

IOS-XE Datapath信息包踪影功能

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[参考拓扑](#)

[在使用中的包跟踪](#)

[快速入门指南](#)

[Enable \(event\)平台有条件调试](#)

[Enable \(event\)信息包踪影](#)

[出口与信息包踪影的情况限制](#)

[显示信息包踪影结果](#)

[FIA跟踪](#)

[显示信息包踪影结果](#)

[检查与接口产生关联的FIA](#)

[转存被跟踪的信息包](#)

[丢弃跟踪](#)

[示例丢弃跟踪方案](#)

[注入并且踢跟踪](#)

[信息包跟踪示例](#)

[信息包跟踪示例- NAT](#)

[信息包跟踪示例- VPN](#)

[性能影响](#)

[Related Information](#)

Introduction

本文描述如何通过信息包踪影功能执行datapath Cisco IOS XE软件的信息包踪影。

为了识别问题例如误配置，容量超载，甚至普通的软件Bug，当排除故障时，了解是必要的什么发生在系统内的一个信息包。Cisco IOS XE信息包踪影功能针对此需要。它提供使用认为和为了捕获根据用户定义的情况组的每个信息包进程详细资料的字段SAFE方法。

Prerequisites

Requirements

Cisco建议您有是可用的在Cisco IOS XE版本3.10和以上信息包踪影功能的知识，以及在运行Cisco IOS XE软件，例如Cisco 1000 Series Aggregation Services Routers的所有平台(ASR1K)，Cisco 1000V Series Cloud Services Router (CSR1000v)和Cisco 4451-X Series Integrated Services Router (ISR4451-X)。

Components Used

本文档中的信息基于以下软件和硬件版本：

- Cisco IOS XE软件版本3.10S (15.3(3)S)和以后
- ASR1K

本文的信息从设备在特定实验室环境里被创建了。All of the devices used in this document started with a cleared (default) configuration.如果您的网络实际，请保证您了解使用的所有命令的潜在影响。

参考拓扑

此图表说明使用示例在本文描述的拓扑：



在使用中的包跟踪

为了说明使用信息包跟踪功能，使用在此部分中的示例描述互联网控制消息协议(ICMP)数据流的跟踪从本地工作站172.16.10.2的(在ASR1K后)到远端主机172.16.20.2 (ASR1K的入口方向在Gig0/0/1接口)。

您能跟踪在ASR1K的信息包与这两个步骤：

1. Enable (event)您在ASR1K要跟踪的平台有条件调试为了选择信息包或数据流。
2. Enable (event)平台信息包跟踪(PATH跟踪或功能调用阵列(FIA)跟踪)。

快速入门指南

这是快速入门指南，如果已经熟悉本文内容，并且想要快速查找的一个部分在CLI。这些是说明使用的仅一些个示例工具。请参见详细讨论语法的后面的章节，并且保证您使用是适当的对您的需求的配置。

1. 配置平台情况：

```
debug platform condition ipv4 10.0.0.1/32 both --> matches in and out packets with source or destination as 10.0.0.1/32
```

```
debug platform condition ipv4 access-list 198 egress --> (Ensure access-list 198 is defined prior to configuring this command) - matches egress packets corresponding to access-list 198
```

```
debug platform condition interface gig 0/0/0 ingress --> matches all ingress packets on interface gig 0/0/0
```

debug platform condition mpls 10 1 ingress --> matches MPLS packets with top ingress label 10

debug platform condition ingress --> matches all ingress packets on all interfaces (use cautiously)

在配置后平台情况，请开始平台条件用此CLI命令：

debug platform condition start

2. 配置信息包踪影：

debug platform packet-trace packet 1024 -> basic path-trace, and automatically stops tracing packets after 1024 packets. You can use "circular" option if needed **debug platform packet-trace packet 1024 fia-trace** -> enables detailed fia trace, stops tracing packets after 1024 packets **debug platform packet-trace drop [code <dropcode>]** -> if you want to trace/capture only packets that are dropped. Refer to Drop Trace section for more details.

Note:在初期的Cisco IOS XE 3.x版本中，也要求debug命令平台信息包跟踪enable (event)开始信息包跟踪功能。这在Cisco IOS XE 16.x版本不再需要。

输入此命令为了清楚跟踪缓冲区和重置信息包跟踪：

clear platform packet-trace statistics --> clear the packet trace buffer

命令清除平台调节，并且信息包踪影配置是：

clear platform condition all --> clears both platform conditions and the packet trace configuration

[显示命令](#)

验证平台情况和信息包踪影配置，在您适用早先命令为了保证后您有什么您需要。

show platform conditions --> shows the platform conditions configured

show platform packet-trace configuration --> shows the packet-trace configurations

show debugging --> this will show both platform conditions and platform packet-trace configured
这是命令检查被跟踪的/获取信息包：

show platform packet-trace statistics --> statistics of packets traced

show platform packet-trace summary --> summary of all the packets traced, with input and output interfaces, processing result and reason. **show platform packet-trace packet 12** -> Tracing the 12th packet, with complete path trace or FIA trace details.

Enable (event)平台有条件调试

信息包踪影功能依靠有条件调试基础设施为了确定将被跟踪的信息包。有条件调试基础设施提供能力给基于的过滤流量：

- 协议
- IP地址和掩码
- 访问控制表(ACL)
- 接口
- 流量方向(入口或出口)

这些情况定义了过滤器何时何地应用于信息包。

在本例中使用的数据流，enable (event)在入口方向的平台有条件调试自172.16.10.2的ICMP信息包的到172.16.20.2。换句话说，请选择您要跟踪的数据流。有您能使用为了选择此数据流的多种选项

。

```
ASR1000#debug platform condition ?
egress Egress only debug
feature For a specific feature
ingress Ingress only debug
interface Set interface for conditional debug
ipv4 Debug IPv4 conditions
ipv6 Debug IPv6 conditions
start Start conditional debug
stop Stop conditional debug
```

在本例中，访问列表用于为了定义情况，如显示这里：

```
ASR1000#show access-list 150
Extended IP access list 150
10 permit icmp host 172.16.10.2 host 172.16.20.2
ASR1000#debug platform condition interface gig 0/0/1 ipv4
access-list 150 ingress
```

为了开始条件调试，请输入此命令：

```
ASR1000#debug platform condition start
```

Note:为了终止或禁用条件调试基础设施，请输入stop命令调试平台的情况。

为了查看配置的有条件调试过滤器，输入此命令：

```
ASR1000#show platform conditions

Conditional Debug Global State: Start
Conditions Direction
-----|-----
GigabitEthernet0/0/1 & IPV4 ACL [150] ingress

Feature Condition Format Value
-----|-----|-----
```

```
ASR1000#
```

总之，至今运用了此配置：

```
access-list 150 permit icmp host 172.16.10.2 host 172.16.20.2
debug platform condition interface gig 0/0/1 ipv4 access-list 150 ingress
debug platform condition start
```

Enable (event)信息包踪影

Note:此部分详细描述信息包和复制选项，并且其它选项在本文以后描述。

物理和逻辑接口支持信息包踪影，例如隧道或虚拟访问接口。

这是信息包踪影CLI语法：

```
ASR1000#debug platform packet-trace ?
copy Copy packet data
drop Trace drops only
inject Trace injects only
packet Packet count
punt Trace punts only
```

```
debug platform packet-trace packet <pkt-size/pkt-num> [fia-trace | summary-only]
[circular] [data-size <data-size>]
```

这是此命令关键字的说明：

- **Pkt数字**-数据包编号指定一次被维护的最大信息包的数量。
- **summary-only** -这指定仅概略的数据是获取的。默认值是获取概略的数据和功能PATH数据。
- **FIA跟踪**-除路径数据信息之外，这可选地执行FIA跟踪。
- **数据大小**-这允许您指定路径数据缓冲区的大小，从2,048个到16,384个字节。默认值是2,048个字节。

```
debug platform packet-trace copy packet {in | out | both} [L2 | L3 | L4]
[size <num-bytes>]
```

这是此命令关键字的说明：

- **in/out**这指定将被复制的信息包流的方向-入口和出口。
- **L2/L3/L4** -这允许您指定信息包的复制开始的位置。第2层(L2)是默认位置。
- **大小**-这允许您指定被复制八位位组的最大数量。默认值是64个八位位组。

对于此示例，这是命令选择与有条件调试基础设施的数据流的使用的为了enable (event)信息包踪影：

```
ASR1000#debug platform packet-trace packet 16
```

为了查看信息包踪影配置，请输入此命令：

```
ASR1000#show platform packet-trace configuration
debug platform packet-trace packet 16 data-size 2048
```

您能也输入show debugging命令为了查看平台有条件调试和信息包踪影配置：

```
ASR1000# show debugging
IOSXE Conditional Debug Configs:
```

```
Conditional Debug Global State: Start
```

```
Conditions
Direction
```

```
-----|-----
GigabitEthernet0/0/1 & IPV4 ACL [150] ingress
```

```
...
IOSXE Packet Tracing Configs:
```

```
Feature Condition Format Value
-----|-----|-----
```

```
Feature Type Submode Level
-----|-----|-----
```

```
IOSXE Packet Tracing Configs:
```

```
debug platform packet-trace packet 16 data-size 2048
```

Note:输入**all**命令清楚的平台的情况为了清除所有平台调试条件和信息包踪影配置和数据。

总之，此配置数据是使用的至今为了enable (event)信息包跟踪：

```
debug platform packet-trace packet 16
```

出口与信息包踪影的情况限制

条件定义了有条件的过滤器，并且，当他们适用于信息包。例如，调试平台情况接口g0/0/0出口意味着信息包被识别作为匹配，当到达在接口g0/0/0时的输出FIA，那么从入口发生的所有信息包处理，直到该点丢失。

Note:Cisco强烈建议您使用入口情况信息包踪影为了得到可能最完全和的有意义的的数据。可以使用出口条件，但是知道限制。

显示信息包踪影结果

Note:此部分假设，PATH跟踪是启用的。

信息包踪影提供检查的三个特定级别：

- 认为
- 每个信息包汇总
- 每个信息包路径数据

当五个ICMP请求信息包从172.16.10.2被发送到172.16.20.2时，这些命令可以用于为了查看信息包踪影结果：

```
ASR1000#show platform packet-trace statistics
Packets Traced: 5
Ingress 5
Inject 0
```

```
Forward 5
Punt 0
Drop 0
Consume 0
```

```
ASR1000#show platform packet-trace summary
```

Pkt	Input	Output	State	Reason
0	Gi0/0/1	Gi0/0/0	FWD	
1	Gi0/0/1	Gi0/0/0	FWD	
2	Gi0/0/1	Gi0/0/0	FWD	
3	Gi0/0/1	Gi0/0/0	FWD	
4	Gi0/0/1	Gi0/0/0	FWD	

```
ASR1000#show platform packet-trace packet 0
```

```
Packet: 0          CBUG ID: 4
```

```
Summary
```

```
Input : GigabitEthernet0/0/1
```

```
Output : GigabitEthernet0/0/0
```

```
State : FWD
```

```
Timestamp
```

```
Start   : 1819281992118 ns (05/17/2014 06:42:01.207240 UTC)
```

```
Stop    : 1819282095121 ns (05/17/2014 06:42:01.207343 UTC)
```

```
Path Trace
```

```
Feature: IPV4
```

```
Source : 172.16.10.2
```

```
Destination : 172.16.20.2
```

```
Protocol : 1 (ICMP)
```

```
ASR1000#
```

Note:第三个命令提供说明如何查看每个信息包的信息包踪影的一个示例。在本例中，被跟踪的第一个信息包显示。

从这些输出，您能看到五个信息包被跟踪，并且您能查看输入接口、输出接口、状态和路径跟踪。

状态 重新标明

FWD 信息包为发运被安排/排队，转发到下一跳通过输出接口。

PUNT 信息包从转发处理器(FP)被踢到路由处理器(RP) (控制层面)。

丢弃 信息包在FP被丢弃。运行FIA跟踪，请使用全局丢弃计数器，或者使用datapath调试为了查找丢弃原因。

CONS 信息包在一个信息包进程中被消耗，例如在ICMP Ping请求期间或crypto信息包。

入口和注入在输出对应于信息包通过一个外部接口和信息包分别输入被看到如被注入从控制层面的信息包踪影统计数据的计数器。

FIA跟踪

FIA拿着由信息包处理器引擎功能的列表(PPE)顺序地执行在Quantum流处理器(QFP)，当转发信息包入口或出口时。功能根据在机器被运用的配置数据。因此，当信息包被处理，FIA跟踪帮助了解信息包的流通过系统。

您必须应用与FIA的此配置数据为了enable (event)信息包踪影：

```
ASR1000#debug platform packet-trace packet 16 fia-trace
```

显示信息包踪影结果

Note:此部分假设，FIA跟踪是启用的。并且，当您添加或修改当前信息包跟踪命令时，清除缓冲信息包跟踪详细资料，因此您必须再发送若干数据流，以便您能跟踪它。

从172.16.10.2请发送五个ICMP信息包到172.16.20.2，在您输入是被使用的为了enable (event) FIA跟踪后的命令，正如前面的部分所描述。

```
ASR1000#show platform packet-trace summary
```

Pkt	Input	Output	State	Reason
0	Gi0/0/1	Gi0/0/0	FWD	
1	Gi0/0/1	Gi0/0/0	FWD	
2	Gi0/0/1	Gi0/0/0	FWD	
3	Gi0/0/1	Gi0/0/0	FWD	
4	Gi0/0/1	Gi0/0/0	FWD	

```
ASR1000#show platform packet-trace packet 0
```

```
Packet: 0          CBUG ID: 9
```

```
Summary
```

```
Input      : GigabitEthernet0/0/1
Output     : GigabitEthernet0/0/0
State      : FWD
```

```
Timestamp
```

```
Start      : 1819281992118 ns (05/17/2014 06:42:01.207240 UTC)
Stop       : 1819282095121 ns (05/17/2014 06:42:01.207343 UTC)
```

```
Path Trace
```

```
Feature: IPV4
```

```
Source      : 172.16.10.2
Destination : 172.16.20.2
Protocol    : 1 (ICMP)
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x8059dbe8 - DEBUG_COND_INPUT_PKT
Timestamp   : 3685243309297
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x82011a00 - IPV4_INPUT_DST_LOOKUP_CONSUME
Timestamp   : 3685243311450
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
Timestamp   : 3685243312427
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x82004b68 - IPV4_OUTPUT_LOOKUP_PROCESS
Timestamp   : 3685243313230
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x8034f210 - IPV4_INPUT_IPOPTIONS_PROCESS
Timestamp   : 3685243315033
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x82013200 - IPV4_OUTPUT_GOTO_OUTPUT_FEATURE
Timestamp   : 3685243315787
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x80321450 - IPV4_VFR_REFRAG
Timestamp   : 3685243316980
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x82014700 - IPV6_INPUT_L2_REWRITE
Timestamp   : 3685243317713
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x82000080 - IPV4_OUTPUT_FRAG
Timestamp   : 3685243319223
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x8200e500 - IPV4_OUTPUT_DROP_POLICY
Timestamp   : 3685243319950
```

```
Feature: FIA_TRACE
```

```
Entry       : 0x8059aff4 - PACTRAC_OUTPUT_STATS
Timestamp   : 3685243323603
```



```
Feature: FIA_TRACE
Entry      : 0x82016100 - MARMOT_SPA_D_TRANSMIT_PKT
Timestamp  : 3685243326183
```

ASR1000#

检查与接口产生关联的FIA

当您enable (event)平台有条件调试，它被添加到FIA作为功能。根据位置被添加到列表，您也许需要调整您的平台条件，例如，当您跟踪PRE encap和POST encap信息包时。

此输出在入口方向被启用的平台条件调试的FIA显示功能的定货：

```
ASR1000#show platform hardware qfp active interface if-name GigabitEthernet 0/0/1
```

General interface information

Interface Name: GigabitEthernet0/0/1

Interface state: VALID

Platform interface handle: 10

QFP interface handle: 8

Rx uidb: 1021

Tx uidb: 131064

Channel: 16

Interface Relationships

BGPPA/QPPB interface configuration information

Ingress: BGPPA/QPPB not configured. flags: 0000

Egress : BGPPA not configured. flags: 0000

ipv4_input enabled.

ipv4_output enabled.

layer2_input enabled.

layer2_output enabled.

ess_ac_input enabled.

Features Bound to Interface:

2 GIC FIA state

48 PUNT INJECT DB

39 SPA/Marmot server

40 ethernet

1 IFM

31 icmp_svr

33 ipfrag_svr

34 ipreass_svr

36 ipvfr_svr

37 ipv6vfr_svr

12 CPP IPSEC

Protocol 0 - ipv4_input

FIA handle - CP:0x108d99cc DP:0x8070f400

IPV4_INPUT_DST_LOOKUP_ISSUE (M)

IPV4_INPUT_ARL_SANITY (M)

CBUG_INPUT_FIA

DEBUG_COND_INPUT_PKT

IPV4_INPUT_DST_LOOKUP_CONSUME (M)

IPV4_INPUT_FOR_US_MARTIAN (M)

IPV4_INPUT_IPSEC_CLASSIFY

IPV4_INPUT_IPSEC_COPROC_PROCESS

IPV4_INPUT_IPSEC_RERUN_JUMP

IPV4_INPUT_LOOKUP_PROCESS (M)

IPV4_INPUT_IPOPTIONS_PROCESS (M)

IPV4_INPUT_GOTO_OUTPUT_FEATURE (M)

```
Protocol 1 - ipv4_output
FIA handle - CP:0x108d9a34 DP:0x8070eb00
IPV4_OUTPUT_VFR
MC_OUTPUT_GEN_RECYCLE (D)
IPV4_VFR_REFRAG (M)
IPV4_OUTPUT_IPSEC_CLASSIFY
IPV4_OUTPUT_IPSEC_COPROC_PROCESS
IPV4_OUTPUT_IPSEC_RERUN_JUMP
IPV4_OUTPUT_L2_REWRITE (M)
IPV4_OUTPUT_FRAG (M)
IPV4_OUTPUT_DROP_POLICY (M)
PACTRAC_OUTPUT_STATS
MARMOT_SPA_D_TRANSMIT_PKT
DEF_IF_DROP_FIA (M)
Protocol 8 - layer2_input
FIA handle - CP:0x108d9bd4 DP:0x8070c700
LAYER2_INPUT_SIA (M)
CBUG_INPUT_FIA
DEBUG_COND_INPUT_PKT
LAYER2_INPUT_LOOKUP_PROCESS (M)
LAYER2_INPUT_GOTO_OUTPUT_FEATURE (M)
Protocol 9 - layer2_output
FIA handle - CP:0x108d9658 DP:0x80714080
LAYER2_OUTPUT_SERVICEWIRE (M)
LAYER2_OUTPUT_DROP_POLICY (M)
PACTRAC_OUTPUT_STATS
MARMOT_SPA_D_TRANSMIT_PKT
DEF_IF_DROP_FIA (M)
Protocol 14 - ess_ac_input
FIA handle - CP:0x108d9ba0 DP:0x8070cb80
PPPOE_GET_SESSION
ESS_ENTER_SWITCHING
PPPOE_HANDLE_UNCLASSIFIED_SESSION
DEF_IF_DROP_FIA (M)
```

```
QfpEth Physical Information
DPS Addr: 0x11215eb8
Submap Table Addr: 0x00000000
VLAN Ethertype: 0x8100
QOS Mode: Per Link
```

ASR1000#

Note:CBUG_INPUT_FIA和DEBUG_COND_INPUT_PKT对应于在路由器被配置的有条件调试功能。

转存被跟踪的信息包

您能复制和转存信息包，当他们被跟踪，当此部分描述。此示例显示如何复制最多2,048字节的在入口方向(172.16.10.2的信息包到172.16.20.2)。

这是需要的其它命令：

```
ASR1000#debug platform packet-trace copy packet input size 2048
```

Note:被复制信息包的大小是在16个到2,048个字节范围内。

输入此命令为了转存被复制的信息包：

```
ASR1000#show platform packet-trace packet 0
Packet: 0 CBUG ID: 14
Summary
Input : GigabitEthernet0/0/1
Output : GigabitEthernet0/0/0
State : FWD
Timestamp
  Start   : 1819281992118 ns (05/17/2014 06:40:01.207240 UTC)
  Stop    : 1819282095121 ns (05/17/2014 06:40:01.207343 UTC)
Path Trace
Feature: IPV4
Source : 172.16.10.2
Destination : 172.16.20.2
Protocol : 1 (ICMP)
Feature: FIA_TRACE
Entry : 0x8059dbe8 - DEBUG_COND_INPUT_PKT
Timestamp : 4458180580929
```

<some content excluded>

```
Feature: FIA_TRACE
Entry : 0x82016100 - MARMOT_SPA_D_TRANSMIT_PKT
Timestamp : 4458180593896
```

Packet Copy In

```
a4934c8e 33020023 33231379 08004500 00640160 0000ff01 5f16ac10 0201ac10
01010800 1fd40024 00000000 000184d0 d980abcd abcdabcd abcdabcd abcdabcd
abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd
abcdabcd abcdabcd abcdabcd abcdabcd abcd
```

ASR1000#

丢弃跟踪

丢弃跟踪是可用的在Cisco IOS XE软件版本3.11及以后。它enable (event)仅信息包踪影丢弃的数据包的。这是功能的一些高亮度显示：

- 它可选地允许您为一个特定丢弃代码指定信息包的挽留。
- 它可以用于，不用全局或接口情况为了捕获丢弃事件。
- 丢弃事件捕获意味着仅丢弃被跟踪，信息包的不是生活。然而，它仍然允许您获取概略的数据，元组数据，并且信息包为了帮助精炼情况或提供提示向下调试跨步。

这是被使用的为了enable (event)丢弃型的信息包踪影的命令句法：

```
debug platform packet-trace drop [code <code-num>]
```

丢弃代码是相同的象丢弃ID，如在显示平台硬件qfp的报告活动统计信息丢弃detail命令输出：

```
ASR1000#show platform hardware qfp active statistics drop detail
```

```
-----
  ID  Global Drop Stats                               Packets                               Octets
-----
  60  IpTtlExceeded                                   3                                       126
  8   Ipv4Acl                                         32                                      3432
```

示例丢弃跟踪方案

适用在ASR1K的Gig 0/0/0接口的此ACL为了从172.16.10.2降低数据流到172.16.20.2 :

```
ASR1000#show platform hardware qfp active statistics drop detail
```

ID	Global Drop Stats	Packets	Octets
60	IpTtlExceeded	3	126
8	Ipv4Acl	32	3432

使用到位ACL，从本地主机降低数据流到远端主机，请运用此丢弃跟踪配置：

```
debug platform condition interface Gig 0/0/1 ingress
debug platform condition start
debug platform packet-trace packet 1024 fia-trace
debug platform packet-trace drop
```

从172.16.10.2发送五个ICMP请求信息包到172.16.20.2。丢弃跟踪获取由ACL丢弃的这些信息包，如显示：

```
ASR1000#show platform packet-trace statistics
```

```
Packets Summary
Matched 5
Traced 5
Packets Received
Ingress 5
Inject 0
Packets Processed
Forward 0
Punt 0
Drop      5
Count Code Cause
5 8 Ipv4Acl
Consume 0
```

```
ASR1000#show platform packet-trace summary
```

```
Pkt Input Output State Reason
0 Gi0/0/1 Gi0/0/0 DROP 8 (Ipv4Acl)
1 Gi0/0/1 Gi0/0/0 DROP 8 (Ipv4Acl)
2 Gi0/0/1 Gi0/0/0 DROP 8 (Ipv4Acl)
3 Gi0/0/1 Gi0/0/0 DROP 8 (Ipv4Acl)
4 Gi0/0/1 Gi0/0/0 DROP 8 (Ipv4Acl)
```

```
ASR1K#debug platform condition stop
```

```
ASR1K#show platform packet-trace packet 0
```

```
Packet: 0 CBUG ID: 140
Summary
Input : GigabitEthernet0/0/1
Output : GigabitEthernet0/0/0
State   : DROP 8 (Ipv4Acl)
Timestamp
Start : 1819281992118 ns (05/17/2014 06:42:01.207240 UTC)
Stop  : 1819282095121 ns (05/17/2014 06:42:01.207343 UTC)
Path Trace
Feature: IPV4
Source : 172.16.10.2
Destination : 172.16.20.2
Protocol : 1 (ICMP)
```

```

Feature: FIA_TRACE
Entry : 0x806c7eac - DEBUG_COND_INPUT_PKT
Lapsed time: 1031 ns
Feature: FIA_TRACE
Entry : 0x82011c00 - IPV4_INPUT_DST_LOOKUP_CONSUME
Lapsed time: 657 ns
Feature: FIA_TRACE
Entry : 0x806a2698 - IPV4_INPUT_ACL
Lapsed time: 2773 ns
Feature: FIA_TRACE
Entry : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
Lapsed time: 1013 ns
Feature: FIA_TRACE
Entry : 0x82004500 - IPV4_OUTPUT_LOOKUP_PROCESS
Lapsed time: 2951 ns
Feature: FIA_TRACE
Entry : 0x8041771c - IPV4_INPUT_IPOPTIONS_PROCESS
Lapsed time: 373 ns
Feature: FIA_TRACE
Entry : 0x82013400 - MPLS_INPUT_GOTO_OUTPUT_FEATURE
Lapsed time: 2097 ns
Feature: FIA_TRACE
Entry : 0x803c60b8 - IPV4_MC_OUTPUT_VFR_REFRAG
Lapsed time: 373 ns
Feature: FIA_TRACE
Entry : 0x806db148 - OUTPUT_DROP
Lapsed time: 1297 ns
Feature: FIA_TRACE
Entry : 0x806a0c98 - IPV4_OUTPUT_ACL
Lapsed time: 78382 ns

```

ASR1000#

注入并且踢跟踪

注入和平底船信息包踪影功能在Cisco IOS XE软件版本在FP被接受被踢到控制层面)的3.12被添加及以后为了跟踪平底船(信息包和注入(被注入对从控制层面的FP)的信息包信息包。

Note:平底船跟踪能工作，不用全局或建立接口条件，正如丢弃跟踪。然而，必须定义条件为了注入跟踪能工作。

在这里平底船的示例并且注入信息包踪影，当您从ASR1K连接到邻接路由器时：

```

ASR1000#debug platform condition ipv4 172.16.10.2/32 both
ASR1000#debug platform condition start
ASR1000#debug platform packet-trace punt
ASR1000#debug platform packet-trace inject
ASR1000#debug platform packet-trace packet 16
ASR1000#
ASR1000#ping 172.16.10.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.2, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 14/14/15 ms
ASR1000#

```

现在您能验证平底船和注入跟踪结果：

```
ASR1000#show platform packet-trace summary
```

```
Pkt Input Output State Reason
0 INJ.2 Gi0/0/1 FWD
1 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
2 INJ.2 Gi0/0/1 FWD
3 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
4 INJ.2 Gi0/0/1 FWD
5 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
6 INJ.2 Gi0/0/1 FWD
7 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
8 INJ.2 Gi0/0/1 FWD
9 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
```

```
ASR1000#show platform packet-trace packet 0
```

```
Packet: 0 CBUG ID: 120
```

```
Summary
```

```
Input      : INJ.2
```

```
Output : GigabitEthernet0/0/1
```

```
State : FWD
```

```
Timestamp
```

```
Start : 115612780360228 ns (05/29/2014 15:02:55.467987 UTC)
```

```
Stop : 115612780380931 ns (05/29/2014 15:02:55.468008 UTC)
```

```
Path Trace
```

```
Feature: IPV4
```

```
Source : 172.16.10.1
```

```
Destination : 172.16.10.2
```

```
Protocol : 1 (ICMP)
```

```
ASR1000#
```

```
ASR1000#show platform packet-trace packet 1
```

```
Packet: 1 CBUG ID: 121
```

```
Summary
```

```
Input : GigabitEthernet0/0/1
```

```
Output : internal0/0/rp:0
```

```
State      : PUNT 11 (For-us data)
```

```
Timestamp
```

```
Start : 115612781060418 ns (05/29/2014 15:02:55.468687 UTC)
```

```
Stop : 115612781120041 ns (05/29/2014 15:02:55.468747 UTC)
```

```
Path Trace
```

```
Feature: IPV4
```

```
Source : 172.16.10.2
```

```
Destination : 172.16.10.1
```

```
Protocol : 1 (ICMP)
```

信息包跟踪示例

此部分提供信息包跟踪功能是有用的为了实现故障排除目的一些示例。

信息包跟踪示例- NAT

使用此示例，接口源网络地址转换(NAT)在ASR1K (Gig0/0/0)的广域网接口被配置本地子网的(172.16.10.0/24)。

这是使用为了跟踪从172.16.10.2的数据流到172.16.20.2，变得转换的平台情况和信息包跟踪配置(NAT)在Gig0/0/0接口：

```
ASR1000#show platform packet-trace summary
```

```
Pkt Input Output State Reason
```

```
0 INJ.2 Gi0/0/1 FWD
```

```
1 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
```

```
2 INJ.2 Gi0/0/1 FWD
3 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
4 INJ.2 Gi0/0/1 FWD
5 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
6 INJ.2 Gi0/0/1 FWD
7 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
8 INJ.2 Gi0/0/1 FWD
9 Gi0/0/1 internal0/0/rp:0 PUNT 11 (For-us data)
```

```
ASR1000#show platform packet-trace packet 0
```

```
Packet: 0 CBUG ID: 120
```

```
Summary
```

```
Input      : INJ.2
```

```
Output : GigabitEthernet0/0/1
```

```
State : FWD
```

```
Timestamp
```

```
Start : 115612780360228 ns (05/29/2014 15:02:55.467987 UTC)
```

```
Stop : 115612780380931 ns (05/29/2014 15:02:55.468008 UTC)
```

```
Path Trace
```

```
Feature: IPV4
```

```
Source : 172.16.10.1
```

```
Destination : 172.16.10.2
```

```
Protocol : 1 (ICMP)
```

```
ASR1000#
```

```
ASR1000#show platform packet-trace packet 1
```

```
Packet: 1 CBUG ID: 121
```

```
Summary
```

```
Input : GigabitEthernet0/0/1
```

```
Output : internal0/0/rp:0
```

```
State      : PUNT 11 (For-us data)
```

```
Timestamp
```

```
Start : 115612781060418 ns (05/29/2014 15:02:55.468687 UTC)
```

```
Stop : 115612781120041 ns (05/29/2014 15:02:55.468747 UTC)
```

```
Path Trace
```

```
Feature: IPV4
```

```
Source : 172.16.10.2
```

```
Destination : 172.16.10.1
```

```
Protocol : 1 (ICMP)
```

当五个ICMP信息包从172.16.10.2被发送到与接口来源NAT配置时的172.16.20.2，这些是信息包踪影结果：

```
ASR1000#show platform packet-trace summary
```

```
Pkt Input Output State Reason
```

```
0 Gi0/0/1 Gi0/0/0 FWD
```

```
1 Gi0/0/1 Gi0/0/0 FWD
```

```
2 Gi0/0/1 Gi0/0/0 FWD
```

```
3 Gi0/0/1 Gi0/0/0 FWD
```

```
4 Gi0/0/1 Gi0/0/0 FWD
```

```
ASR1000#show platform packet-trace statistics
```

```
Packets Summary
```

```
Matched 5
```

```
Traced 5
```

```
Packets Received
```

```
Ingress 5
```

```
Inject 0
```

```
Packets Processed
```

```
Forward 5
```

```
Punt 0
```

```
Drop 0
```

```
Consume 0
```

ASR1000#show platform packet-trace packet 0

Packet: 0 CBUG ID: 146

Summary

Input : GigabitEthernet0/0/1

Output : GigabitEthernet0/0/0

State : FWD

Timestamp

Start : 3010217805313 ns (05/17/2014 07:01:52.227836 UTC)

Stop : 3010217892847 ns (05/17/2014 07:01:52.227923 UTC)

Path Trace

Feature: IPV4

Source : 172.16.10.2

Destination : 172.16.20.2

Protocol : 1 (ICMP)

Feature: FIA_TRACE

Entry : 0x806c7eac - DEBUG_COND_INPUT_PKT

Lapsed time: 1031 ns

Feature: FIA_TRACE

Entry : 0x82011c00 - IPV4_INPUT_DST_LOOKUP_CONSUME

Lapsed time: 462 ns

Feature: FIA_TRACE

Entry : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN

Lapsed time: 355 ns

Feature: FIA_TRACE

Entry : 0x803c6af4 - IPV4_INPUT_VFR

Lapsed time: 266 ns

Feature: FIA_TRACE

Entry : 0x82004500 - IPV4_OUTPUT_LOOKUP_PROCESS

Lapsed time: 942 ns

Feature: FIA_TRACE

Entry : 0x8041771c - IPV4_INPUT_IPOPTIONS_PROCESS

Lapsed time: 88 ns

Feature: FIA_TRACE

Entry : 0x82013400 - MPLS_INPUT_GOTO_OUTPUT_FEATURE

Lapsed time: 568 ns

Feature: FIA_TRACE

Entry : 0x803c6900 - IPV4_OUTPUT_VFR

Lapsed time: 266 ns

Feature: NAT

Direction : IN to OUT

Action : Translate Source

Old Address : 172.16.10.2 00028

New Address : 192.168.10.1 00002

Feature: FIA_TRACE

Entry : 0x8031c248 - IPV4_NAT_OUTPUT_FIA

Lapsed time: 55697 ns

Feature: FIA_TRACE

Entry : 0x801424f8 - IPV4_OUTPUT_THREAT_DEFENSE

Lapsed time: 693 ns

Feature: FIA_TRACE

Entry : 0x803c60b8 - IPV4_MC_OUTPUT_VFR_REFRAG

Lapsed time: 88 ns

Feature: FIA_TRACE

Entry : 0x82014900 - IPV6_INPUT_L2_REWRITE

Lapsed time: 444 ns

Feature: FIA_TRACE

Entry : 0x82000080 - IPV4_OUTPUT_FRAG

Lapsed time: 88 ns

Feature: FIA_TRACE

Entry : 0x8200e600 - IPV4_OUTPUT_DROP_POLICY

Lapsed time: 1457 ns

Feature: FIA_TRACE

Entry : 0x82017980 - MARMOT_SPA_D_TRANSMIT_PKT

Lapsed time: 7431 ns
ASR1000#

信息包跟踪示例- VPN

使用此示例， Site to Site VPN隧道用于在ASR1K和Cisco IOS路由器之间为了保护流在172.16.10.0/24和172.16.20.0/24之间的数据流(本地和远程子网)。

这是使用为了跟踪VPN流量从172.16.10.2流到在Gig 0/0/1接口的172.16.20.2的平台情况和信息包跟踪配置：

```
ASR1000#show platform packet-trace summary
```

```
Pkt Input Output State Reason
0 Gi0/0/1 Gi0/0/0 FWD
1 Gi0/0/1 Gi0/0/0 FWD
2 Gi0/0/1 Gi0/0/0 FWD
3 Gi0/0/1 Gi0/0/0 FWD
4 Gi0/0/1 Gi0/0/0 FWD
```

```
ASR1000#show platform packet-trace statistics
```

```
Packets Summary
Matched 5
Traced 5
Packets Received
Ingress 5
Inject 0
Packets Processed
Forward 5
Punt 0
Drop 0
Consume 0
```

```
ASR1000#show platform packet-trace packet 0
```

```
Packet: 0 CBUG ID: 146
Summary
Input : GigabitEthernet0/0/1
Output : GigabitEthernet0/0/0
State : FWD
Timestamp
Start : 3010217805313 ns (05/17/2014 07:01:52.227836 UTC)
Stop : 3010217892847 ns (05/17/2014 07:01:52.227923 UTC)
Path Trace
Feature: IPV4
Source : 172.16.10.2
Destination : 172.16.20.2
Protocol : 1 (ICMP)
Feature: FIA_TRACE
Entry : 0x806c7eac - DEBUG_COND_INPUT_PKT
Lapsed time: 1031 ns
Feature: FIA_TRACE
Entry : 0x82011c00 - IPV4_INPUT_DST_LOOKUP_CONSUME
Lapsed time: 462 ns
Feature: FIA_TRACE
Entry : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
Lapsed time: 355 ns
Feature: FIA_TRACE
Entry : 0x803c6af4 - IPV4_INPUT_VFR
Lapsed time: 266 ns
Feature: FIA_TRACE
Entry : 0x82004500 - IPV4_OUTPUT_LOOKUP_PROCESS
Lapsed time: 942 ns
```

```

Feature: FIA_TRACE
Entry : 0x8041771c - IPV4_INPUT_IPOPTIONS_PROCESS
Lapsed time: 88 ns
Feature: FIA_TRACE
Entry : 0x82013400 - MPLS_INPUT_GOTO_OUTPUT_FEATURE
Lapsed time: 568 ns
Feature: FIA_TRACE
Entry : 0x803c6900 - IPV4_OUTPUT_VFR
Lapsed time: 266 ns
Feature: NAT
Direction : IN to OUT
Action : Translate Source
Old Address : 172.16.10.2 00028
New Address : 192.168.10.1 00002
Feature: FIA_TRACE
Entry : 0x8031c248 - IPV4_NAT_OUTPUT_FIA
Lapsed time: 55697 ns
Feature: FIA_TRACE
Entry : 0x801424f8 - IPV4_OUTPUT_THREAT_DEFENSE
Lapsed time: 693 ns
Feature: FIA_TRACE
Entry : 0x803c60b8 - IPV4_MC_OUTPUT_VFR_REFRAG
Lapsed time: 88 ns
Feature: FIA_TRACE
Entry : 0x82014900 - IPV6_INPUT_L2_REWRITE
Lapsed time: 444 ns
Feature: FIA_TRACE
Entry : 0x82000080 - IPV4_OUTPUT_FRAG
Lapsed time: 88 ns
Feature: FIA_TRACE
Entry : 0x8200e600 - IPV4_OUTPUT_DROP_POLICY
Lapsed time: 1457 ns
Feature: FIA_TRACE
Entry : 0x82017980 - MARMOT_SPA_D_TRANSMIT_PKT
Lapsed time: 7431 ns
ASR1000#

```

当五个ICMP信息包从172.16.10.2被发送到172.16.20.2时，由在ASR1K和Cisco IOS路由器之间的VPN隧道加密在本例中，这些是信息包trace输出：

Note:信息包踪影在使用为了加密信息包，是有用的跟踪显示QFP安全关联(SA)把柄，当您排除IPSec VPN问题故障为了验证时正确的SA使用加密。

```

ASR1000#show platform packet-trace summary
Pkt Input Output State Reason
0 Gi0/0/1 Gi0/0/0 FWD
1 Gi0/0/1 Gi0/0/0 FWD
2 Gi0/0/1 Gi0/0/0 FWD
3 Gi0/0/1 Gi0/0/0 FWD
4 Gi0/0/1 Gi0/0/0 FWD

ASR1000#show platform packet-trace packet 0
Packet: 0 CBUG ID: 211
Summary
Input : GigabitEthernet0/0/1
Output : GigabitEthernet0/0/0
State : FWD
Timestamp
Start : 4636921551459 ns (05/17/2014 07:28:59.211375 UTC)
Stop : 4636921668739 ns (05/17/2014 07:28:59.211493 UTC)
Path Trace

```

Feature: IPV4
Source : 172.16.10.2
Destination : 172.16.20.2
Protocol : 1 (ICMP)
Feature: FIA_TRACE
Entry : 0x806c7eac - DEBUG_COND_INPUT_PKT
Lapsed time: 622 ns
Feature: FIA_TRACE
Entry : 0x82011c00 - IPV4_INPUT_DST_LOOKUP_CONSUME
Lapsed time: 462 ns
Feature: FIA_TRACE
Entry : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
Lapsed time: 320 ns
Feature: FIA_TRACE
Entry : 0x82004500 - IPV4_OUTPUT_LOOKUP_PROCESS
Lapsed time: 1102 ns
Feature: FIA_TRACE
Entry : 0x8041771c - IPV4_INPUT_IPOPTIONS_PROCESS
Lapsed time: 88 ns
Feature: FIA_TRACE
Entry : 0x82013400 - MPLS_INPUT_GOTO_OUTPUT_FEATURE
Lapsed time: 586 ns
Feature: FIA_TRACE
Entry : 0x803c6900 - IPV4_OUTPUT_VFR
Lapsed time: 266 ns
Feature: FIA_TRACE
Entry : 0x80757914 - MC_OUTPUT_GEN_RECYCLE
Lapsed time: 195 ns
Feature: FIA_TRACE
Entry : 0x803c60b8 - IPV4_MC_OUTPUT_VFR_REFRAG
Lapsed time: 88 ns

Feature: IPSec
Result : IPSEC_RESULT_SA
Action : ENCRYPT
SA Handle : 6
Peer Addr : 192.168.20.1
Local Addr: 192.168.10.1

Feature: FIA_TRACE
Entry : 0x8043caec - IPV4_OUTPUT_IPSEC_CLASSIFY
Lapsed time: 9528 ns
Feature: FIA_TRACE
Entry : 0x8043915c - IPV4_OUTPUT_IPSEC_DOUBLE_ACL
Lapsed time: 355 ns
Feature: FIA_TRACE
Entry : 0x8043b45c - IPV4_IPSEC_FEATURE_RETURN
Lapsed time: 657 ns
Feature: FIA_TRACE
Entry : 0x8043ae28 - IPV4_OUTPUT_IPSEC_RERUN_JUMP
Lapsed time: 888 ns
Feature: FIA_TRACE
Entry : 0x80436f10 - IPV4_OUTPUT_IPSEC_POST_PROCESS
Lapsed time: 2186 ns
Feature: FIA_TRACE
Entry : 0x8043b45c - IPV4_IPSEC_FEATURE_RETURN
Lapsed time: 675 ns
Feature: FIA_TRACE
Entry : 0x82014900 - IPV6_INPUT_L2_REWRITE
Lapsed time: 1902 ns
Feature: FIA_TRACE
Entry : 0x82000080 - IPV4_OUTPUT_FRAG
Lapsed time: 71 ns
Feature: FIA_TRACE
Entry : 0x8200e600 - IPV4_OUTPUT_DROP_POLICY
Lapsed time: 1582 ns

Feature: FIA_TRACE
Entry : 0x82017980 - MARMOT_SPA_D_TRANSMIT_PKT
Lapsed time: 3964 ns
ASR1000#

性能影响

信息包跟踪缓冲区消耗QFP DRAM，如此是配置要求是可用的记住内存数量和内存数量。

性能影响变化，从属在是启用的信息包踪影选项。信息包踪影只影响被跟踪信息包的转发性能，例如匹配用户配置的条件的那些信息包。更加粒状和详细信息您配置信息包踪影捕获，越非常地将影响资源。

当调试情况担保它时，和与其中任一排除故障，采取仅一迭代方法和enable (event)更多详细的跟踪选项是最佳的。

QFP DRAM使用方法可以估计与此公式：

所需的内存= (顶上的stats) + 数字pkts * (概略的大小+路径数据大小+复制大小)

Note:那里stats开销和汇总大小是固定的在2 KB，并且128 B，路径数据大小和复制大小分别为用户可配置的。

Related Information

- [Cisco ASR1000 Series Aggregation Series Routers软件配置指南-信息包踪影](#)
- [在Cisco ASR1000 Series Service Routers的信息包丢弃](#)
- [Technical Support & Documentation - Cisco Systems](#)