

ELAM の概要

目次

[概要](#)

[Background 情報](#)

[ELAM チャレンジ](#)

[ELAM 基本](#)

[ELAM 作業の流れ](#)

[分散フォワーディング vs 中心にされる](#)

[データバス \(DBUS \) および結果バス \(RBUS \)](#)

[Local Target Logic \(LTL \)](#)

[フラッドビット](#)

[ELAM 例](#)

[内部 ASIC 名前](#)

[ELAM を使用する追加方法](#)

[関連情報](#)

概要

どんな組み込みロジック アナライザ モジュール (ELAM) であるかこの資料に、欠点、およびどのように最もよい使用それ説明されています。

Background 情報

ネットワークデバイスおよびプロトコルのより複雑な状況を使うと、ネットワークング問題のものを検出することは非常に困難である場合があります。多くの場合、フレームが特定のデバイスで正しく受信され、転送されるかどうか確認して下さい。複数のキャプチャツールが、デバッグあり、この質問に答えるのを助けるために利用可能トリックします。ただし、すべてが実行可能または実稼働 ネットワークで動作して利用可能ではないです。

内部 Cisco ASIC を検知し、パケットがどのように転送されるか理解する機能を与える ELAM はエンジニアリング ツールです。それはフォワーディング パイプラインの内で組み込みであり、パフォーマンスが control plane resources に中断なしでリアルタイムのパケットをキャプチャすることができます。それは質問に答えるのを助けます (以下を参照) :

- パケットはフォワーディングエンジン (FE) に到着しましたか。
- パケットはどんなポートおよび VLAN で受信されますか。
- パケットはどのように現われる (レイヤ2 (L2) -レイヤ4 (L4) データ) か。
- パケットはどのように変わり、どこで送信 されますか。

ELAM は非常に強力、粒状、不侵入です。それはハードウェア スイッチング プラットフォームに取り組む Cisco Technical Assistance Center (TAC) エンジニアのための貴重なトラブルシューティング ツールです。

ELAM チャレンジ

ELAM は内部 使用のための診察道具として設計されていました。CLI 構文は Cisco ASIC のために内部 コードネームを使用します、従って ELAM データの interpretation はハードウェア固有のアーキテクチャおよびフォワーディング ナレッジを必要とします。これらの詳細の多数は Cisco デバイスをクラス最高にする Cisco 内部 独自の 機能を露出するので説明することができません。

これらの理由により、ELAM は顧客サポートされた機能でし、内部 使用のための診察道具に残りました。外部コンフィギュレーション ガイドがないし、構文およびオペレーションは表記なしでバージョンからバージョンに変更されるかもしれません。

これらの身元証明要求および免責事項を、ここに ELAM が今記述されているという理由は与えられません:

- 最初に、それは TAC エンジニアのために非常によくあります問題を特定するために ELAM を使用するように。TAC は問題が断続的である場合 ELAM を行うように要求するかもしれませんが。これらのステップが不侵入である、どのように根本的な原因の分析の提供を助けることができることを理解しておくことは重要であり。
- また、時々問題の特定を助けることができる他の利用可能なツールがありません。たとえば、コンフィギュレーション変更が SPAN のための本番時間の間に割り当てられないとき、ACL は、または嵌入的なデバッグ見つかります。TAC に達する時間がないかもしれないし ELAM は最終的な解決策として持つべき非常に役立つツールである場合もあります。

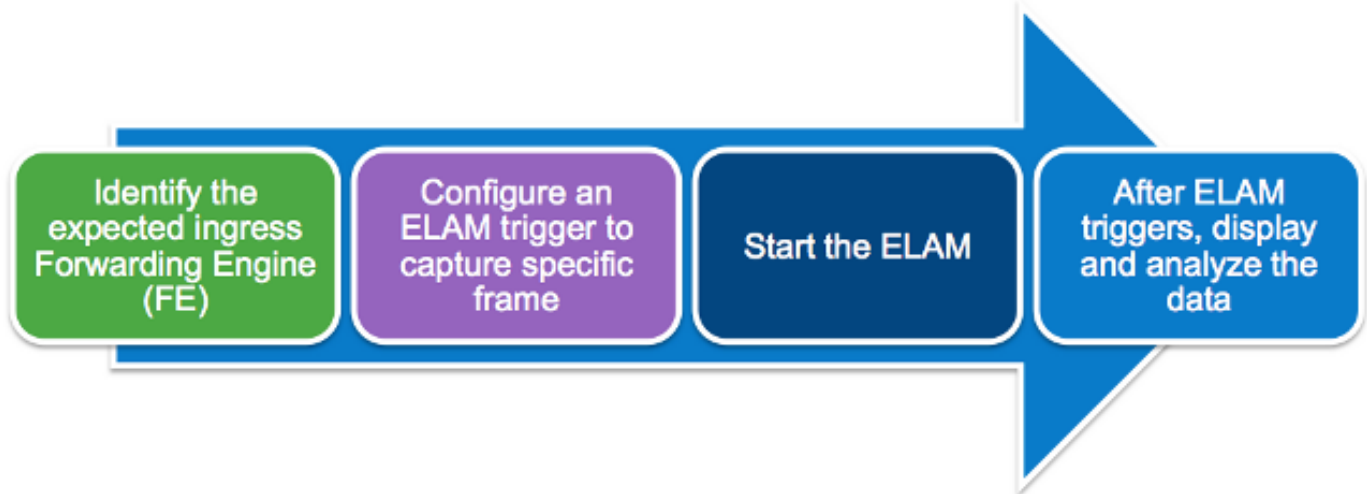
ELAM 基本

ELAM は各プラットフォームの完全な建築ナレッジなしで実行されたことができます。このセクションは Nexus 7000 シリーズ スイッチ プラットフォームと共に Cisco Catalyst 6500 および 7600 シリーズ スイッチ プラットフォームの ELAM を (単に 6500 とおよび 7600、それぞれ言われる)、行うのに必要とされる基本を記述します。

ELAM 作業の流れ

以前に述べられるように、ELAM は根本的なハードウェアに依存しています; 従って、CLI 構文は使用中のハードウェアに依存しています。ただし、各プラットフォームはこのイメージに示すように同じような作業の流れに、続きます:

注: この作業の流れが異なるプラットフォームでどのように適用されるか見るためにセクション [ELAM 例](#)を参照して下さい。



このセクションの更に詳しい情報以降であるこの4つのステップは作業の流れを記述します:

1. 期待された入力 FE を識別して下さい。プラットフォームに複数の FE があるとき、キャプチャしたいと思うパケットのための転送の決定を作る FE を識別することは重要です。正しい FE の ELAM を設定して下さい。
2. ELAM トリガーを設定して下さい。キャプチャしたいと思うパケットに特定の詳細でトリガーを設定して下さい。よくあるトリガーは送信元 および 宛先 IP アドレスか L4 ポート番号が含まれています。ELAM は複数のフィールドが規定されるようにし、論理的の設定されるすべてのフィールドで行い。
3. ELAM を開始して下さい。
4. 結果を引き起こし、表示するために ELAM を待つして下さい。

分散フォワーディング vs 中心にされる

ELAM を行うために完了する必要がある第一歩は正しい FE を識別することです。古典的な中央集中型フォワーディング (CFC) ラインカードとの 6500 はアクティブ監視プログラムが転送の決定を作る中央集中型フォワーディングを使用します。古典的または CFC ラインカードの入力が、アクティブ監視プログラムの ELAM を行う必要があるパケットに関しては。

分散フォワーディング (DFC) によって-イネーブルになったラインカードはスーパーバイザなしでラインカードで FE によって、転送の決定 ローカルでなされます。入力 DFC ラインカードが、ラインカードの ELAM 自体を行う必要があるパケットに関しては。

Nexus 7000 シリーズ スイッチ プラットフォームに関しては、すべてのラインカードは完全配られます。さらに、ほとんどのラインカードに複数の FE があります。ELAM を設定するとき、パケットが知り受信される、そのポートにマッピングする FE をポートを判別して下さい。

ハードウェアおよびフォワーディング アーキテクチャのその他の情報に関しては、これらの Cisco Live 365 技術情報を参照して下さい:

- [BRKARC-3465 Cisco Catalyst 6500 スイッチ アーキテクチャ](#)
- [BRKARC-3470 - Cisco Nexus 7000 スイッチ アーキテクチャ](#)

データバス (DBUS) および結果バス (RBUS)

転送の決定を作るために FE によって使用する DBUS は情報が含まれています。それはフレームのためのヘッダー情報と共に複数のプラットフォーム別の Internal フィールドが、含まれています。パケットがどこにの受信される、およびパケット L2-L4 情報を表示して下さいか判別を助けるために DBUS。

RBUS は FE によってなされる転送の決定が含まれています。フレームが変わった、ところで送信されるかどうか確認を助けるために RBUS を表示すれば。

Local Target Logic (LTL)

LTL はポートかポートの集まりを代表するために使用されるインデックスです。フレームがどこに受信される、そしてどこに送信されるか出典 LTL インデックスおよび宛先 LTL インデックスは示します。

注: 異なるプラットフォームおよびスーパーバイザは LTL 値をデコードするために異なるコマンドを使用します。

フラッドビット

LTL 値はように 5 つのまたはより少ない hex 数 (0xa2c、たとえば) 表示する。フラッドビットは LTL 結果の第 16 ビットです。多くの場合、RBUS は宛先 LTL インデックスのフィールドを表示する、フラッドビットのための別途のフィールドがあります。正しい LTL のためのこれらの結果をマージすることは重要です。次に、例を示します。

RBUS:

```
FLOOD ..... [1] = 1
DEST_INDEX ..... [19] = 0x48
```

この例では、宛先 LTL インデックスは **0x48** です。フラッドビットが 1 であるので、1 に LTL の第 16 ビットを設定して下さい:

```
0x00048 = 0000 0000 0000 0100 1000
           |
           +---- Flood bit, set to 1 = 0x08048
```

フラッドビットを説明した後、宛先インデックスになった **0x8048** があります。

ELAM 例

これらの例の目的は基本的な IPv4 か IPV6 ユニキャスト フローを検証するために ELAM がどのように使用されるか説明することです。この資料の [ELAM 身元証明要求](#) セクションに記述されているようにマルチキャスト、トンネルおよび MPLS のための再循環のようすすべての Internal フィールドかパケットタイプを、説明するために、それは実用的ではないです。

さまざまなデバイスによって ELAM 使用の例に関してはこれらのリンクに従って下さい:

- [Supervisor Engine 720 を含む Catalyst 6500 シリーズ スイッチの ELAM 手順](#)
- [Supervisor Engine 2T を含む Catalyst 6500 シリーズ スイッチの ELAM 手順](#)
- [Nexus 7000 M シリーズ モジュールの ELAM 手順](#)

- [Nexus 7000 M3 モジュール ELAM プロシージャ](#)
- [Nexus 7000 F1 モジュールの ELAM 手順](#)
- [Nexus 7000 F2 モジュールの ELAM 手順](#)
- [Nexus 6000 スイッチ ELAM の概要](#)

内部 ASIC 名前

参照として、各モジュールタイプに ELAM に割り当てられる内部 ASIC 名前はこの表にリストされています:

プラットフォーム	モジュールタイプ	内部 ASIC 名前
Catalyst 6500/ Cisco 7600	Sup720 (PFC3、DFC3)	スーパーマン
Catalyst 6500	Sup2T (PFC4、DFC4)	Eureka
Nexus 7000	M シリーズ (M1 および M2)	Eureka
Nexus 7000	M3 モジュール	F4
Nexus 7000	F1	オリオン
Nexus 7000	F2	クリッパー
Nexus 7000	F3	フランカー
Nexus 6000	N/A	Bigsur

ELAM を使用する追加方法

ELAM を使用するより顧客に適する方法があります。Cisco IOS[®] リリース 12.2(50)SY およびそれ以降によって、Cisco は Supervisor Engine 2T (Sup2T) を実行する 6500s のための提示プラットフォーム `datapath` コマンドを追加しました。このコマンドは特定のパケットのフォワーディング結果をキャプチャし、表示するために ELAM を使用します。

Nexus 7000 シリーズ スイッチ プラットフォームに関しては ELAM を活用するために、使いやすいスクリプトは、`elame`、Cisco IOS Release 6.2(2)に追加されました:

```
N7KA# source sys/elame
elam helper, version 1.015
```

Usage:

```
elame [<src>] <dest> [vlan <vlan#>] [vrf <vrf_name>] [int <interface> | vdc] [trace]
```

出力:

- `<src>` および `<dest>` は形式 1.2.3.4 の IPV4 アドレスです。
- `<vlan>` および `<interface>` は入力 VLAN/interface を示します。
- `vdc` は現在の仮想デバイス コンテキスト (VDC) のすべての ELAMs が使用されることを示します。
- システムが揮発性 (`elame.log`) ですべての出力のレコードを保存することを[トレース]示します。

`elame` スクリプトが F3 モジュールおよび他の N77xx ラインカードでこの時点でサポートされないことに注目して下さい。少数の機能拡張バグは `Elame` スクリプトを改良するためにファイルされ、まだビジネスユニット 検知されています。

- Cisco バグ ID [CSCuy42559](#)
- Cisco バグ ID [CSCuw60869](#)

注: Elam は内部 ツールであり、TAC/BU 管理の下でだけ使用されなければなりません。

関連情報

- [BRKARC-2011 - Ciscoスイッチおよびルータのトラブルシューティング ツールの外観- Cisco Live 365](#)
- [テクニカルサポートとドキュメント - Cisco Systems](#)