

IOS-XE 用の ZBFW 設定のトラブルシューティング ガイド

目次

[概要](#)

[リンクおよびドキュメント](#)

[コマンド リファレンス](#)

[データパストラブルシューティング手順](#)

[設定の確認](#)

[接続状態の確認](#)

[ファイアウォール ドロップ カウンタのチェック](#)

[QFP 上のグローバルドロップ カウンタ](#)

[QFP 上のファイアウォール機能ドロップ カウンタ](#)

[ファイアウォール ドロップのトラブルシューティング](#)

[ロギング](#)

[ローカル バッファ syslog](#)

[ローカル バッファ syslog の制限](#)

[リモート高速ロギング](#)

[条件一致を使用したパケット トレース](#)

[Embedded Packet Capture](#)

[デバッグ](#)

[条件付きデバッグ](#)

[デバッグの収集と表示](#)

概要

このドキュメントでは、アグリゲーション サービス ルータ (ASR) 上のハードウェア ドロップ カウンタをポーリングするコマンドを使用して、ASR 1000 のゾーンベース ファイアウォール (ZBFW) 機能をトラブルシューティングするベスト プラクティスについて説明します。ASR 1000 はハードウェアベースの転送プラットフォームです。Cisco IOS-XE® のソフトウェア設定によって、ハードウェア ASIC Quantum Flow Processor (QFP) が機能転送機能を実行するようにプログラムできます。これにより、スループットが増加し、パフォーマンスが向上します。欠点は、トラブルシューティングがやりにくいことです。これまでゾーンベース ファイアウォール (ZBFW) 経由で現在のセッションとドロップ カウンタをポーリングするために使用されていた Cisco IOS コマンドは、ソフトウェアでドロップが行われなくなったため、無効になっています。

。

リンクおよびドキュメント

コマンド リファレンス

- [Cisco ASR 1000 シリーズ アグリゲーション サービス ルータ コマンド リファレンス](#)
- [Cisco IOS XE 3S コマンド リファレンス](#)

データパス トラブルシューティング手順

データパスをトラブルシューティングするには、トラフィックが ASR と Cisco IOS-XE のコードを正しくパス スルーするかどうかを確認する必要があります。ファイアウォール機能に特有のデータパスのトラブルシューティングは次の手順に従います。

1. **設定の確認** : 設定を収集して、出力を調査し、接続を確認します。
2. **接続状態の確認** : トラフィックが正しく通過する場合は、Cisco IOS-XE が ZBFW 機能の接続を有効にします。この接続は、クライアントとサーバ間のトラフィックと状態情報を追跡します。
3. **ドロップ カウンタの確認** : トラフィックが正しく通過しない場合は、Cisco IOS-XE がドロップされたパケットに関するドロップ カウンタを記録します。この出力をチェックして、トラフィック障害の原因を特定します。
4. **ロギング** : syslog を収集して、接続の構築やパケットのドロップに関するより詳細な情報を調査します。
5. **パケットトレース ドロップ パケット** : パケットトレースを使用して、ドロップされたパケットを収集します。
6. **デバッグ** : デバッグの収集は最も冗長なオプションです。デバッグは、パケットの正確な転送パスを確認するために、条件付きで入手できます。

設定の確認

show tech support firewall の出力を以下に示します。

```
----- show clock -----  
----- show version -----  
----- show running-config -----  
----- show parameter-map type inspect -----  
----- show policy-map type inspect -----  
----- show class-map type inspect -----  
----- show zone security -----  
----- show zone-pair security -----  
----- show policy-firewall stats global -----  
----- show policy-firewall stats zone -----  
----- show platform hardware qfp active feature firewall datapath <submode> -----  
----- show platform software firewall RP <submode> -----
```

接続状態の確認

接続情報を入手すれば、ZBFW 上のすべての接続を網羅することができます。コマンド

```
ASR#show policy-firewall sessions platform
```

```
--show platform hardware qfp active feature firewall datapath scb any any any any all any --  
[s=session i=imprecise channel c=control channel d=data channel]  
14.38.112.250 41392 14.36.1.206 23 proto 6 (0:0) [sc]
```

これは、14.38.112.250 と 14.36.1.206 間の TCP Telnet 接続を表示します。

注: このコマンドを実行すると、デバイス上の接続数が多い場合に時間がかかることに注意してください。このコマンドは、次のように、特定のフィルタを指定して実行することをお勧めします。

接続テーブルは、特定の送信元アドレスまたは宛先アドレスに絞り込むことができます。

platform サブモードの後でフィルタを使用します。フィルタリングのオプションを以下に示します。

```
radar-ZBFW1#show policy-firewall sessions platform ?
```

```
all detailed information  
destination-port Destination Port Number  
detail detail on or off  
icmp Protocol Type ICMP  
imprecise imprecise information  
session session information  
source-port Source Port  
source-vrf Source Vrf ID  
standby standby information  
tcp Protocol Type TCP  
udp Protocol Type UDP  
v4-destination-address IPv4 Desination Address  
v4-source-address IPv4 Source Address  
v6-destination-address IPv6 Desination Address  
v6-source-address IPv6 Source Address  
| Output modifiers  
<cr>
```

この接続テーブルは、14.38.112.250 からの接続だけが表示されるようにフィルタリングされています。

```
ASR#show policy-firewall sessions platform v4-source-address 14.38.112.250
```

```
--show platform hardware qfp active feature firewall datapath scb 14.38.112.250  
any any any any all any --  
[s=session i=imprecise channel c=control channel d=data channel]
```

```
14.38.112.250 41392 14.36.1.206 23 proto 6 (0:0) [sc]
```

接続テーブルをフィルタリングすれば、詳細な接続情報を入手してより包括的な分析を行うことができます。この出力を表示するには、**detail** キーワードを使用します。

```
ASR#show policy-firewall sessions platform v4-source-address 14.38.112.250 detail
```

```
--show platform hardware qfp active feature firewall datapath scb 14.38.112.250  
any any any all detail--
```

```
[s=session i=imprecise channel c=control channel d=data channel]
```

```
14.38.112.250 41426 14.36.1.206 23 proto 6 (0:0) [sc]
```

```
pscb : 0x8c5d4f20, bucket : 64672, fw_flags: 0x204 0x20419441,  
scb state: active, scb debug: 0
```

```
nxt_timeout: 360000, refcnt: 1, ha nak cnt: 0, rg: 0, sess id: 117753
```

```
hostdb: 0x0, L7: 0x0, stats: 0x8e118e40, child: 0x0
```

```
l4blk0: 78fae7a7 l4blk1: e36df99c l4blk2: 78fae7ea l4blk3: 39080000
```

```
l4blk4: e36df90e l4blk5: 78fae7ea l4blk6: e36df99c l4blk7: fde0000
```

```
l4blk8: 0 l4blk9: 1
```

```
root scb: 0x0 act_blk: 0x8e1115e0
```

```
ingress/egress intf: GigabitEthernet0/0/2 (1021), GigabitEthernet0/0/0 (131065)
current time 34004163065573 create tstamp: 33985412599209 last access: 33998256774622
nat_out_local_addr:port: 0.0.0.0:0 nat_in_global_addr:port: 0.0.0.0:0
syncookie fixup: 0x0
halfopen linkage: 0x0 0x0
cxsc_cft_fid: 0x0
tw timer: 0x0 0x0 0x372ba 0x1e89c181
Number of simultaneous packet per session allowed: 25
  bucket 125084 flags 1 func 1 idx 8 wheel 0x8ceb1120
```

ファイアウォール ドロップ カウンタのチェック

XE 3.9 でドロップ カウンタ出力が変更されました。XE 3.9 以前は、ファイアウォール ドロップの理由は非常に一般的なものでした。XE 3.9 以降は、ファイアウォール ドロップの理由が拡張され、より詳しくなりました。

ドロップ カウンタを確認するには、次の 2 つのステップを実行します。

1. Cisco IOS-XE でグローバル ドロップ カウンタを確認します。このカウンタは、どの機能がトラフィックをドロップしたかを示します。機能の例として、Quality of Service (QoS)、ネットワーク アドレス変換 (NAT)、ファイアウォールなどが挙げられます。
2. サブ機能が特定された場合は、サブ機能によって提供される詳細なドロップ カウンタを照会します。このガイドでは、分析対象のサブ機能がファイアウォール機能です。

QFP 上のグローバル ドロップ カウンタ

基になる基本コマンドは QFP 上のすべてのドロップを提供します。

```
Router#show platform hardware qfp active statistics drop
```

このコマンドは QFP 上のドロップをまとめて表示します。このドロップには任意の機能を含めることができます。機能の例を以下に示します。

```
Router#show platform hardware qfp active statistics drop
```

すべてのドロップを表示するには、値が 0 のカウンタを追加して、次のコマンドを使用します。

```
show platform hardware qfp active statistics drop all
```

カウンタをクリアするには、次のコマンドを使用します。これは、画面に表示された出力をクリアします。このコマンドは読み取り時クリアされるため、出力は画面に表示された後で 0 にリセットされます。

```
show platform hardware qfp active statistics drop all
```

QFP グローバル ファイアウォール ドロップ カウンタと説明のリストを以下に示します。

ファイアウォール グローバル ドロップの理由	説明
------------------------	----

FirewallBackpressure	ロギング メカニズムによるバックプレッシャが原因のパケット ドロップ。
FirewallInvalidZone	インターフェイス用に設定されたセキュリティ ゾーンが存在しません。
FirewallL4Insp	L4 ポリシー チェック エラー。より詳細なドロップの理由については、次の (プの理由) を参照してください。
FirewallNoForwardingZone	ファイアウォールが初期化されておらず、トラフィックが通過を許可されてセッションの作成が失敗します。これは、最大セッション制限に到達したか
FirewallNonsession	

FirewallPolicy	可能性があります。 設定されたファイアウォールポリシーがドロップになっています。
FirewallL4	L4 検査エラー。より詳細なドロップの理由については、次の表 (ファイアウォール機能ドロップの理由) を参照してください。
FirewallL7	L7 検査が原因の packets ドロップ。より詳細な L7 ドロップの理由のリスト (ファイアウォール機能ドロップの理由) を参照してください。
FirewallNotInitiator	TCP、UDP、または ICMP のセッション イニシエータが存在しません。セッションが、ICMP の場合は、受信された最初の packet が ECHO でも TIMESTAMP ではなく、SYN ではありません。
FirewallNoNewSession	この現象は、通常の packet 処理または不正なチャネル処理で発生する可能性があります。
FirewallSyncookieMaxDst	ファイアウォール ハイ アベイラビリティで新しいセッションが許可されません。ホストベースの SYN フラッド保護を提供するために、SYN フラッド制限と SYN Cookie が有効になります。
FirewallSyncookie	宛先エントリの数が上限に達すると、新しい SYN packet がドロップされ、SYN COOLIE ロジックがトリガーされます。これは、SYN Cookie を含む SYN packet がドロップされたことを示します。
FirewallARStandby	非対称ルーティングが有効になっておらず、冗長グループがアクティブ状態

QFP 上のファイアウォール機能ドロップ カウンタ

QFP グローバル ドロップ カウンタに伴う制限はドロップの理由が細分化されないことであり、**FirewallL4** などの一部のドロップの理由によって特定の箇所に過負荷が生じ、ほとんどトラブルシューティングに使用できません。これは、ファイアウォール機能ドロップ カウンタが追加された Cisco IOS-XE 3.9 (15.3(2)S) 以降で改善されています。これにより、ドロップのかなり詳細な理由のセットが提供されます。

```
ASR#show platform hardware qfp active feature firewall drop all
```

```
-----  
Drop Reason Packets  
-----
```

```
Invalid L4 header 0  
Invalid ACK flag 0  
Invalid ACK number 0  
....
```

ファイアウォール機能ドロップの理由と説明のリストを以下に示します。

ファイアウォール機能ドロップの理由 説明

Invalid Header length	データグラムが非常に小さいため、レイヤ 4TCP、UDP、または ICMP のドロップされる理由は次のとおりです。 1. TCP ヘッダー長 < 20 2. UDP/ICMP ヘッダー長 < 8
Invalid UDP data length	UDP データグラム長が UDP ヘッダーで指定された長さとは一致しません。このドロップは、次の理由のいずれかで発生する可能性があります。 1. ACK が TCP ピアの next_seq# と一致しない。 2. ACK が TCP ピアから送信された最新の SEQ# を超えている
Invalid ACK Number	TCP SYNRCVD 状態と SYNSENT 状態では、ACK# が ISN+1 になる必要があります。このドロップは、次の理由のいずれかで発生する可能性があります。 1. ACK フラグが想定されているが、別の TCP 状態でセットされている。 2. ACK フラグ以外の他のフラグ (RST など) もセットされている。
Invalid ACK Flag	この状態が発生するのは、以下の場合です。 1. TCP イニシエータからの最初の packet が SYN ではない (RST など) せずに受信される)。 2. 初期 SYN packet で ACK フラグがセットされている。
Invalid TCP Initiator	

SYN with data	SYN パケットにペイロードが含まれています。そのような使用は無効な TCP フラグの原因を以下に示します。
Invalid TCP Flags	<ol style="list-style-type: none"> 1. TCP 初期 SYN パケットに SYN 以外のフラグが含まれている。 2. TCP リッスン状態で、TCP ピアが RST または ACK を受信する。 3. 他のレスポンドの packets が SYN/ACK の前に受信される。 4. 想定された SYN/ACK がレスポンドから受信されない。
Invalid Segment in SYNSENT state	SYNSENT 状態の無効な TCP セグメントの原因を以下に示します。 <ol style="list-style-type: none"> 1. SYN/ACK にペイロードが含まれている。 2. SYN/ACK に他のフラグ (PSH、URG、FIN) がセットされている。 3. ペイロードを含む通過 SYN を受信する。 4. イニシエータから非 SYN パケットを受信する。
Invalid Segment in SYNRCVD state	SYNRCVD 状態の無効な TCP セグメントの原因を以下に示します。 <ol style="list-style-type: none"> 1. イニシエータからペイロードを含む再通過 SYN を受信する。 2. レスポンドから SYN/ACK、RST、または FIN 以外の無効な packets が受信される。
Invalid SEQ	これは、セグメントがイニシエータから届いたときに SYNRCVD 状態にあることを示します。 <ol style="list-style-type: none"> 1. Seq# が ISN 未満である。 2. レシーバの rcvd ウィンドウ サイズが 0 で、セグメントにペイロードまたは不正なセグメント (seq# がレシーバ LASTACK を超えている) が含まれている。 3. レシーバの rcvd ウィンドウ サイズが 0 で、seq# がウィンドウの範囲外である。 4. Seq# が ISN と一致するが、SYN パケットと一致しない。
Invalid Window Scale Option	無効な TCP ウィンドウ スケール オプションは、誤ったウィンドウ サイズを指定する可能性があります。あるウィンドウが相手側の ACK より遅く受信される可能性があります。
TCP out of window	無効な TCP ウィンドウ スケール オプションは、誤ったウィンドウ サイズを指定する可能性があります。あるウィンドウが相手側の ACK より遅く受信される可能性があります。
TCP extra payload after FIN sent	ペイロードが FIN の送信後に受信されました。これは、CLOSE_WAIT 状態に遷移する可能性があります。
TCP Window Overflow	これは、入力セグメント サイズがレシーバのウィンドウを超えている場合は、この状態が許可されます。これは、ファイアウォールがパケットをドロップする必要があるためです。
Retran with Invalid Flags	再通過パケットがすでにレシーバによって確認応答されています。不正なパケットが検査用に L7 に配信されようとしています。L7 パケットがドロップされます。
TCP out-of-order Segment	TCP SYN フラッド攻撃を受けています。特定の条件下で、このカウンタ値を超えると、ファイアウォールはこの IP アドレスへの新しい接続がドロップされます。
SYN フラッド	synflood チェック中に、hostdb の割り当てに失敗します。推奨処置 : "show platform hardware qfp active feature firewall memory" を実行してください。
Internal Err - synflood check alloc Failed	設定されたハーフオープン接続数を超過し、ブラックアウト時間中にすべての新しい接続がドロップされます。
Synflood blackout drop	許容ハーフオープン セッション数を超過したためにパケットがドロップされます。また、"max-incomplete high/low" と "one-minute high/low" の設定値を確認してください。これらの設定の影響を受けないことを確認します。
Half-Open Session Limit Exceed	フローごとに許可される検査可能パケットの最大数を超過しています。
Too Many Pkt per flow	フローごとに許可される ICMP エラー パケットの最大数を超過しています。
Too many ICMP error packets per flow	SYNRCVD 状態では、TCP がレスポンドからイニシエータの方向に送信されるべきです。
Unexpect TCP payload from Rsp to Init	パケット方向が定義されていません。
Internal Error - Undefined Direction	SYN パケットが、すでに確立された TCP 接続のウィンドウ内で受信されています。
SYN inside current window	RST パケットが、すでに確立された TCP 接続のウィンドウ内で受信されています。
RST inside current window	

Stray Segment	レスポンスからリッスン状態で受信される TCP SYN パケットなどの TCP セグメントが受信されます。
ICMP Internal Error - Missed ICMP NAT info	ICMP パケットが NAT されていますが、内部 NAT 情報が見つかりません。
ICMP packet in SCB close state	SCB CLOSE 状態で ICMP パケットが受信されました。
Missed IP header in ICMP packet	ICMP パケット内に IP ヘッダーがありません。
ICMP Error No IP or ICMP	ペイロード内に IP または ICMP がない ICMP エラー パケット。この可能性もあります。
ICMP Err Pkt Too Short	ICMP エラー パケットが短すぎます。
ICMP Err Exceed Burst Limit	ICMP エラー パケットが 10 のバースト制限を超えています。
ICMP Err Unreachable	到達不能な ICMP エラー パケットが制限を超えています。最初の ICMP エラー 組み込みパケットの seq# が ICMP エラーを引き起こすパケットの seq# と一致していません。
ICMP Err Invalid Seq#	ICMP エラー 組み込みパケット内の無効な ACK。
ICMP Err Invalid Ack	設定された ICMP アクションがドロップになっています。
ICMP action drop	ゾーン ペアに対するポリシーが存在しません。アプリケーションレイヤの ALG (アプリケーション レイヤ ゲートウェイ) が設定されていない場合、拡張性の問題でピンホールが開かなかったことが原因の可能性もあります。
Zone-pair without policy-map	セッション ルックアップが失敗して、このパケットを検査するポリシーが存在しません。
Session Missed And Policy Not Present	ゾーン ペアに対するポリシーが設定されていない ICMP エラー。
ICMP Error And Policy Not Present	プロトコルが検査可能かどうかをファイアウォールが判断しようとした。
Classification Failed	分類アクションがドロップになっています。
Classification Action Drop	セキュリティ ポリシーの誤設定によって分類が失敗しました。この場合も発生する可能性があります。
Security Policy Misconfig	ACK# が ISN+1 と一致しないときに SYNSENT 状態でレスポンスが送信されず、ポリシー アクションがドロップになっています。
Send RST to responder	最初のフラグメントがドロップされると、残りのフラグメントもドロップされます。
Firewall Policy Drop	ICMP 組み込みパケットのポリシー アクションがドロップになります。
Fragment Drop	L7 (ALG) がパケットのドロップを決定します。理由は、さまざまな ALG が受け付けないセグメント化されたパケットが受信されたり、ALG が受け付けないフラグメント化されたパケット (または VFP) が認識されないプロトコル タイプです。
ICMP Firwall Policy Drop	
L7 inspection returns DROP	
L7 Segment Pkt Not Allow	
L7 Fragment Pkt Not Allow	
Unknown L7 Proto Type	

ファイアウォール ドロップのトラブルシューティング

前述したグローバルまたはファイアウォール機能ドロップ カウンタからドロップの理由が特定されたが、それらのドロップが想定されていなかった場合は、追加のトラブルシューティング手順が必要になります。有効になっているファイアウォール機能に対して設定が正しいことを保証するための設定検証は別として、大抵の場合は、疑わしいトラフィック フローのパケット キャプチャを取って、パケットが不正かどうか、プロトコルまたはアプリケーションの実装上の問題が発生していないかどうかを確認する必要があります。

ロギング

ASR ロギング機能は、ドロップされたパケットを記録する syslog を生成します。この syslog は、パケットがドロップされた理由の詳細を提供します。次の 2 種類の syslog があります。

1. ローカル バッファ syslog

2. リモート高速ロギング

ローカル バッファ syslog

ドロップの原因を特定するには、ログ ドロップの有効化などの一般的な ZBFW トラブルシューティングを使用できます。パケット ドロップ ロギングを設定するには、次の 2 つの方法があります。

方法 1: すべてのドロップされたパケットを記録するためにインスペクト グローバル パラメータ マップを使用します。

```
ASR#show platform hardware qfp active feature firewall drop all
```

```
-----  
Drop Reason Packets  
-----
```

```
Invalid L4 header 0  
Invalid ACK flag 0  
Invalid ACK number 0  
....
```

方法 2: 特定のクラスのみドロップされたパケットを記録するためにカスタム インスペクト パラメータ マップを使用します。

```
parameter-map type inspect LOG_PARAM  
log dropped-packets  
!  
policy-map type inspect ZBFW_PMAP  
class type inspect ZBFW_CMAP  
inspect LOG_PARAM
```

ASR がロギング用にどのように設定されているかに応じて、次のメッセージがログまたはコンソールに送信されます。ドロップ ログ メッセージの例を以下に示します。

```
parameter-map type inspect LOG_PARAM  
log dropped-packets  
!  
policy-map type inspect ZBFW_PMAP  
class type inspect ZBFW_CMAP  
inspect LOG_PARAM
```

ローカル バッファ syslog の制限

1. このログは、Cisco Bug ID [CSCud09943](#) によってレート制限されています。
2. このログは、特定の設定が適用されていない限り、出力されません。たとえば、log キーワードが指定されなかった場合は、class-default パケットによってドロップされたパケットが記録されません。

```
policy-map type inspect ZBFW_PMAP  
class class-default  
drop log
```

リモート高速ロギング

高速ロギング (HSL) は、QFP から直接 syslog を生成し、それを設定された NetFlow HSL コレクタに送信します。これは、ASR 上の ZBFW に対して推奨されているロギング ソリューション

です。

HSL では、次の設定を使用します。

```
policy-map type inspect ZBFW_PMAP
class class-default
drop log
```

この設定を使用するには、NetFlow バージョン 9 の NetFlow コレクタ機能が必要です。これについては、以下を参照してください。

[『設定ガイド：ゾーンベース ポリシー ファイアウォール Cisco IOS XE Release 3S \(ASR 1000 \) ファイアウォール高速ロギング』](#)

条件一致を使用したパケット トレース

パケット トレースを有効にしてから、次の機能のパケット トレースを有効にするには、条件付きデバッグをオンにします。

```
policy-map type inspect ZBFW_PMAP
class class-default
drop log
```

注: ACL が必要ないため、照合条件で直接 IP アドレスを使用できます。これは、双方向トレースを許可する送信元または宛先として照合されます。この方法は、設定の変更が許可されていない場合に使用できます。次に、例を示します。 `debug platform condition ipv4 address 192.168.1.1/32`

packet-tracing 機能をオンにします。

```
policy-map type inspect ZBFW_PMAP
class class-default
drop log
```

この機能を使用するには、次の 2 つの方法があります。

1. ドロップされたパケットだけをトレースするには、`debug platform packet-trace drop` コマンドを入力します。
2. `debug platform packet-trace drop` コマンドを除外すると、デバイスで検査/転送されるものを含め、条件と一致するパケットが追跡されます。

条件付きデバッグをオンにします。

```
policy-map type inspect ZBFW_PMAP
class class-default
drop log
```

テストを実行してから、デバッグをオフにします。

```
policy-map type inspect ZBFW_PMAP
class class-default
drop log
```

これで、情報を画面上に表示できます。この例では、ファイアウォール ポリシーが原因で ICMP パケットがドロップされています。

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

```
Packets Summary
```

```
Matched 2
```

```
Traced 2
```

```
Packets Received
```

```
Ingress 2
```

```
Inject 0
```

```
Packets Processed
```

```
Forward 0
```

```
Punt 0
```

```
Drop 2
```

```
Count      Code Cause  
2          183 FirewallPolicy
```

```
Consume 0
```

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

Pkt	Input	Output	State	Reason
0	Gi0/0/2	Gi0/0/0	DROP	183 (FirewallPolicy)
1	Gi0/0/2	Gi0/0/0	DROP	183 (FirewallPolicy)

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

```
Packet: 0          CBUG ID: 2980
```

```
Summary
```

```
Input      : GigabitEthernet0/0/2
```

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

```
State      : DROP 183 (FirewallPolicy)
```

```
Timestamp
```

```
Start      : 1207843476722162 ns (04/15/2014 12:37:01.103864 UTC)
```

```
Stop       : 1207843477247782 ns (04/15/2014 12:37:01.104390 UTC)
```

```
Path Trace
```

```
Feature: IPV4
```

```
Source      : 10.1.1.1
```

```
Destination : 192.168.1.1
```

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

```
Feature: ZBFW
```

```
Action      : Drop
```

```
Reason      : ICMP policy drop:classify result
```

```
Zone-pair name : INSIDE_OUTSIDE_ZP
```

```
Class-map name : class-default
```

```
Packet Copy In
```

```
c89c1d51 5702000c 29f9d528 08004500 00540000 40004001 ac640e26 70fa0e24  
01010800 172a2741 00016459 4d5310e4 0c000809 0a0b0c0d 0e0f1011 12131415
```

```
Packet Copy Out
```

```
c89c1d51 5702000c 29f9d528 08004500 00540000 40003f01 ad640e26 70fa0e24  
01010800 172a2741 00016459 4d5310e4 0c000809 0a0b0c0d 0e0f1011 12131415
```

show platform packet-trace packet <num> decode コマンドは、パケット ヘッダー情報とコンテンツをデコードします。この機能は XE3.11 で導入されました。

```
Router#show platform packet-trace packet all decode
```

```
Packet: 0          CBUG ID: 2980
```

```
Summary
```

```
Input      : GigabitEthernet0/0/2
```

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

```
State      : DROP 183 (FirewallPolicy)
```

```
Timestamp
```

```
Start      : 1207843476722162 ns (04/15/2014 12:37:01.103864 UTC)
```

```
Stop       : 1207843477247782 ns (04/15/2014 12:37:01.104390 UTC)
```

```
Path Trace
```

```
Feature: IPV4
```

```
Source      : 10.1.1.1
```

```
Destination : 192.168.1.1
```

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

```
Feature: ZBFW
```

Action : Drop
Reason : ICMP policy drop:classify result
Zone-pair name : INSIDE_OUTSIDE_ZP
Class-map name : class-default

Packet Copy In

c89c1d51 5702000c 29f9d528 08004500 00540000 40004001 ac640e26 70fa0e24
01010800 172a2741 00016459 4d5310e4 0c000809 0a0b0c0d 0e0f1011 12131415

ARPA

Destination MAC : c89c.1d51.5702

Source MAC : 000c.29f9.d528

Type : 0x0800 (IPV4)

IPv4

Version : 4
Header Length : 5
ToS : 0x00
Total Length : 84
Identifier : 0x0000
IP Flags : 0x2 (Don't fragment)
Frag Offset : 0
TTL : 64
Protocol : 1 (ICMP)
Header Checksum : 0xac64
Source Address : 10.1.1.1
Destination Address : 192.168.1.1

ICMP

Type : 8 (Echo)
Code : 0 (No Code)
Checksum : 0x172a
Identifier : 0x2741
Sequence : 0x0001

Packet Copy Out

c89c1d51 5702000c 29f9d528 08004500 00540000 40003f01 ad640e26 70fa0e24
01010800 172a2741 00016459 4d5310e4 0c000809 0a0b0c0d 0e0f1011 12131415

ARPA

Destination MAC : c89c.1d51.5702

Source MAC : 000c.29f9.d528

Type : 0x0800 (IPV4)

IPv4

Version : 4
Header Length : 5
ToS : 0x00
Total Length : 84
Identifier : 0x0000
IP Flags : 0x2 (Don't fragment)
Frag Offset : 0
TTL : 63
Protocol : 1 (ICMP)
Header Checksum : 0xad64
Source Address : 10.1.1.1
Destination Address : 192.168.1.1

ICMP

Type : 8 (Echo)
Code : 0 (No Code)
Checksum : 0x172a
Identifier : 0x2741
Sequence : 0x0001

Embedded Packet Capture

Embedded Packet Capture のサポートが Cisco IOS XE 3.7 (15.2(4)S) で追加されました。詳細を参照してください。

デバッグ

条件付きデバッグ

XE3.10 で、条件付きデバッグが導入されます。条件文を使用して、ZBFW 機能が条件に関するデバッグメッセージのみを記録することを保証することができます。条件付きデバッグは、ACL を使用して、ACL 要素と一致するログを制限します。また、XE3.10 以前は、デバッグメッセージを読むのが大変でした。XE3.10 では、デバッグ出力が読みやすく改良されています。

このデバッグを有効にするには、次のコマンドを発行します。

```
Router#show platform packet-trace packet all decode
Packet: 0          CBUG ID: 2980
Summary
Input      : GigabitEthernet0/0/2
Output     : GigabitEthernet0/0/0
State      : DROP 183 (FirewallPolicy)
Timestamp
  Start    : 1207843476722162 ns (04/15/2014 12:37:01.103864 UTC)
  Stop     : 1207843477247782 ns (04/15/2014 12:37:01.104390 UTC)
Path Trace
Feature: IPV4
  Source    : 10.1.1.1
  Destination : 192.168.1.1
  Protocol   : 1 (ICMP)
Feature: ZBFW
  Action    : Drop
  Reason    : ICMP policy drop:classify result
  Zone-pair name : INSIDE_OUTSIDE_ZP
  Class-map name : class-default
Packet Copy In
c89c1d51 5702000c 29f9d528 08004500 00540000 40004001 ac640e26 70fa0e24
01010800 172a2741 00016459 4d5310e4 0c000809 0a0b0c0d 0e0f1011 12131415
ARPA
  Destination MAC : c89c.1d51.5702
  Source MAC      : 000c.29f9.d528
  Type            : 0x0800 (IPV4)
IPv4
  Version         : 4
  Header Length   : 5
  ToS             : 0x00
  Total Length    : 84
  Identifier      : 0x0000
  IP Flags        : 0x2 (Don't fragment)
  Frag Offset     : 0
  TTL             : 64
  Protocol        : 1 (ICMP)
  Header Checksum : 0xac64
  Source Address   : 10.1.1.1
  Destination Address : 192.168.1.1
ICMP
  Type           : 8 (Echo)
  Code          : 0 (No Code)
  Checksum      : 0x172a
  Identifier     : 0x2741
```

```
Sequence          : 0x0001
Packet Copy Out
c89c1d51 5702000c 29f9d528 08004500 00540000 40003f01 ad640e26 70fa0e24
01010800 172a2741 00016459 4d5310e4 0c000809 0a0b0c0d 0e0f1011 12131415
ARPA
Destination MAC   : c89c.1d51.5702
Source MAC        : 000c.29f9.d528
Type              : 0x0800 (IPV4)
IPv4
Version           : 4
Header Length     : 5
ToS               : 0x00
Total Length      : 84
Identifier        : 0x0000
IP Flags          : 0x2 (Don't fragment)
Frag Offset       : 0
TTL               : 63
Protocol          : 1 (ICMP)
Header Checksum   : 0xad64
Source Address    : 10.1.1.1
Destination Address : 192.168.1.1
ICMP
Type              : 8 (Echo)
Code              : 0 (No Code)
Checksum          : 0x172a
Identifier        : 0x2741
Sequence         : 0x0001
```

条件コマンドは ACL と方向性を指定して設定する必要があることに注意してください。条件付きデバッグは、**debug platform condition start** コマンドを使用して開始されるまで実装されません。条件付きデバッグをオフにするには、**debug platform condition stop** コマンドを使用します。

debug platform condition stop

条件付きデバッグをオフにするために、**undebug all** コマンドを使用しないでください。すべての条件付きデバッグをオフにするには、次のコマンドを使用します。

ASR#clear platform condition all

XE3.14 以前は、**ha** と **event** デバッグが条件付きではありませんでした。そのため、以下で選択される条件に関係なく、**debug platform condition feature fw dataplane submode all** コマンドによってすべてのログが作成されます。これにより、デバッグを困難にする不要な情報が追加される場合があります。

デフォルトで、条件付きロギングレベルは **info** です。ロギングレベルを上げ下げするには、次のコマンドを使用します。

```
debug platform condition feature fw dataplane submode all [verbose | warning]
```

デバッグの収集と表示

デバッグ ファイルはコンソールまたはモニタに出力されません。すべてのデバッグが ASR のハードディスクに書き込まれます。デバッグは、ハードディスクのフォルダ **tracelogs** の下に、**cpp_cp_F0-0.log.<date>** という名前書き込まれます。デバッグが書き込まれたファイルを確認するには、次の出力を使用します。

```
debug platform condition feature fw dataplane submode all [verbose | warning]
```

各デバッグ ファイルは **cpp_cp_F0-0.log.<date>** ファイルとして保存されます。これらは、TFTP を使用して ASR をコピー可能な通常のテキスト ファイルです。ASR 上のログ ファイルの上限は 1 MB です。1 MB を超えると、デバッグが新しいログ ファイルに書き込まれます。このため

に、各ログ ファイルにはファイルの開始を示すタイムスタンプが付けられます。

ログ ファイルは、次の場所に保存することができます。

```
debug platform condition feature fw dataplane submode all [verbose | warning]
```

ログ ファイルはローテート後にしか表示されないため、次のコマンドを使用して手動でローテートすることができます。

```
ASR# test platform software trace slot f0 cpp-control-process rotate
```

これにより、QFP 上で "cpp_cp" ログ ファイルが即座に作成され、新しいファイルが開始されます。次に、例を示します。

```
ASR#test platform software trace slot f0 cpp-control-process rotate
```

```
Rotated file from: /tmp/fp/trace/stage/cpp_cp_F0-0.log.7311.20140408134406,  
Bytes: 82407, Messages: 431
```

```
ASR#more tracelogs/cpp_cp_F0-0.log.7311.20140408134406
```

```
04/02 10:22:54.462 : btrace continued for process ID 7311 with 159 modules  
04/07 16:52:41.164 [cpp-dp-fw]: (info): QFP:0.0 Thread:110 TS:00000531990811543397  
:FW_DEBUG_FLG_HA:[]: HA[1]: Changing HA state to 9  
04/07 16:55:23.503 [cpp-dp-fw]: (info): QFP:0.0 Thread:120 TS:00000532153153672298  
:FW_DEBUG_FLG_HA:[]: HA[1]: Changing HA state to 10  
04/07 16:55:23.617 [buginf]: (debug): [system] Svr HA bulk sync CPP(0) complex(0)  
epoch(0) trans_id(26214421) rg_num(1)
```

このコマンドを使用すれば、デバッグ ファイルを処理しやすいように単一のファイルにマージすることができます。このコマンドは、ディレクトリ内のすべてのファイルをマージして時系列に整理します。これは、ログが非常に冗長で、複数のファイルにまたがっている場合に便利です。

```
ASR#request platform software trace slot rp active merge target bootflash:MERGED_OUTPUT.log
```

```
Creating the merged trace file: [bootflash:MERGED_OUTPUT.log]  
including all messages
```

```
Done with creation of the merged trace file: [bootflash:MERGED_OUTPUT.log]
```