

Firepower Data Path Troubleshooting Phase 8: Network Analysis Policy

Contents

[Introduction](#)

[Prerequisites](#)

[Troubleshooting the Network Analysis Policy Feature](#)

[Using the "trace" Tool to Find Preprocessor Drops \(FTD Only\)](#)

[Verify NAP Configuration](#)

[View NAP Settings](#)

[NAP Settings That Can Cause Silent Drops](#)

[Verify the Backend Configuration](#)

[Creating a Targeted NAP](#)

[False Positive Analysis](#)

[Mitigation Steps](#)

[Data to Provide to TAC](#)

Introduction

This article is part of a series of articles which explain how to systematically troubleshoot the data path on Firepower systems to determine whether components of Firepower may be affecting traffic. Please refer to the [Overview article](#) for information about the architecture of Firepower platforms and links to the other Data Path Troubleshooting articles.

This article covers the eighth stage of the Firepower data path troubleshooting, the Network Analysis Policy feature.



Prerequisites

- This article is applicable to all Firepower platforms
The **trace** feature is only available in software version 6.2.0 and above for the Firepower Threat Defense (FTD) platform only.
- Knowledge of open source Snort is helpful, though not required For information on open source Snort, please visit <https://www.snort.org/>

Troubleshooting the Network Analysis Policy Feature

The Network Analysis Policy (NAP) contains snort preprocessor settings which perform inspections on traffic, based on the application identified. The preprocessors have the ability to

drop traffic, based on configuration. This article addresses how to verify the NAP configuration and check for preprocessor drops.

Note: Preprocessor rules have a Generator ID (GID) other than '1' or '3' (i.e. 129, 119, 124). More information about the GID to preprocessor mappings can be found in the [FMC Configuration Guides](#).

Using the "trace" Tool to Find Preprocessor Drops (FTD Only)

The **system support trace** tool can be used to detect drops performed at the preprocessor level.

In the example below, the TCP normalization preprocessor detected an anomaly. As a result, the traffic is dropped by rule **129:14**, which looks for missing timestamps within a TCP stream.

```
> system support trace
[omitted for brevity...]
172.16.111.226-51174 - 50.19.123.95-443 6 Packet: TCP, ACK, seq 3849839667, ack 1666843207
172.16.111.226-51174 - 50.19.123.95-443 6 Stream: TCP normalization error in timestamp, window, seq, ack, fin, flags, or unexpected data, drop
172.16.111.226-51174 - 50.19.123.95-443 6 AppID: service unknown (0), application unknown (0)
172.16.111.226-51174 > 50.19.123.95-443 6 AS 4 | 0 Starting with minimum 3, 'block urls', and SrcZone first with zones -1 -> -1, geo 0 -> 0, vlan 0, inline sgt tag: untagged, ISE sgt id: 0, svc 0, payload 0, client 0, misc 0, user 9999997, icmpType 0, icmpCode 0
172.16.111.226-51174 > 50.19.123.95-443 6 Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, sgt 65535, user 9999997, icmpType 0, icmpCode 0
172.16.111.226-51174 > 50.19.123.95-443 6 AS 4 | 0 pending rule order 3, 'block urls', URL
172.16.111.226-51174 > 50.19.123.95-443 6 Firewall: pending rule-matching, 'block urls', pending URL
172.16.111.226-51174 > 50.19.123.95-443 6 Snort: processed decoder alerts or actions queue, drop
172.16.111.226-51174 > 50.19.123.95-443 6 IPS Event: gid 129, sid 14, drop
172.16.111.226-51174 > 50.19.123.95-443 6 NAP id 1, IPS id 0, Verdict BLOCK
172.16.111.226-51174 > 50.19.123.95-443 6 ==> Blocked by Stream
```

Note: Although the **TCP Stream Configuration** pre-processor drops the traffic, it is able to do so because the **Inline Normalization** preprocessor is also enabled. For more on Inline Normalization, you can read this [article](#).

Verify NAP Configuration

On the Firepower Management Center (FMC) UI, the NAP can be viewed under **Policies > Access Control > Intrusion**. Then, click on the **Network Analysis Policy** option in the top right, after which you can view the NAPs, create new ones and edit existing ones.

Deploy System Help admin

Import/Export Intrusion Rules Access Control **Network Analysis Policy**

Policy Information

Name: My Custom NAP

Description:

Inline Mode

Inline Result | **Source IP** | **Destination IP** | **Source Port / ICMP Type** | **Destination Port / ICMP Code** | **Message**

Inline Result	Source IP	Destination IP	Source Port / ICMP Type	Destination Port / ICMP Code	Message
	172.16.111.226	50.19.123.95	51177 / tcp	443 (https) / tcp	STREAM5_NO_TIMESTAMP (129:14:2)
	172.16.111.226	50.19.123.95	51174 / tcp	443 (https) / tcp	STREAM5_NO_TIMESTAMP (129:14:2)

Annotations:

- Edit or create a Network Analysis Policy
- Uncheck this box to disable Inline Mode
- Inline Mode disabled = No Inline Result
- Inline Mode enabled = "Dropped" Inline Result

As seen in the illustration above, the NAPs contain an "Inline Mode" feature, which is the equivalent of the "Drop When Inline" option in the Intrusion Policy. A quick mitigation step for preventing the NAP from dropping traffic would be to uncheck **Inline Mode**. The Intrusion Events generated by the NAP don't display anything in the **Inline Result** tab with **Inline Mode** disabled.

View NAP Settings

Within the NAP, you can view the current settings. This includes the total enabled preprocessors, followed by the

preprocessors enabled with non-default settings (ones which were manually tweaked) and ones which are enabled with default settings, as seen in the illustration below.

Edit Policy: My Custom NAP

Policy Information

Settings

Application Layer Preprocessors

- DCE/RPC Configuration: Enabled
- DNS Configuration: Enabled
- FTP and Telnet Configuration: Enabled
- HTTP Configuration: Enabled
- Sun RPC Configuration: Enabled
- SIP Configuration: Enabled
- GTP Command Channel Configuration: Enabled
- IMAP Configuration: Enabled
- POP Configuration: Enabled
- SMTP Configuration: Enabled
- SSH Configuration: Enabled
- SSL Configuration: Enabled

SCADA Preprocessors

- Modbus Configuration: Enabled
- DNP3 Configuration: Enabled

Transport/Network Layer Preprocessors

- Checksum Verification: Enabled
- Inline Normalization: Enabled

Annotations:

- View preprocessors
- Currently Enabled
- Enabled with non-default settings
- Enabled with default settings

NAP Settings That Can Cause Silent Drops

In the example mentioned in the trace section, the rule TCP Stream Configuration rule **129:14** is dropping traffic. This is determined by looking at the **system support trace** output. However, if the said rule is not enabled within the respective Intrusion Policy, no Intrusion Events are sent to the FMC.

The reason why this happens is due to a setting within the **Inline Normalization** preprocessor called **Block Unresolvable TCP Header Anomalies**. This option basically allows Snort to perform a block action when certain GID 129 rules detect anomalies in the TCP stream.

If **Block Unresolvable TCP Header Anomalies** is enabled, it is recommended to turn on the GID 129 rules per the illustration below.

The screenshot displays the 'Intrusion Policy' configuration page for GID: "129". It shows a list of 19 rules, with 12 selected. A context menu is open over rule 129:14, showing options: 'Generate Events', 'Drop and Generate Events', and 'Disable'. The 'Policy Information' sidebar is open, showing the 'Settings' section with 'Inline Normalization' selected. The 'Inline Normalization' settings are visible, with 'Block Unresolvable TCP Header Anomalies' checked and highlighted by a red box.

Rule ID	Rule Name	Selected
129 4	STREAM5_BAD_TIMESTAMP	✓
129 5	STREAM5_BAD_SEGMENT	☐
129 6	STREAM5_WINDOW_TOO_LARGE	✓
129 7	STREAM5_EXCESSIVE_TCP_OVERLAPS	☐
129 8	STREAM5_DATA_AFTER_RESET	✓
129 9	STREAM5_SESSION_HIJACKED_CLIENT	☐
129 10	STREAM5_SESSION_HIJACKED_SERVER	☐
129 11	STREAM5_DATA_WITHOUT_FLAGS	✓
129 12	STREAM5_SMALL_SEGMENT	☐
129 13	STREAM5_4WAY_HANDSHAKE	☐
129 14	STREAM5_NO_TIMESTAMP	✓
129 15	STREAM5_BAD_RST	✓
129 16	STREAM5_BAD_FIN	✓
129 17	STREAM5_BAD_ACK	✓
129 18	STREAM5_DATA_AFTER_RST_RCVD	✓
129 19	STREAM5_WINDOW_SLAM	✓

Network Analysis Policy

Inline Normalization

- Normalize IPv4
- Normalize Don't Fragment Bit
- Normalize Reserved Bit
- Normalize TOS Bit
- Normalize Excess Payload
- Normalize IPv6
- Normalize ICMPv4
- Normalize ICMPv6
- Normalize/Clear Reserved Bits
- Normalize/Clear Option Padding Bytes
- Clear Urgent Pointer if URG=0
- Clear Urgent Pointer/URG on Empty Payload
- Clear URG if Urgent Pointer Is Not Set
- Normalize Urgent Pointer
- Normalize TCP Payload
- Remove Data on SYN
- Remove Data on RST
- Trim Data to Window
- Trim Data to MSS
- Block Unresolvable TCP Header Anomalies**

Turning on the GID 129 rules causes Intrusion Events to be sent to the FMC when they take action on the traffic. However, as long as the **Block Unresolvable TCP Header Anomalies** is enabled, it can still drop traffic even if the **Rule State** in the Intrusion Policy is set to only **Generate Events**. This behavior is explained in the FMC Configuration Guides.

Still drops after setting to generate



Inline Result	Source IP	Destination IP	Source Port / ICMP Type	Destination Port / ICMP Code	Message
↓	172.16.111.226	50.19.123.95	51174 / tcp	443 (https) / tcp	STREAMS_NO_TIMESTAMP (129:14:2)
↓	172.16.111.226	50.19.123.95	51174 / tcp	443 (https) / tcp	STREAMS_NO_TIMESTAMP (129:14:2)

Check configuration guide for relative protocols/preprocessors:

Block Unresolvable TCP Header Anomalies

When you enable this option, the system blocks anomalous TCP packets that, if normalized, would be invalid and likely would be blocked by the receiving host. For example, the system blocks any SYN packet transmitted subsequent to an established session.

The system also drops any packet that matches any of the following TCP stream preprocessor rules, regardless of whether the rules are enabled:

- 129:1
- 129:3
- 129:4
- 129:6
- 129:8
- 129:11
- 129:14 through 129:19

The Total Blocked Packets performance graph tracks the number of packets blocked in inline deployments and, in passive deployments and inline deployments in tap mode, the number that would have been blocked in an inline deployment.

The above documentation can be found in this [article](#) (for version 6.4, which is the most recent version at the time of this article posting).

Verify the Backend Configuration

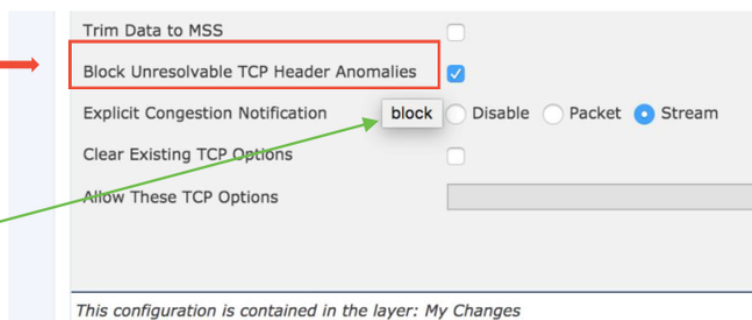
Another layer of complexity is added to the behavior of the preprocessor in that certain settings can be enabled on the backend, without being reflected in the FMC. These are some possible reasons.

- Other enabled features have the ability to force enable preprocessor settings (the main one being File Policy)
- Some Intrusion Policy rules require certain preprocessor options in order to perform detection
- A defect may cause the behavior We have seen one instance of this: [CSCuz50295](#) - "File policy with Malware block enables TCP normalization with block flag"

Before looking at the backend configuration, note that the Snort keywords, which are used in the backend Snort configuration files, can be seen by hovering over a specific setting within the NAP. Please refer to the illustration below.

Hover over option to see backend snort configuration keyword

Snort config keyword is "block"



The **Block Unresolvable TCP Header Anomalies** option in the NAP tab translates to the **block** keyword on the backend. With that information in mind, the backend configuration can be checked

from the expert shell.

```
root@ciscoasa:~# de_info.pl
-----
DE Name      : Primary Detection Engine (c9ef19d6-e187-11e6-ba76-99617d53da68)
DE Type     : ids
DE Description : Primary detection engine for device c9ef19d6-e187-11e6-ba76-99617d53da68
DE Resources  : 1
DE UUID     : 0d82120c-e188-11e6-8606-a4827d53da68
-----

root@ciscoasa:~# cd /var/sf/detection_engines/0d82120c-e188-11e6-8606-a4827d53da68/network_analysis/
root@ciscoasa: network_analysis# ls
b50f27b0-e31a-11e6-b866-dd9e65c01d56 object_b50f27b0-e31a-11e6-b866-dd9e65c01d56 snort.conf.b50f27b0-e31a-11e6-b866-
dd9e65c01d56 snort.conf.b50f27b0-e31a-11e6-b866-dd9e65c01d56.default
root@ciscoasa: network_analysis# cat b50f27b0-e31a-11e6-b866-dd9e65c01d56/normalize.conf
#
# generated from My Changes
#
preprocessor normalize_tcp: ips, rsv, pad, req_urg, req_pay, req_urp, block
```

“block” option is enabled in normalize.conf

Creating a Targeted NAP

If certain hosts are triggering preprocessor events, a custom NAP can be used to inspect traffic to or from said hosts. Within the custom NAP, the settings which are causing issues can be disabled.

These are the steps for implementing a targeted NAP.

1. Create the NAP per the instructions mentioned in Verify NAP configuration section of this article.
2. In the **Advanced** tab of the Access Control Policy, navigate to the **Network Analysis and Intrusion Policies** section. Click **Add Rule** and create a rule, using the targeted hosts and choose the newly created NAP in the **Network Analysis Policy** section.

Click to expand NA Rules

Add rule(s) to target traffic with certain NAP

False Positive Analysis

Checking for false positives in Intrusion Events for preprocessor rules is quite different than that of

the Snort rules used for rule evaluation (which contain a GID of 1 and 3).

In order to perform a false positive analysis for preprocessor rule events, a full session capture is necessary to look for anomalies within the TCP stream.

In the example below, false positive analysis is being performed on rule **129:14**, which is shown to be dropping traffic in the examples above. Since **129:14** is looking for TCP streams in which timestamps are missing, you can clearly see why the rule was triggered per the packet capture analysis illustrated below.

Full session pcap

Internet Protocol Version 4, Src: 172.16.111.226, Dst: 50.19.123.95
Transmission Control Protocol, Src Port: 51174, Dst Port: 443, Seq: 3849839666, Len: 0
Source Port: 51174
Destination Port: 443
[Stream index: 2]
[TCP Segment Len: 0]
Sequence number: 3849839666
Acknowledgment number: 0
Header Length: 40 bytes
Flags: 0x002 (SYN)
Window size value: 8192
[Calculated window size: 8192]
Checksum: 0x70ba [correct]
[Checksum Status: Good]
[Calculated Checksum: 0x70ba]
Urgent pointer: 0
Options: 20 bytes, Maximum segment size, No-Operation (NOP), Window scale, SACK permitted, Timestamps
Maximum segment size: 1380 bytes
No-Operation (NOP)
Window scale: 8 (multiply by 256)
TCP SACK Permitted Option: True
Timestamps: Tsvail 2054852, TSecr 0

SYN packet has TCP Timestamps

Packet that triggered event

Internet Protocol Version 4, Src: 172.16.111.226, Dst: 50.19.123.95
Transmission Control Protocol, Src Port: 51174, Dst Port: 443, Seq: 3849839667, Ack: 1666843207, Len: 0
Source Port: 51174
Destination Port: 443
[Stream index: 0]
[TCP Segment Len: 0]
Sequence number: 3849839667
Acknowledgment number: 1666843207
Header Length: 20 bytes
Flags: 0x010 (ACK)
Window size value: 57
[Calculated window size: 57]
[Window size scaling factor: -1 (unknown)]
Checksum: 0xed47 [correct]
[Checksum Status: Good]
[Calculated Checksum: 0xed47]
Urgent pointer: 0

No TCP Timestamps in event packet (violates RFC)

Mitigation Steps

To quickly mitigate possible issues with the NAP, the following steps can be performed.

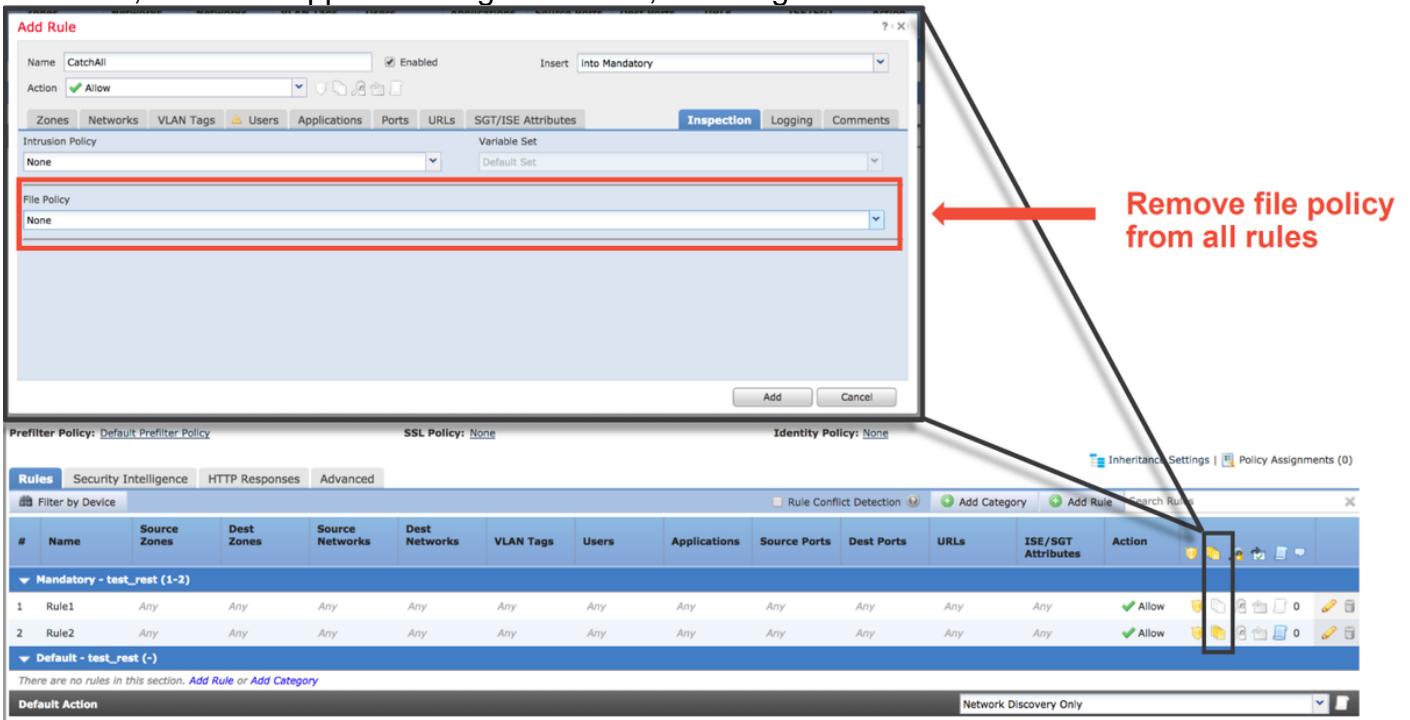
- If a custom NAP is being used and you aren't sure if a NAP setting is dropping traffic but you suspect it might be, you can try replacing it with a "Balanced Security and Connectivity" or "Connectivity over Security" policy.

The screenshot shows the 'Advanced' settings for Security Intelligence. The 'Network Analysis and Intrusion Policies' section is highlighted, showing a dialog box for configuration. The 'Default Network Analysis Policy' is set to 'Balanced Security and Connectivity'. Other settings include 'Intrusion Policy used before Access Control rule is determined' (No Rules Active), 'Intrusion Policy Variable Set' (Default-Set), and 'Network Analysis Rules' (No Custom Rules). The 'Default Network Analysis Policy' dropdown is highlighted with a red box.

- If any "Custom Rules" are being used, make sure to set the NAP to one of the defaults

mentioned above

- If any Access Control rules use a File Policy, you may need to try temporarily removing it as a File Policy can enable pre-processor settings on the backend which are not be reflected in the FMC, and this happens at a "global" level, meaning all NAPs are modified.



Each protocol has a different preprocessor and troubleshooting them can be very specific to the preprocessor. This article does not cover all preprocessor settings and troubleshooting methods for each.

You can check the documentation for each preprocessor to get a better idea of what each option does, which is helpful when troubleshooting a specific preprocessor.

Data to Provide to TAC

Data

Instructions

Troubleshoot

File from the
Firepower

<http://www.cisco.com/c/en/us/support/docs/security/sourcefire-defense-center/117663-tech>

Device

Full Session

Packet

Capture from
the

<http://www.cisco.com/c/en/us/support/docs/security/sourcefire-firepower-8000-series-applia>

Firepower

device