by IPS
```

```
IPS: deny-attacker-inline
ASA: %ASA-4-420002: IPS requested to drop TCP packet from … to …
ASA: %ASA-6-302014: Teardown TCP connection … Flow terminated by IPS
```

- ASP drop counters are incremented on ASA:

```
ASA/C1# show asp drop

Frame drop:
  IPS Module requested drop (ips-request)                          4

Flow drop:
  Flow terminated by IPS (ips-request)                             8
```

ASA – IPS Interactions

- Of course, list of “denied attackers” is kept on IPS module and is used by IPS module to drop future packets coming from the attacker IP or IP and source/destination port combinations
  - deny-attacker-inline
  - deny-attacker-service-pair-inline
  - deny-attacker-victim-pair-inline

```
sensor# show statistics denied-attackers

Statistics for Virtual Sensor vs0
   Denied Attackers and hit count for each.
      195.1.1.1 = 4

ASA: %ASA-4-420002: IPS requested to drop TCP packet from … to …
```

- IPS module does not execute “shun” commands on ASA by default, but this can be configured
  - request-block-host
  - request-block-connection

Tracing Captured Packet Example

- It's possible to trace non-SYN packets, but note that very limited information is collected about them:

```
ASA/C1# show capture cap-out trace packet-number 5

Phase: 3
Type: FLOW-LOOKUP
Subtype:
Result: ALLOW
Config:
Additional Information:
Found flow with id 262421, using existing flow
Module information for forward flow ...
snp_fp_inspect_ip_options
snp_fp_tcp_normalizer
snp_fp_translate
snp_fp_adjacency
snp_fp_fragment
snp_ifc_stat

Module information for reverse flow ...

Result:
Action: allow
```

ASP Drop Capture

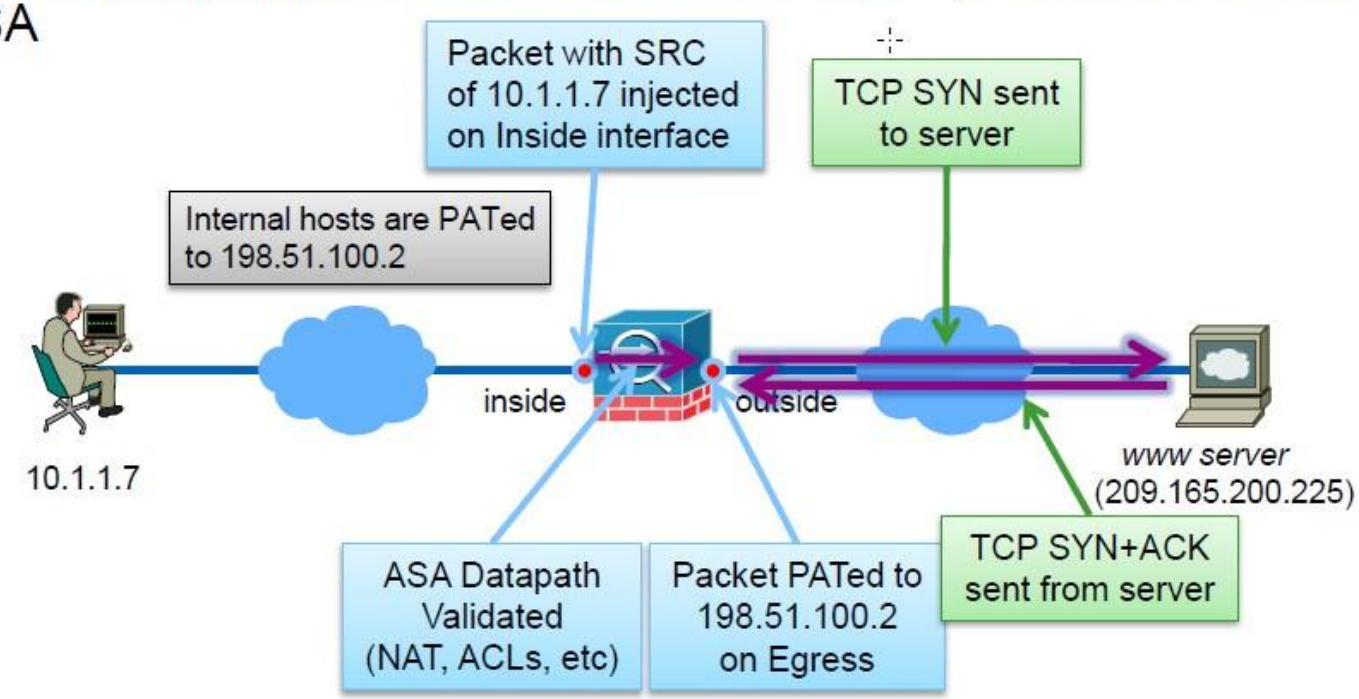
```
ASA/C1# capture cap1 type asp-drop {all | <specific-ASP-drop-code>}
```

- ASP Drop capture is used to capture packets dropped in Accelerated Security Path
- This capture can be run from both user context and system context
- If it is run from system context, all packets, dropped in user contexts, are captured
- Neither “access-list”, nor “match” filtering options work in ASP drop capture

TCP Ping

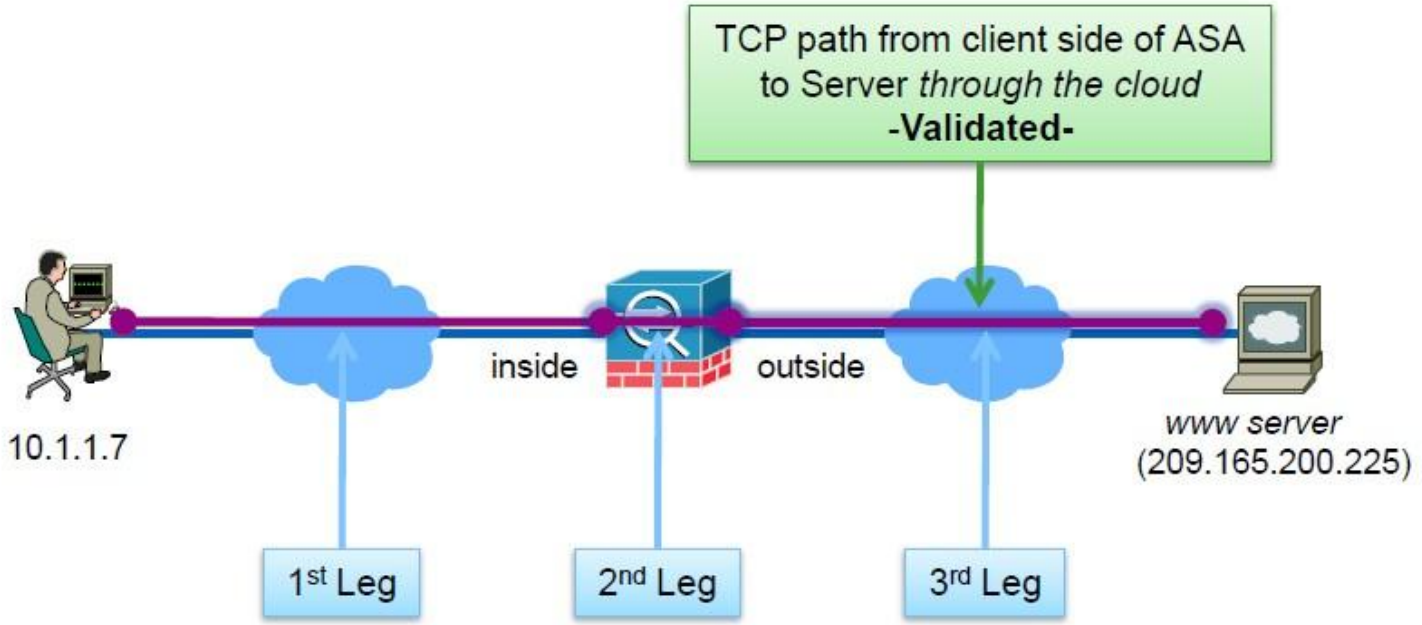
TCP ping

- Sources TCP SYN packet with *Client's IP* and injects it into *Client's interface* of the ASA



TCP ping

- Validates 2 of the 3 legs of the connection from client to server



TCP Ping Example

```
ASA/C1# ping tcp <input-interface> <destination-IP> <destination-port> source <source-IP>
<source-port> repeat <number-of-packets> timeout <timeout>
```

```
ASA/C1# ping tcp inside 195.1.1.1 23 source 10.1.1.2 1234 repeat 1

Type escape sequence to abort.
Sending 1 TCP SYN requests to 195.1.1.1 port 23
from 10.1.1.2 starting port 1234, timeout is 2 seconds:
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 2/2/2 ms
```

- ASA sends TCP RST to both IPs to terminate TCP connections at the end

Опрос #1

Опрос #1: если бы было нужно оставить на ASA только несколько фич, выкинув все остальные, то какие бы вы оставили из списка. Выберите 4 варианта из 9.

- NAT
- Identity Firewall (интеграция с AD Agent)
- Cloud Web Security (ScanSafe)
- Botnet Traffic Filtering
- Модуль IPS (аппаратный или программный в ASA5500-X)
- Модуль CX (аппаратный или программный в ASA5500-X)
- IPv6
- Clientless WebVPN
- ASA 9.0 Clustering

Case Study: Infected Local Host

Case Study: Infected Local Host

- Problem: The number of connections and xlates is little bit higher than usual during non-working hours
- There are no syslog messages at level 3 (errors)
- ASDM graphs do not show abnormal activity
- Threat Detection Statistics is not supported in multiple context mode (except TCP Intercept Statistics) and cannot be enabled

```
ASA/C1# show conn count
13648 in use, 13655 most used

ASA/C1# show xlate count
27682 in use, 27686 most used

ASA/C1# show nat pool
TCP PAT pool outside:PAT-POOL, address 192.0.2.12, range 1-511, allocated 0
TCP PAT pool outside:PAT-POOL, address 192.0.2.12, range 512-1023, allocated 0
TCP PAT pool outside:PAT-POOL, address 192.0.2.12, range 1024-65535, allocated 27653
TCP PAT pool outside, address 192.0.2.1, range 1-511, allocated 2
TCP PAT pool outside, address 192.0.2.1, range 512-1023, allocated 0
TCP PAT pool outside, address 192.0.2.1, range 1024-65535, allocated 0
```

Case Study: Infected Local Host

- Always look at the perfmon statistics first

```
ASA/C1# show perfmon detail

Context: C1
PERFMON STATS:                          Current       Average
Xlates                                   326/s           1/s
Connections                              326/s           3/s
TCP Conns                                326/s           3/s
UDP Conns                                  0/s           0/s
URL Access                                 0/s           0/s
URL Server Req                             0/s           0/s
TCP Fixup                                  0/s           0/s
TCP Intercept Established Conns            0/s           0/s
TCP Intercept Attempts                     0/s           0/s
TCP Embryonic Conns Timeout              212/s           3/s
HTTP Fixup                                 0/s           0/s
FTP Fixup                                  0/s           0/s
AAA Authen                                 0/s           0/s
AAA Author                                 0/s           0/s
AAA Account                                0/s           0/s

VALID CONNS RATE in TCP INTERCEPT:     Current       Average
                                         N/A           0.00%

SETUP RATES:
Connections for 1 minute = 455/s; 5 minutes = 118/s
TCP Conns for 1 minute = 455/s; 5 minutes = 118/s
UDP Conns for 1 minute = 0/s; 5 minutes = 0/s
```

Most TCP connections fail to establish.

This needs to be investigated.

Note that this statistics is collected even if TCP Intercept is not enabled on ASA.

Case Study: Infected Local Host

- Do we have hosts with high number of embryonic TCP connections?

```
ASA/C1# show local-host brief conn embryonic 100

Interface inside: 0 active, 0 maximum active, 0 denied
Interface outside: 13528 active, 43750 maximum active, 0 denied
Interface dmz: 1 active, 3 maximum active, 0 denied
```

- Hmm… This looks strange at first…
- We see a server in DMZ and 13K outside hosts, but none of them has more than 100 embryonic connections…

Case Study: Infected Local Host

- Take a look at the traffic load

```
ASA/C1# show traffic
dmz:
        received (in 208.540 secs):
                67334 packets    2693402 bytes
                322 pkts/sec     12915 bytes/sec
        transmitted (in 208.540 secs):
                8 packets        236 bytes
                0 pkts/sec       1 bytes/sec
      1 minute input rate 455 pkts/sec,  18204 bytes/sec
      1 minute output rate 0 pkts/sec,  0 bytes/sec
      1 minute drop rate, 0 pkts/sec
      5 minute input rate 118 pkts/sec,  4731 bytes/sec
      5 minute output rate 0 pkts/sec,  0 bytes/sec
      5 minute drop rate, 0 pkts/sec
outside:
        received (in 208.540 secs):
                0 packets        0 bytes
                0 pkts/sec       0 bytes/sec
        transmitted (in 208.540 secs):
                67326 packets    2693040 bytes
                322 pkts/sec     12913 bytes/sec
      1 minute input rate 0 pkts/sec,  0 bytes/sec
      1 minute output rate 455 pkts/sec,  18202 bytes/sec
      1 minute drop rate, 0 pkts/sec
      5 minute input rate 0 pkts/sec,  0 bytes/sec
      5 minute output rate 118 pkts/sec,  4730 bytes/sec
      5 minute drop rate, 0 pkts/sec
```

All traffic goes from DMZ interface to outside interface…

So, we have a host on the DMZ interface which opens 300+ connections per second.

Case Study: Infected Local Host

- Which hosts have many connections?

```
ASA/C1# show local-host brief conn tcp 100

Interface inside: 0 active, 0 maximum active, 0 denied
Interface outside: 13526 active, 43750 maximum active, 0 denied
Interface dmz: 1 active, 3 maximum active, 0 denied
local host: <172.16.1.2>,
    TCP flow count/limit = 13580/unlimited
    TCP embryonic count to host = 0
    TCP intercept watermark = unlimited
    UDP flow count/limit = 0/unlimited
```

- DMZ host identified. It has 13,580 connections (we don't know yet whether they are established or half-open connections)
- Important point here is that "local-host" data structure keeps track of "to-the-host" embryonic connections only
- This explains "show local-host brief conn embryonic 100" output, which is empty

Case Study: Infected Local Host

- Check if the host 172.16.1.2 has many half-open connections

```
ASA/C1# show conn long address 172.16.1.2 state tcp_embryonic
…
TCP outside: 207.196.96.126/139 (207.196.96.126/139) dmz: 172.16.1.2/1986
(192.0.2.12/1986), flags saA , idle 6s, uptime 6s, timeout 30s, bytes 0

TCP outside: 207.195.0.75/139 (207.195.0.75/139) dmz: 172.16.1.2/64812
(192.0.2.12/64812), flags saA , idle 12s, uptime 12s, timeout 30s, bytes 0

TCP outside: 207.53.11.165/139 (207.53.11.165/139) dmz: 172.16.1.2/64295
(192.0.2.12/64295), flags saA , idle 13s, uptime 13s, timeout 30s, bytes 0

TCP outside: 207.122.116.13/139 (207.122.116.13/139) dmz: 172.16.1.2/4377
(192.0.2.12/4377), flags saA , idle 0s, uptime 0s, timeout 30s, bytes 0
…
```

s – awaiting outside SYN
a - awaiting outside ACK to SYN
A - awaiting inside ACK to SYN

- It seems this host is infected by a virus and tries to establish TCP connections to random Internet hosts over TCP/139…

Case Study: Infected Local Host

- Solution: Limit the number of half-open connections?
- This won't work, because source IP address is not spoofed!
  - Limiting half-open connections enables TCP Intercept on ASA
  - As soon as the number of half-open connections reaches the limit, TCP Intercept kicks in and checks if the sender is real
  - It does this by sending TCP SYN/ACK to sender on behalf of the receiver:

```
207.19.80.102.139 > 172.16.1.2.1026: S 91583821:91583821(0) ack 960609065 win 0 <mss 536>
```

  - Connection is not allowed to go through if ACK is not received in response
  - Syslog messages are generated when connection limit is exceeded:

```
%ASA-6-201010: Embryonic connection limit exceeded 1000/1000 for input packet from
172.16.1.2/29449 to 207.154.160.149/139 on interface dmz

! Or in case of a per-host limit:

%ASA-6-201012: Per-client embryonic connection limit exceeded 1000/1000 for input packet
from 172.16.1.2/2024 to 207.12.10.49/139 on interface dmz
```

Case Study: Infected Local Host

- Solution: Limit the number of all connections!
- We can either set an aggregate limit (per MPF class) or a per-host limit

```
policy-map global_policy
 class dmz_hosts
  set connection per-client-max 1000

service-policy global_policy global
```

```
access-list dmz_hosts extended permit
tcp 172.16.1.0 255.255.255.0 any

class-map dmz_hosts
 match access-list dmz_hosts
```

- Syslog messages when the limit is exceeded:

```
%ASA-3-201013: Per-client connection limit exceeded 1000/1000 for input packet from
172.16.1.2/2026 to 207.37.27.155/139 on interface dmz

! Or in case of an aggregate limit:

%ASA-3-201011: Connection limit exceeded 1000/1000 for input packet from 172.16.1.2/19612
to 207.204.201.85/139 on interface dmz
```

Case Study: Infected Local Host

- The number of connections doesn't grow anymore:

```
ASA/C1# show conn count
1000 in use, 1000 most used

ASA/C1# show xlate count
2008 in use, 3008 most used
```

- Statistics:

```
ASA/C1# show service-policy set conn detail

Global policy:
  Service-policy: global_policy
    Class-map: dmz_hosts
      Set connection policy: per-client-max 1000
        current conns 1000, drop 63614
        Per client                  Embryonic     Total
        dmz         172.16.1.2      1000          1000
```

Case Study: A Note About Syslog

- Note that syslog messages 201010-201013 are not rate-limited by default. You may want to configure something like:

```
logging rate-limit 5 60 message 201010
logging rate-limit 5 60 message 201011
logging rate-limit 5 60 message 201012
logging rate-limit 5 60 message 201013
```

- Or you can set rate-limit for all messages under specified severity level. All messages will be rate-limited individually

```
logging rate-limit 1000 600 level 6
```

- It's not recommended to rate-limit levels 3 and above, as you can miss important system messages; unneeded level 3 and level 2 messages can be disabled or rate-limited individually
- Few messages should be moved from level 4 to level 2 due to their importance (will be discussed soon)

Case Study: Understanding Outputs

- In this example we protect DMZ server with TCP Intercept
  - set connection embryonic-conn-max 2
  - Three HTTP connections were opened with spoofed source IP
  - After that two SMTP connections were opened

```
ASA/C1# show local-host
…
Interface dmz: 1 active, 3 maximum active, 0 denied
local host: <172.16.1.2>,
    TCP flow count/limit = 4/unlimited
    TCP embryonic count to host = 2
    TCP intercept watermark = unlimited
    UDP flow count/limit = 0/unlimited


  Conn:
    TCP outside  195.1.1.1:64303 dmz  172.16.1.2:25, idle 0:00:02, bytes 0, flags UB
    TCP outside  195.1.1.1:34811 dmz  172.16.1.2:25, idle 0:00:03, bytes 0, flags UB
    TCP outside  195.1.1.1:54063 dmz  172.16.1.2:80, idle 0:00:06, bytes 0, flags aB
    TCP outside  195.1.1.1:24611 dmz  172.16.1.2:80, idle 0:00:09, bytes 0, flags aB
```

Four connection entries were created (2 + 2)

Three connections were intercepted (1 + 2)
Three %ASA-6-201010 syslogs produced
One intercepted connection dropped (1)

Limits are only valid in local-host structure in 8.2 and below, when “static” or “nat” commands are used to set them

Case Study: Understanding Outputs

- In this example we protect DMZ server with TCP Intercept
  - set connection embryonic-conn-max 2
  - Three HTTP connections were opened with spoofed source IP
  - After that two SMTP connections were opened

```
ASA/C1# show service-policy int outside set connection detail

Interface outside:
  Service-policy: outside_policy
    Class-map: to_dmz_server
      Set connection policy: embryonic-conn-max 2
        current embryonic conns 2, drop 0
```

So, one connection was dropped by Intercept.

But drop counter was not updated.

Case Study: Final Notes

- Note that connection limits are very important to prevent depletion of resources
  - PAT pools can be depleted, unless the number of connections is limited
  - ASA can run out of memory if too many connections are established
  - High CPU can sometimes be observed
- Choose the right tool depending on situation
  - Aggregate and per-host embryonic connection limits are important to protect internal servers from SYN flood attacks and outside hosts from SYN scans generated by inside hosts
  - Aggregate connection limits are useful to protect internal servers from overloading
  - Per-host connection limits are useful to impose restrictions on your local users
  - Aggregate connection limits can help protect ASA itself from resource depletion

Resource Management

ASA Resources

- Most critical resources are:
  - Heap memory
  - DMA memory
  - CPU
  - Connection slots
  - Slots in NAT/PAT pools
- Most ASA models have enough DRAM and CPU power to run well under heavy traffic load with the number of concurrent connections defined in the specification:
  - http://www.cisco.com/en/US/products/ps6120/prod_models_comparison.html

Example

- ASA5555 supports up to 1M connections and 50K conn setup rate
- As you can see below, 934,895 connections are established and 47% of memory is still free, CPU load is negligible

```
ASA# show resource usage
Resource                 Current         Peak      Limit         Denied Context
Syslogs [rate]                 0            4  unlimited              0 admin
Inspects [rate]                0            2  unlimited              0 admin
Routes                         2            2  unlimited              0 admin
Conns                     934895       934895  unlimited              0 C1
Xlates                    934903       934903  unlimited              0 C1
Hosts                    1718768      1718768  unlimited              0 C1
Conns [rate]                6449         8869  unlimited              0 C1
Routes                         5            5  unlimited              0 C1

ASA# show memory
Free memory:        4031382464 bytes (47%)
Used memory:        4558552128 bytes (53%)
-------------      ------------------
Total memory:       8589934592 bytes (100%)

ASA# show cpu detailed

Break down of per-core data path versus control point cpu usage:
Core         5 sec               1 min               5 min
Core 0       16.6 (15.6 + 1.0)   20.1 (20.0 + 0.1)   13.4 (13.3 + 0.0)
Core 1       17.0 (16.0 + 1.0)   20.3 (20.2 + 0.1)   13.1 (13.1 + 0.0)
```

ASA Resources

- Note, however, that some ASA features can allocate lots of memory and significantly increase CPU load
- This always depends on traffic profile
- For example, what works well on an ASA located inside your corporate network may not work that well on network perimeter, due to the risk of DoS attacks
  - Memory usage increases if traffic is inspected at application layer
  - Extended PAT can allocate lots of memory for PAT pools
  - Threat Detection Statistics can be memory and CPU intensive
  - Other subsystems, such as WebVPN, can consume lots of memory
  - SMP platforms use per-core application caches, ASA5585 SSP-60 has 24 CPU cores…
  - Etc.

ASA Resources

- Bottom line: it's very important to protect ASA itself from resource depletion
  - We've already discussed MPF connection limits
  - Another tool that can help is Resource Manager (RM)
- Resource Manager
  - Limits the number of concurrent connections in single context mode
  - Every platform has its own default connection limit
  - In multiple context mode administrators can create resource classes to restrict context access to system resources
  - System-wide connection limit is still enforced in multiple context mode

```
%ASA-5-321001: Resource 'conns' limit of 1001000 reached for system
```

For your reference

Default Platform Connection Limits

Model	ASA DRAM	Connection Limit
5512-X Base License	2,048	100,000
5512-X Sec+ License	2,048	250,000
5515-X	4,096	251,000
5525-X	4,096	500,000
5545-X	6,144	750,000
5555-X	8,192	1,000,000
5580-20	8,192	2,000,000
5580-40	12,288	4,000,000
5585 SSP-10	6,144	1,000,000
5585 SSP-20	12,288	2,000,000
5585 SSP-40	12,288	4,000,000
5585 SSP-60	24,576	10,000,000

Resource Manager Configuration

```
class TEST
  limit-resource Conns 30.0%

context C1
  member TEST
  allocate-interface GigabitEthernet0/1.75-GigabitEthernet0/1.76
  allocate-interface GigabitEthernet0/1.103
  allocate-ips vs0 default
  config-url disk0:/C1.cfg

ASA# show run all class
class default
  limit-resource All 0
  limit-resource Mac-addresses 65535
  limit-resource ASDM 5
  limit-resource SSH 5
  limit-resource Telnet 5

class TEST
  limit-resource Conns 30.0%

ASA# show resource allocation
Resource                    Total          % of Avail
 Conns [rate]           unlimited
 Inspects [rate]        unlimited
 Syslogs [rate]         unlimited
 Conns                  300300(U)            30.00%
 Hosts                  unlimited
 SSH                           10            10.00%
…
U = Unlimited: Some contexts have no limit and are not included in the total
```

Note that "class default" limits are inherited by all contexts, unless the limit is overridden explicitly by another resource class.

For example, each context is restricted to 5 SSH sessions, 5 telnet sessions, etc.

This command shows how many system resources are allocated to all contexts (in total)

Resource Manager Configuration

- Use this command to see current and peak resource usage for each context

```
ASA# show resource usage
Resource                  Current         Peak      Limit         Denied Context
Syslogs [rate]                  0            6  unlimited              0 admin
Inspects [rate]                 0            3  unlimited              0 admin
Routes                          2            2  unlimited              0 admin
Syslogs [rate]                  0            7  unlimited              0 C1
Conns                      300300       300300     299300          31320 C1
Xlates                     331626       331626  unlimited              0 C1
Hosts                      609585       609585  unlimited              0 C1
Conns [rate]                 3598         4977  unlimited              0 C1
```

- Use “show resource usage system” to see totals
- Syslog message will be generated in the admin context should a user context go over limit

```
system : %ASA-5-321001: Resource 'conns' limit of 300300 reached for context 'C1'
```

Resource Manager Configuration

- There are many different types of resources available

```
ASA(config)# class TEST
ASA(config-class)# limit-resource ?

Following resources available:
  ASDM            ASDM Connections
  All             All Resources
  Conns           Connections
  Hosts           Hosts
  Mac-addresses   MAC Address table entries
  Routes          Routing Table Entries
  SSH             SSH Sessions
  Telnet          Telnet Sessions
  VPN             VPN resources
  Xlates          XLATE Objects

ASA(config-class)# limit-resource rate ?

class mode commands/options:
Following resources available:
  Conns      Connections/sec
  Inspects   Inspects/sec
  Syslogs    Syslogs/sec
```

Resource Manager Syslogs

- Resource Manager generates the following syslogs when a limit is reached
  - %ASA-5-321001: Resource … limit of … reached
  - %ASA-5-321002: Resource … rate limit of … reached
  - %ASA-6-321003: Resource … log level of … reached
  - %ASA-6-321004: Resource … rate log level of … reached
- Note that these messages are not rate-limited by default and have low severity for unknown reasons
- Several other syslogs were introduced in 8.4(1) (read on!)

Monitoring Resource Utilization

Monitoring Resource Utilization

- There are many different ways to monitor memory, CPU usage, etc.
  - Periodic data collection via Unix Expect scripts or SecureCRT VB scripts
  - Periodic data collection via Smart Call Home
  - Periodic data collection via EEM (will be available soon on ASA)
  - Syslog events
  - SNMP polling
  - SNMP traps
- Significant enhancements were made in 8.4(1), new syslog events and SNMP traps were introduced
- ASA MIBs are beyond the scope of this presentation. Refer to:
  - http://www.cisco.com/en/US/docs/security/asa/asa91/configuration/general/monitor_snmp.html
  - ftp://ftp.cisco.com/pub/mibs/supportlists/asa/asa-supportlist.html

Memory Utilization: Terminology

- ASA system memory is divided into Global Shared Pool and DMA memory
- DMA memory is used for packet processing and also used by various ASA processes:
  - Syslog
  - ASA HTTP Server (ASDM)
  - WebVPN (Clientless/AnyConnect)
  - SSH
  - IPSec (IKEv1/IKEv2)
  - Etc.
- Global Shared Pool is a general purpose memory (or heap)

Memory Utilization Guidelines

- Note that high memory utilization on ASA doesn't always mean that something is wrong and ASA cannot service new connections and create xlates
- Memory can be held in application caches in anticipation for another traffic spike
- On the other hand, low free heap memory can be a indication of a memory leak, device oversubscription or a DoS attack
- Another possible problem with the heap memory is a memory fragmentation
- Low-end platforms, such as ASA5505, ASA5510, ASA5520 may experience shortage of DMA memory when many syslog destinations and other features are configured
- Always open TAC case if in trouble

Memory Utilization: Understanding Output

- Modern software versions should display the same amount of free memory in “show memory” and “show memory detail”, although always use “detail” option if in doubt

```
ASA# show memory
Free memory:        6804260944 bytes (79%)
Used memory:        1785673648 bytes (21%)
-------------      ------------------
Total memory:       8589934592 bytes (100%)

ASA# show memory detail
Free memory:                       6804260944 bytes (79%)
Used memory:
     Allocated memory in use:       526686128 bytes ( 6%)
     Reserved memory:              1258987520 bytes (15%)
----------------------------    ------------------
Total memory:                      8589934592 bytes (100%)

Least free memory:        5741150400 bytes (67%)
Most used memory:         2848784192 bytes (33%)

MEMPOOL_DMA POOL STATS:
…
MEMPOOL_GLOBAL_SHARED POOL STATS:
…
Max contiguous free mem      =    5741473552
Allocated memory in use      =     526686128
Free memory                  =    6804260944
```

Free heap memory

Low and high watermarks

DMA and Heap Pools

Memory is fragmented if this value is significantly lower than the amount of free memory

Memory Utilization: Example

- Example: ASA running low on heap memory, almost all DMA memory is free

```
ASA# show memory detail
Free memory:                    286432880 bytes ( 3%)
Used memory:
     Allocated memory in use:     7044514192 bytes (82%)
     Reserved memory:             1258987520 bytes (15%)
----------------------------   ------------------
Total memory:                    8589934592 bytes (100%)

Least free memory:            118976 bytes ( 0%)
Most used memory:        8589815616 bytes (100%)

MEMPOOL_DMA POOL STATS:
…
Free memory                =    253327520
…
MEMPOOL_GLOBAL_SHARED POOL STATS:
…
Max contiguous free mem    =         3552
Allocated memory in use    =   7330242576
Free memory                =       704496
```

Memory Utilization: Error Messages

- Generic syslog message:

```
%ASA-3-211001: Memory allocation Error
```

- Sample console messages:

```
process_create: out of stack memory
Unable to create Unicorn Admin Handler

process_create: out of memory
_listen_telnet: failed to create thread for interface 65537 port 23

ERROR: Unable to allocate memory for usage display

Out of memory, cannot allocate memory for log message.
```

- Use the following command to check console output:

```
ASA# show console-output
```

Memory Utilization: Error Messages

- New syslog was introduced in 8.4(1). It is produced in admin context when system memory usage reaches hardcoded value of 80% and stays there for a period of 5 minutes

```
%ASA-2-321006: System Memory usage reached 89%
```

- Also, it is now possible to send SNMP trap in this case

```
snmp-server enable traps memory-threshold
```

- DISMAN-EVENT-MIB is used to send the trap
- CISCO-ENHANCED-MEMPOOL-MIB can be used for polling (since 8.4(1))
  - In multiple context mode use admin context for polling
  - First row returned should report on admin memory usage; second row should return information for the whole system

Blocks Utilization

- Blocks are packet buffers mostly used to hold packets
  - 1550 Byte blocks are used for Ethernet frames
  - 9216 Byte blocks are used for jumbo Ethernet frames (disabled by default)
  - 2048 Byte blocks are used by ASA5505 Ethernet driver
  - Etc.
- ASA traffic forwarding and associated performance issues is a extremely overcomplicated subject
  - For details refer to fantastic BRKSEC-3021 Networkers session led by Andrew Ossipov
  - http://www.ciscolive.com/global/
- Remember that “show blocks” command shows you just the tip of the iceberg…
- In ASA 8.4(1) new syslog message was introduced:

```
%ASA-3-321007: System is low on free memory blocks of size 1550 (1 CNT out of 30000 MAX)
```

CPU Utilization

- The following output was taken on the ASA5555 (which has two CPU cores) and edited a bit to make it easier to read
- Note how Data Path and Control Point utilization add up

```
ASA# show cpu detailed

Break down of per-core data path versus control point cpu usage:
Core          5 sec                 1 min                 5 min
Core 0        47.6 (17.6 + 30.0)    21.4 (12.4 + 9.0)     5.7 (3.4 + 2.3)
Core 1        47.0 (17.8 + 29.2)    21.5 (12.6 + 8.9)     5.7 (3.5 + 2.2)


Total CPU utilization for:
      5 seconds = 47.5%; 1 minute: 21.5%; 5 minutes: 5.7%

ASA# show processes cpu-usage sorted non-zero
PC          Thread       5Sec     1Min     5Min    Process
   -           -          9.0%     6.4%     1.8%   DATAPATH-1-1533
   -           -          8.8%     6.2%     1.7%   DATAPATH-0-1532
                          1.4%     0.3%     0.1%   CP Processing
                          1.2%     0.3%     0.1%   Logger
                         27.1%     8.4%     2.1%   ci/console
```

High CPU in ci/console was caused by running "show …" | include

DP utilization

CP utilization

CPU Utilization: CPU Hogs

- CPU hog events are recorded by ASA when a process runs on the CPU longer than the minimum platform threshold. Function stack (traceback) is also recorded
- Such events are usually benign, unless the hog lasts longer than 10-20 ms or NUMHOG counter increments rapidly for the process

```
ASA# show processes cpu-hog

Process:       CP Threat-Detection Processing, NUMHOG: 30, MAXHOG: 9, LASTHOG: 2
LASTHOG At:    12:47:29 MSK Jan 13 2014
PC:            0x000000000079a06f (suspend)
Call stack:    0x000000000079a06f  0x0000000000428d45

CPU hog threshold (msec):  1.542
Last cleared: None
```

Number of CPU hogs

Maximum CPU hog in milliseconds

Last CPU hog in milliseconds

Use "clear process cpu-hog" to clear the table

- Syslog messages are produced in admin context

```
%ASA-4-711004: Task ran for 9 msec, Process = CP Threat-Detection Processing, PC = …, Call
stack = …
```

CPU Utilization: Error Messages

- New syslog was introduced in 8.4(1). It is produced in admin context when CPU utilization reaches 95% or more and stays there for a period of 5 minutes

```
%ASA-2-321005: System CPU utilization reached 95%
```

- Also, it is now possible to send SNMP traps when either “high” threshold (70% by default) or hardcoded threshold (95%) is crossed
- The default monitoring period for “high” threshold is one minute, which means that CPU utilization should remain above the threshold during this period to produce the trap

```
snmp cpu trap threshold rising <10-94%> <minutes>
snmp-server enable traps cpu threshold rising
```

- CISCO-PROCESS-MIB is used to send the trap and can also be used for polling
  - Same rules apply as for memory polling

DP – CP Queues

- DP–CP queues are used to punt “packets” to CP from Data Path

```
ASA/C1# show asp event dp-cp

DP-CP EVENT QUEUE                  QUEUE-LEN  HIGH-WATER
Punt Event Queue                           0           1
Routing Event Queue                        0           0
Identity-Traffic Event Queue               0           1
General Event Queue                        0           1
Syslog Event Queue                         0           3
Non-Blocking Event Queue                   0           4
Midpath High Event Queue                   0           0
Midpath Norm Event Queue                   0           3
Crypto Event Queue                         0          61
SRTP Event Queue                           0           0
HA Event Queue                             0           0
Threat-Detection Event Queue               0           0
SCP Event Queue                            0           0
ARP Event Queue                            0           9
IDFW Event Queue                           0           0
CXSC Event Queue                           0           0

EVENT-TYPE          ALLOC ALLOC-FAIL ENQUEUED ENQ-FAIL  RETIRED 15SEC-RATE
punt                   54          0       54        0       54          0
  inspect-smtp         54          0       54        0       54          0
arp-in                734          0      734        0      734          2
identity-traffic        5          0        5        0        5          0
syslog                165          0      165        0      165          1
ips-cplane            433          0      433        0      433          0
```

DP – CP Queues

- The above command can be helpful when inspected traffic causes high CPU in CP Thread, as it shows 15 sec. rate
- DP–CP statistics can be cleared by “clear asp event dp-cp”
- Also use “show service-policy” to double-check

```
ASA/C1# show service-policy inspect esmtp

Global policy:
  Service-policy: global_policy
    Class-map: inspection_default
      Inspect: esmtp _default_esmtp_map, packet 54, lock fail 0, drop 0, reset-drop 0, v6-
fail-close 0
```

- DP–CP queues have limited depth, ENQ-FAIL increments when the queue is full and syslog message is produced. Examples:

```
%ASA-4-447001: ASP DP to CP Punt Event Queue was full. Queue length 2048, limit 2048
%ASA-4-447001: ASP DP to CP General Event Queue was full. Queue length 8192, limit 8192
```

DP – CP Queues

- Below is the list of ASA features which do not require punting to CP

```
ASA# show asp multiprocessor accelerated-features

MultiProcessor accelerated feature list:
        Access Lists
        DNS Guard
        Failover Stateful Updates
        Flow Operations(create, update,
                        and tear-down)
        Inspect HTTP URL Logging
        Inspect HTTP (AIC)
        Inspect IPSec Pass through
        Inspect ICMP and ICMP error
        Inspect RTP/RTCP
        IP Audit
        IP Fragmentation & Re-assembly
        IPSec data-path
        MPF L2-L4 Classify
        Multicast forwarding
        NAT/PAT
        Netflow using UDP transport
        Non-AIC Inspect DNS
        Packet Capture
        QOS
        Resource Management
        Routing Lookup
        Shun
        SSL data-path
        Syslogging using UDP transport
        TCP Intercept
        TCP Security Engine
        TCP Transport
        Threat Detection
        Unicast RPF
        WCCP Re-direct

Above list applies to routed, transparent, single and multi mode.
```

NAT/PAT Pools Utilization

- NAT on ASA is another complicated subject
- Troubleshooting tool include:
  - "show nat [detail]" command, which prints NAT rule table
  - Set of "show asp table classify" commands which display NAT rules downloaded to softNP
  - "show nat pool" command which prints information about utilization of NAT/PAT pools
  - Various syslog messages
  - NAT-MIB introduced in 8.4(1)
- In general, NAT troubleshooting is beyond the scope of this presentation
- We will only talk about PAT scalability

NAT/PAT Scalability

- Example: PAT pool is completely exhausted

```
ASA/C1# show nat pool
TCP PAT pool outside:obj-192.0.2.3, address 192.0.2.3, range 1-511, allocated 511
TCP PAT pool outside:obj-192.0.2.3, address 192.0.2.3, range 512-1023, allocated 512
TCP PAT pool outside:obj-192.0.2.3, address 192.0.2.3, range 1024-65535, allocated 64512
```

- Remember that it is possible to have only 64K TCP xlates and 64K UDP xlates for a single global IP in a PAT pool and low ports cannot be used if a sender source port is >1023
  - Low ports can be made available by “flat include-reserve” NAT option (8.4(3))
- Also, take into account that xlates live for a period of 30 seconds by default after associated TCP or UDP connection is closed
  - “timeout pat-xlate …” minimum value is 30 seconds
- This means that the maximum translation rate for a single PAT address for one IP protocol is about to 2100 xlates/sec (64K / 30)

NAT/PAT Troubleshooting

- Also, note that syslogs %ASA-3-305006 and %ASA-3-202010 (8.4(1)) do not necessarily indicate that the NAT or PAT pool is exhausted

```
%ASA-3-305006: portmap translation creation failed for tcp src dmz:172.16.1.2/3329 dst
outside:207.155.110.226/80

%ASA-3-202010: PAT pool exhausted. Unable to create TCP connection from
dmz:172.16.1.2/13171 to outside:207.54.99.10/80
```

- Such messages can be produced due to a completely different NAT issue or due to a software bug, although %ASA-3-202010 usually tells truth
- So, troubleshooting is difficult

NAT/PAT Scalability Options

- There are three options to improve PAT scalability
  - Use several global IPs in a PAT pool, instead of a single one
  - Use extended PAT (“pat-pool … extended”) – 8.4(3)
  - Use per-session PAT – 9.0(1)
- 1st option is bulletproof, but not all customers have many global IPs
- 2nd option cannot be recommended
  - It is incompatible with VoIP inspects: http://www.cisco.com/en/US/docs/security/asa/asa91/configuration/firewall/inspect_overview.html#wpxref53568
  - It can cause more problems than it solves due to excessive memory usage
  - It solves scalability issue by dynamically allocating a new PAT pool for each new destination IP, so two local hosts can be translated to the same global IP / global port if they go to different destinations
- 3rd one remains

Per-session PAT Configuration Example

```
ASA/C1# show run all xlate

xlate per-session permit tcp any4 any4
xlate per-session permit tcp any4 any4
xlate per-session permit tcp any4 any6
xlate per-session permit tcp any6 any4
xlate per-session permit tcp any6 any6
xlate per-session permit udp any4 any4 eq domain
xlate per-session permit udp any4 any6 eq domain
xlate per-session permit udp any6 any4 eq domain
xlate per-session permit udp any6 any6 eq domain
```

Note that per-session PAT is disabled automatically if you upgrade from a previous version.

It is enabled by default in new setups.

You can create your own per-session PAT rules and they will be placed above default system rules.

'x' flag is added to connection entry ("show conn long") when user traffic hits per-session PAT permit rule.

- This feature was created for ASA clustering, but can also help in non-clustering setups. It is only used by dynamic PAT
- Per-session PAT disables "timeout pat-xlate" for matching traffic
- Note, however, that PAT collisions can occur if two internal hosts go to the same server/port through the ASA PAT one after another.
  - ASA can reuse the same mapped IP/port for the 2nd connection. Should the server still keep the 1st connection in the TIME_WAIT state it would refuse the 2nd one

Conclusion

Conclusion

- Protect ASA to help it protect your network
- Baseline CPU load, connection counts, xlate counts, and traffic
- Set embryonic and maximum connection limits via MPF, use Resource Manager in multiple context mode
- Perform monitoring via syslog, SNMP, ASDM graphs
- Log at level 3 (errors) or 2 (critical), move important messages to level 2, disable unneeded syslog messages, rate-limit messages if necessary
- Follow KISS principle, don’t enable features unless you really need them and clearly understand what they do
- Use failover, but don’t set millisecond failover timers
- Run the latest maintenance release in your software train
- Upgrade major feature trains only when you need new features, or after train has matured

Using Smart Call Home for Monitoring

- Example: Collecting memory utilization every hour:

```
service call-home
 call-home
  alert-group-config snapshot
   add-command "show conn count"
   add-command "show memory detail"
  contact-email-addr user@cisco.com
  sender from user@cisco.com
  sender reply-to user@cisco.com
  mail-server <email_server> priority 1
 profile TAC
  active
  destination address email user@cisco.com
  destination transport-method email
  destination preferred-msg-format long-text
  subscribe-to-alert-group snapshot periodic interval 60
```

- More info:
  - http://www.cisco.com/en/US/docs/security/asa/asa91/configuration/general/monitor_smart_call_home.html
  - https://supportforums.cisco.com/community/netpro/solutions/smart_services/smartcallhome

Recommended Reading

Recommended Reading

Опрос #2

Опрос #2: Что Cisco должна изменить в первую очередь в своих продуктах, имеющих отношение к информационной безопасности. Выберите не более 5 вариантов из 8

- Отказаться от функций типа firewall на маршрутизаторах и сосредоточиться на специализированных устройствах вроде ASA
- Отказаться от специализированных устройств и реализовать все функции безопасности на маршрутизаторах и коммутаторах
- Создать специализированное решение для защиты от DoS и DDoS атак
- Сосредоточить основные усилия на развитие Application Firewall (межсетевых экранах “нового поколения” – NGFW), например, ASA CX
- Синтегрировать развитый Application Firewall и традиционный Firewall в едином программном коде на платформе ASA
- Кардинально улучшить функциональные возможности системы IDS/IPS
- Кардинально улучшить масштабируемость решений на основе Cisco ASA, например, за счет доработки и функционального насыщения ASA clustering
- Реализовать единый графический интерфейс управления всеми устройствами безопасности

Q & A

Эксперт ответит на некоторые Ваши вопросы. Используйте Q&A панель, чтобы задать еще вопросы

Сессия «Спросить Эксперта»

Получить дополнительную информацию, а также задать вопросы экспертам в рамках данной темы вы можете в течение двух недель, на странице, доступной по ссылке

https://supportforums.cisco.com/community/russian/expert-corner

Вы можете получить видеозапись данного семинара и текст сессии Q&A в течении ближайших 5 дней по следующей ссылке

https://supportforums.cisco.com/community/russian/expert-corner/webcast

Next Expert Series Webcast на Русском

Тема: Расширенные возможности Cisco Unified Border Element. Настройка, поиск и устранение неисправностей

во вторник, 11 февраля, в

12.00 Moscow Time

Присоединяйтесь к эксперту Cisco **Владимиру Савостину**

Во время презентации эксперт Cisco TAC Владимир Савостин рассмотрит некоторые возможности Cisco UBE, как широко известные и часто используемые, так и новые, недавно появившиеся. Также Вы узнаете о настройке, поиске и устранении проблем при использовании данного функционала.

Регистрируйтесь на вебкаст по ссылке:

http://tools.cisco.com/gems/cust/customerSite.do?METHOD=E&LANGUAGE_ID=R&SEMINAR_CODE=S19807&PRIORITY_CODE

Приглашаем Вас активно участвовать в Cisco Support Community и социальных сетях

https://supportforms.cisco.com/community/russian

http://www.facebook.com/CiscoRu

http://twitter.com/CiscoRussia

http://www.youtube.com/user/CiscoRussiaMedia

http://itunes.apple.com/us/app/cisco-technical-support/id398104252?mt=8

http://www.linkedin.com/groups/CSC-Cisco-Support-Community-3210019

Newsletter Subscription:
https://tools.cisco.com/gdrp/coiga/showsurvey.do?surveyCode=589&keyCode=146298_2&PHYSICAL%20FULFILLMENT%20Y/N=NO&SUBSCRIPTION%20CENTER=YES

Спасибо за Ваше время

Пожалуйста, участвуйте в опросе

Thank you.