Ipv6 トラフィックを通過させるための ASA の設定

目次

概要
前提条件
要件
使用するコンポーネント
背景説明
IPv6 機能 情報
IPv6 外観
IPv4 上の IPv6 機能強化
拡張されたアドレス 指定 機能
ヘッダ フォーマット簡素化
拡張およびオプションのための改良されたサポート
機能を分類するフロー
認証およびプライバシー機能
設定
ネットワーク図
IPv6 のための設定 インターフェイス
設定 IPv6 ルーティング
IPv6 のための設定 スタティック ルーティング
OSPFv3 の IPv6 のための設定 ダイナミック ルーティング
確認
トラブルシューティング
L2 接続解決して下さい（ ND ）を
IPv4 ARP vs IPv6 ND
ND デバッグ
ND パケットキャプチャ
ND Syslog
基本的な IPv6 ルーティングを解決して下さい
IPv6 のためのルーティング プロトコル デバッグ
IPv6 のための役に立つ show コマンド
IPv6 のパケット トレーサー
IPv6-Related ASA デバッグの完全なリスト
よくある IPv6-Related 問題
不適当に設定されたサブネット
修正された EUI 64 エンコード
クライアントは一時 IPv6 アドレスをデフォルトで使用します
IPv6 FAQ

インターフェイス、同時に両方の IPv4 のためのトラフィックおよび同じの IPv6 を渡すことができますか。
IPv6 を適用でき、同じへの IPv4 ACL はインターフェイスしますか。
ASA は IPv6 のための QoS をサポートしますか。
IPv6 と NAT を使用する必要がありますか。
なぜ show failover コマンド出力のリンク ローカル IPv6 アドレスを見ますか。
既知の警告/機能拡張要求
関連情報

概要

この資料に Cisco 適応型セキュリティ アプライアンス（ASA）ソフトウェア設定する方法を（ASA） ASA バージョン 7.0(1) および それ 以降のインターネット プロトコル バージョン 6（IPv6）トラフィックを通過させるために記述されています。

前提条件

要件

このドキュメントに関する固有の要件はありません。

使用するコンポーネント

この文書に記載されている情報は Cisco ASA バージョン 7.0(1) および それ 以降に基づいています。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。 このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。 ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

背景説明

現在、IPv6 はまだ市場 浸透の点では比較的新しいです。 ただし、IPv6 設定の参考およびトラブルシューティング 要求は着実に増加しました。 この資料の目的はそれらの必要に対応し、提供することです:

- IPv6 使用方法の全般概要
- ASA の基本的な IPv6 コンフィギュレーション
- ASA を通して IPv6 接続を解決する方法についての情報

- Cisco Technical Assistance Center （TAC）によって識別されるもっとも一般的な IPv6 問題およびソリューションのリスト、

  注: IPv6 が IPv4 置換として初期にグローバルにまだあること与えられて、この資料は定期的に更新済正確さおよび関連性を維持するためです。

IPv6 機能 情報

IPv6 機能性についての重要な情報はここにあります:

- IPv6 プロトコルは ASA バージョン 7.0(1)で最初に導入されました。
- 透過モードの IPv6 のためのサポートは ASA バージョン 8.2(1)で導入されました。

IPv6 外観

IPv6 プロトコルは公共 IPv4 アドレス スペースが枯渇の方に迅速に移動したというファクトによる半ばから終わりにかけての 90 年代に、主に開発されました。 置換 プロトコルは結局必要であることネットワーク アドレス変換（NAT） 劇的に助けられた IPv4 がおよびこの問題遅らせられて、それ明らかになったが。 IPv6 プロトコルは 1998 年 12 月の RFC 2460 で公式に詳述されました。 インターネット技術特別調査委員会 （IETF） Webサイトにいる公式 RFC 2460 資料のプロトコルについて詳細を読むことができます。

IPv4 上の IPv6 機能強化

このセクションは IPv6 プロトコルと vs より古い IPv4 プロトコル含まれている機能強化を記述します。

拡張されたアドレス 指定 機能

IPv6 プロトコルは 32 ビットから 128 ビットにアドレスのアドレス階層、さらに多くのアドレス指定可能なノードおよびより簡単な自動構成のより多くのレベルをサポートするために IP アドレス サイズを増加します。 マルチキャストルーティングのスケーラビリティはマルチキャスト アドレスへの*スコープ* フィールドの付加によって改善されます。 さらに、アドレスの新型は、*エニーキャスト アドレス*を、定義されます呼出しました。 これはグループのあらゆる 1 つのノードにパケットを送信 するために使用されます。

ヘッダ フォーマット簡素化

いくつかの IPv4 ヘッダー フィールドはまたはなされたオプションの処理するパケットの一般的なケース加工費を削減するために IPv6 ヘッダの帯域幅コストを制限するために廃棄され。

拡張およびオプションのための改良されたサポート

IP ヘッダー オプションがオプションの長さの効率的 フォワーディング、より少なく厳しい制限

、および新しいオプションの概要の柔軟性を将来可能にする符号化される方法で変更します。

機能を分類するフロー

新しい機能は送信側が特別な処理を要求するデフォルト以外の Quality of Service （ QoS ）または*リアルタイムサービス*のような特定のトラフィック *トラフィック フロー*に属するパケットの分類を有効に するために追加されます。

認証およびプライバシー機能

認証をサポートするために使用する拡張 データ統合および（ オプションの ）データの機密保持は IPv6 のために規定 されます。

設定

このセクションは IPv6 の使用のための Cisco ASA を設定する方法を記述します。

> **注:** このセクションで使用されているコマンドの詳細を調べるには、Command Lookup Tool（登録ユーザ専用 ）を使用してください。

ネットワーク図

これはこの資料の全体にわたって使用する例のための IPv6 トポロジーです:

IPv6 のための設定 インターフェイス

IPv6 トラフィックを ASA を通して通過させるために、最初に少なくとも 2 つのインターフェイスの IPv6 を有効に して下さい。 この例に **Gi0/0** の内部インターフェイスから **Gi0/1** の outside インターフェイスにトラフィックを通過させることを IPv6 が可能にする方法を記述されています :

```
ASAv(config)# interface GigabitEthernet0/0
ASAv(config-if)# ipv6 enable

ASAv(config)# interface GigabitEthernet0/1
ASAv(config-if)# ipv6 enable
```

今インターフェイスの両方の IPv6 アドレスを設定できます。

> **注:** この例では、fc00::/7 のユニークなローカルアドレス（ ULA ）領域のアドレスは使用されます、従ってアドレスすべては **FD** から始まります（ のような、fdxx: xxxx: xxxx.... ）。 また IPv6 アドレスを書くとき、（ **: FD01::1/64 が FD01:0000:0000:0000:0000:0000:0000:00001** と同じであるようにゼロの行を表すために **:** ）二重コロンを使用できます。

```
ASAv(config)# interface GigabitEthernet0/0
```

```
ASAv(config-if)# ipv6 address fd03::1/64
ASAv(config-if)# nameif inside
ASAv(config-if)# security-level 100

ASAv(config)# interface GigabitEthernet0/1
ASAv(config-if)# ipv6 address fd02::2/64
ASAv(config-if)# nameif outside
ASAv(config-if)# security-level 0
```

今基本層 2 があるはずです（アドレス fd02::1 の外部 VLAN のアップストリーム ルータへの L2)/Layer 3 （L3）接続:

```
ASAv(config-if)# ping fd02::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to fd02::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/10 ms
```

設定 IPv6 ルーティング

IPv4 と全く同様に、直接接続されした サブネットのホストとの IPv6 接続があるのにそれらに達する方法を、まだ外部ネットワークに知るためにルーティングがなければなりません。 最初の例に fd02::1 のネクストホップ アドレスが付いている outside インターフェイスによって IPv6 ネットワークすべてにアクセスするために静的デフォルト ルートを設定する方法を示されています。

IPv6 のための設定 スタティック ルーティング

IPv6 のためのスタティック ルーティングを設定するためにこの情報を使用して下さい:

```
ASAv(config)# ipv6 route outside 0::0/0 fd02::1
ASAv(config)# show ipv6 route

IPv6 Routing Table - 7 entries
Codes: C - Connected, L - Local, S - Static
O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2, B - BGP
L fd02::2/128 [0/0]
via ::, outside
C fd02::/64 [0/0]
via ::, outside
L fd03::1/128 [0/0]
via ::, inside
C fd03::/64 [0/0]
via ::, inside
L fe80::/10 [0/0]
via ::, inside
via ::, outside
L ff00::/8 [0/0]
via ::, inside
via ::, outside
S ::/0 [1/0]
via fd02::1, outsideASAv(config)#
```

このとき示されているように、外部サブネットのホストへの接続があります:

```
ASAv(config)# ping fd99::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to fd99::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

```
ASAv(config)#
```

> **注:** IPv6 のためのルーティングを処理するためにダイナミック ルーティング プロトコルが望まれる場合それを同様に設定できます。 これは次の セクションに説明があります。

OSPFv3 の IPv6 のための設定 ダイナミック ルーティング

最初に、Integrated Services Router（ISR）モジュール アップストリームの Open Shortest Path First バージョン 3（OSPFv3）設定を Cisco 881 シリーズ検査する必要があります（ISR）:

```
C881#show run | sec ipv6
ipv6 unicast-routing

!--- This enables IPv6 routing in the Cisco IOS®.

.....
ipv6 ospf 1 area 0
address-family ipv6 unicast
passive-interface default
no passive-interface Vlan302

!--- This is the interface to send OSPF Hellos to the ASA.

default-information originate always

!--- Always distribute the default route.

redistribute static
ipv6 route ::/0 FD99::2

!--- Creates a static default route for IPv6 to the internet.
```

関連するインターフェイス 設定はここにあります:

```
C881#show run int Vlan302
interface Vlan302
....
ipv6 address FD02::1/64
ipv6 ospf 1 area 0
C881#
```

OSPF *Hello*パケットが outside インターフェイスの ISR から見られることを確認するために ASA パケットキャプチャを使用できます:

```
ASAv(config)# show run access-list test_ipv6
access-list test_ipv6 extended permit ip any6 any6
ASAv(config)# show cap
capture capout type raw-data access-list test_ipv6 interface outside
[Capturing - 37976 bytes]
ASAv(config)# show cap capout

367 packets captured

1: 11:12:04.949474 fe80::250:56ff:fe9d:34a8 > ff02::1:ff9d:34a8: icmp6:
neighbor sol: who has fe80::250:56ff:fe9d:34a8 [class 0xe0]
2: 11:12:06.949444 fe80::250:56ff:fe9d:34a8 > ff02::1:ff9d:34a8: icmp6:
neighbor sol: who has fe80::250:56ff:fe9d:34a8 [class 0xe0]
 3: 11:12:07.854768 fe80::c671:feff:fe93:b516 > ff02::5: ip-proto-89 40
[hlim 1]
4: 11:12:07.946545 fe80::250:56ff:fe9d:34a8 > ff02::1:ff9d:34a8: icmp6:
neighbor sol: who has fe80::250:56ff:fe9d:34a8 [class 0xe0]
```

```
5: 11:12:08.949459 fe80::250:56ff:fe9d:34a8 > ff02::1:ff9d:34a8: icmp6:
neighbor sol: who has fe80::250:56ff:fe9d:34a8 [class 0xe0]
6: 11:12:09.542772 fe80::217:fff:fe17:af80 > ff02::5: ip-proto-89 40
[hlim 1]
....
 13: 11:12:16.983011 fe80::c671:feff:fe93:b516 > ff02::5: ip-proto-89 40
[hlim 1]
14: 11:12:18.947170 fe80::250:56ff:fe9d:34a8 > ff02::1:ff9d:34a8: icmp6:
neighbor sol: who has fe80::250:56ff:fe9d:34a8 [class 0xe0]
15: 11:12:19.394831 fe80::217:fff:fe17:af80 > ff02::5: ip-proto-89 40
[hlim 1]
16: 11:12:19.949444 fe80::250:56ff:fe9d:34a8 > ff02::1:ff9d:34a8: icmp6:
21: 11:12:26.107477 fe80::c671:feff:fe93:b516 > ff02::5: ip-proto-89 40
[hlim 1]
ASAv(config)#
```

前のパケットキャプチャでは、OSPF（ip-proto-89）パケットが ISR の正しいインターフェイスに対応する IPv6 リンク ローカル アドレスから着くことがわかります、:

```
C881#show ipv6 interface brief
......
Vlan302 [up/up]
 FE80::C671:FEFF:FE93:B516
FD02::1
C881#
```

今 ISR の隣接関係を確立するために ASA の OSPFv3 プロセスを作成できます:

```
ASAv(config)# ipv6 router ospf 1
ASAv(config-rtr)# passive-interface default
ASAv(config-rtr)# no passive-interface outside
ASAv(config-rtr)# log-adjacency-changes
ASAv(config-rtr)# redistribute connected
ASAv(config-rtr)# exit
```

ASA outside インターフェイスに OSPF 設定を適用して下さい:

```
ASAv(config)# interface GigabitEthernet0/1
ASAv(config-if)# ipv6 ospf 1 area 0
ASAv(config-if)# end
```

これにより ASA は IPv6 サブネットのブロードキャスト OSPF Helloパケットを送信します必要があります。 ルータとの隣接関係を確認するために**提示 IPv6 OSPFネイバ** コマンドを入力して下さい:

```
ASAv# show ipv6 ospf neighbor

Neighbor ID Pri State Dead Time Interface ID Interface
 14.38.104.1 1 FULL/BDR 0:00:33 14 outside
```

ID のために最も数字の大きい設定された IPv4 アドレスをデフォルトで使用するのでまた ISR のネイバー ID を確認できます:

```
C881#show ipv6 ospf 1
 Routing Process "ospfv3 1" with ID 14.38.104.1
Supports NSSA (compatible with RFC 3101)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
It is an autonomous system boundary router
Redistributing External Routes from,
static
Originate Default Route with always

!--- Notice the other OSPF settings that were configured.

Router is not originating router-LSAs with maximum metric
```

```
....

C881#
```

ASA は今 ISR からのデフォルト IPv6 ルートを学習する必要があります。 これを確認するために、**show ipv6 route** コマンドを入力して下さい:

```
ASAv# show ipv6 route

IPv6 Routing Table - 8 entries
Codes: C - Connected, L - Local, S - Static
O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2, B - BGP
O 2001:aaaa:aaaa:aaaa::/64 [110/10]
via ::, outside
L fd02::2/128 [0/0]
via ::, outside
C fd02::/64 [0/0]
via ::, outside
L fd03::1/128 [0/0]
via ::, inside
C fd03::/64 [0/0]
via ::, inside
L fe80::/10 [0/0]
via ::, inside
via ::, outside
L ff00::/8 [0/0]
via ::, inside
via ::, outside
OE2 ::/0 [110/1], tag 1

!--- Here is the learned default route.

 via fe80::c671:feff:fe93:b516, outside
ASAv#
```

ASA の IPv6 のためのインターフェイス設定およびルーティング機能の基本設定は現在完了しました。

確認

現在、この設定に使用できる確認手順はありません。

トラブルシューティング

IPv6 接続におけるトラブルシューティング 手順は少数の違いと IPv4 接続を解決するために使用される同じ方法論のほとんどに続きます。 トラブルシューティング観点から、IPv4 間の最も重要な違いの 1 つおよび IPv6 はこともはや IPv6 で存在 する アドレス解決プロトコル （ ARP ）ではないです。 ARP の使用の代りにローカルLANセグメントの IP アドレスを解決するために、IPv6 は Neighbor Discovery （ ND ）と呼ばれるプロトコルを使用します。

ND が Media Access Control （ MAC ）アドレス リゾリューションのためのバージョン 6 （ ICMPv6 ）を Internet Control Message Protocol （ ICMP ） 活用 することを理解しておくこともまた重要です。 IPv6 ND についての詳細は *CLI 本 1* の IPv6 Neighbor Discovery セクションの ASA IPv6 コンフィギュレーション ガイドで見つけることができます*: Cisco ASA シリーズ 操作全般 CLI コンフィギュレーション ガイド*、RFC 4861 の *9.4* または。

現在、ほとんどの IPv6-related トラブルシューティングは ND、ルーティング、またはサブネット コンフィギュレーションに関する問題を含みます。 これはこれらがまた IPv4 と IPv6 間の主な違いであるというファクトが多分原因です。 NAT の使用が IPv6 で非常に落胆するプライベート アドレッシングがもはや IPv4 にだった方法活用 されないので、ND 作業別様に ARP と、および内部ネットワーク アドレッシングはまたかなり異なって（RFC 1918 の後で）。 これらの違いが理解されればおよび/または L2/L3 問題が解決されれば、レイヤ4 （L4）のトラブルシューティング プロセスは IPv4 に使用するそれと以上に TCP/UDP およびハイレイヤプロトコルが同じ使用される）本質的に機能するので本質的に同じです（IPバージョンに関係なく。

L2 接続解決して下さい（ND）を

IPv6 で L2 接続を解決するために使用するほとんどの基本的なコマンドは**提示 IPv6 隣接 [nameif]**コマンドです、IPv4 のための **show arp** の等量の。

次に出力例を示します。

```
ASAv(config)# show ipv6 neighbor outside
IPv6 Address Age Link-layer Addr State Interface
fd02::1 0 c471.fe93.b516 REACH outside
fe80::c671:feff:fe93:b516 32 c471.fe93.b516 DELAY outside
fe80::e25f:b9ff:fe3f:1bbf 101 e05f.b93f.1bbf STALE outside
fe80::b2aa:77ff:fe7c:8412 101 b0aa.777c.8412 STALE outside
fe80::213:c4ff:fe80:5f53 101 0013.c480.5f53 STALE outside
fe80::a64c:11ff:fe2a:60f4 101 a44c.112a.60f4 STALE outside
fe80::217:fff:fe17:af80 99 0017.0f17.af80 STALE outside
ASAv(config)#
```

この出力では、**c471.fe93.b516** の MAC アドレスのデバイスに属する **fd02::1** の IPv6 アドレスについては正常な解決を表示できます。

> **注:** ルータがまたこのインターフェイスのための自動割り当て リンク ローカル アドレスを備えているので同一ルータ インターフェイスのMACアドレスが前の出力に二度現われることに注意するかもしれません。 リンク ローカル アドレスは直接接続されたネットワークの通信のためにしか使用できないデバイス特有のアドレスです。 ルータはリンク ローカル アドレスによってパケットを転送しませんが、直接接続されたネットワーク セグメントの通信のためだけむしろです。 多くの IPv6 ルーティング プロトコルは（OSPFv3 のような）L2 セグメントのルーティング プロトコル 情報を共有するためにリンク ローカル アドレスを利用します。

ND キャッシュを消去するために、**クリア IPv6 neighbors コマンドを**入力して下さい。 ND があるホストのために失敗した場合、**デバッグ IPv6 nd** コマンドを入力でき、またパケットキャプチャを行い、L2 レベルに発生する判別するために syslog を、確認します。 IPv6 アドレスのための MAC アドレスを解決するために IPv6 ND が ICMPv6 メッセージを使用することを覚えていて下さい。

IPv4 ARP vs IPv6 ND

IPv4 のための ARP および IPv6 のための ND のこの比較 テーブルを検討して下さい:

IPv4 ARP	IPv6 ND
（だれが 10.10.10.1 があるか ARP要求か。）	隣接要請
ARP応答（10.10.10.1 は dead.dead.dead にあります）	隣接アドバタイズメント

次のシナリオでは、ND は outside インターフェイスにある *fd02::1* ホストの MAC アドレスを解決しません。

ND デバッグ

デバッグ IPv6 nd コマンドの出力はここにあります:

```
ICMPv6-ND: Sending NS for fd02::1 on outside

!--- "Who has fd02::1"

ICMPv6-ND: Sending NS for fd02::1 on outside
ICMPv6-ND: Sending NS for fd02::1 on outside
ICMPv6-ND: INCMP deleted: fd02::1
ICMPv6-ND: INCMP -> DELETE: fd02::1
ICMPv6-ND: DELETE -> INCMP: fd02::1
ICMPv6-ND: Sending NS for fd02::1 on outside
ICMPv6-ND: Sending NS for fd02::1 on outside
ICMPv6-ND: Sending NA for fd02::2 on outside

!--- "fd02::2 is at dead.dead.dead"

ICMPv6-ND: Sending NS for fd02::1 on outside
ICMPv6-ND: INCMP deleted: fd02::1
ICMPv6-ND: INCMP -> DELETE: fd02::1
ICMPv6-ND: DELETE -> INCMP: fd02::1

!--- Here is where the ND times out.

ICMPv6-ND: Sending NS for fd02::1 on outside
ICMPv6-ND: Sending NS for fd02::1 on outside
```

このデバッグ 出力では、**fd02::2** からの隣接アドバタイズメントが決して受け取られないようです。 これが実際に事実であるかどうか確認するためにパケットキャプチャをチェックできます。

ND パケットキャプチャ

> **注:** ASA 現在で 9.4(1) を、まだ access-list 必要とされます IPv6 パケットキャプチャにリリースして下さい。 機能拡張要求は Cisco バグ ID CSCtn09836 とこれをトラッキングするためにファイルされました。

Access Control List （ ACL ）およびパケットキャプチャを設定して下さい:

```
ASAv(config)# access-list test_ipv6 extended permit ip any6 any6
ASAv(config)# cap capout interface outside access-list test_ipv6
```

ASA から **fd02::1** に PING を始めて下さい:

```
ASAv(config)# show cap capout
....
23: 10:55:10.275284 fd02::2 > ff02::1:ff00:1: icmp6: neighbor sol: who has
fd02::1 [class 0xe0]
24: 10:55:10.277588 fd02::1 > fd02::2: icmp6: neighbor adv: tgt is fd02::1
[class 0xe0]
26: 10:55:11.287735 fd02::2 > ff02::1:ff00:1: icmp6: neighbor sol: who has
fd02::1 [class 0xe0]
27: 10:55:11.289642 fd02::1 > fd02::2: icmp6: neighbor adv: tgt is fd02::1
```

```
[class 0xe0]
28: 10:55:12.293365 fd02::2 > ff02::1:ff00:1: icmp6: neighbor sol: who has
fd02::1 [class 0xe0]
29: 10:55:12.298538 fd02::1 > fd02::2: icmp6: neighbor adv: tgt is fd02::1
[class 0xe0]
32: 10:55:14.283341 fd02::2 > ff02::1:ff00:1: icmp6: neighbor sol: who has
fd02::1 [class 0xe0]
33: 10:55:14.285690 fd02::1 > fd02::2: icmp6: neighbor adv: tgt is fd02::1
[class 0xe0]
35: 10:55:15.287872 fd02::2 > ff02::1:ff00:1: icmp6: neighbor sol: who has
fd02::1 [class 0xe0]
36: 10:55:15.289825 fd02::1 > fd02::2: icmp6: neighbor adv: tgt is fd02::1
[class 0xe0]
```

パケットキャプチャに示すように、fd02::1 からの隣接アドバタイズメントは受け取られます。 ただし、アドバタイズメントはデバッグ 出力に示すようにどういうわけか、処理されません。 それ以上のチェックの場合、syslog を表示できます。

ND Syslog

いくつかの ND syslog 例ここにあります:

```
May 13 2015 10:55:10: %ASA-7-609001: Built local-host identity:fd02::2
May 13 2015 10:55:10: %ASA-6-302020: Built outbound ICMP connection for faddr
ff02::1:ff00:1/0 gaddr fd02::2/0 laddr fd02::2/0(any)
May 13 2015 10:55:10: %ASA-3-325003: EUI-64 source address check failed.  Dropped
packet from outside:fd02::1/0 to fd02::2/0 with source MAC address c471.fe93.b516.
May 13 2015 10:55:10: %ASA-3-313008: Denied IPv6-ICMP type=136, code=0 from fd02::1
on interface outside
May 13 2015 10:55:11: %ASA-3-325003: EUI-64 source address check failed.  Dropped
packet from outside:fd02::1/0 to fd02::2/0 with source MAC address c471.fe93.b516.
May 13 2015 10:55:11: %ASA-3-313008: Denied IPv6-ICMP type=136, code=0 from fd02::1
on interface outside
May 13 2015 10:55:12: %ASA-3-325003: EUI-64 source address check failed.  Dropped
packet from outside:fd02::1/0 to fd02::2/0 with source MAC address c471.fe93.b516.
May 13 2015 10:55:12: %ASA-3-313008: Denied IPv6-ICMP type=136, code=0 from fd02::1
on interface outside
May 13 2015 10:55:14: %ASA-3-325003: EUI-64 source address check failed.  Dropped
packet from outside:fd02::1/0 to fd02::2/0 with source MAC address c471.fe93.b516.
May 13 2015 10:55:14: %ASA-3-313008: Denied IPv6-ICMP type=136, code=0 from fd02::1
on interface outside
May 13 2015 10:55:15: %ASA-3-325003: EUI-64 source address check failed.  Dropped
packet from outside:fd02::1/0 to fd02::2/0 with source MAC address c471.fe93.b516.
May 13 2015 10:55:15: %ASA-3-313008: Denied IPv6-ICMP type=136, code=0 from fd02::1
on interface outside
```

これらの syslog の中では、fd02::1 の ISR からの ND 隣接アドバタイズメントパケットが壊れる修正された拡張固有の識別番号（EUI）が廃棄された原因 64 の（修正された EUI-64）形式チェックであることがわかります。

> **ヒント**： この特定の問題に関する詳細についてはこの資料のセクションを*符号化する修正された EUI-64 アドレス*を参照して下さい。 このトラブルシューティング ロジックはいろいろな種類のドロップする原因に ACL が特定のインターフェイスの許可 ICMPv6 か、または IPv6 においての L2 接続上の問題を引き起こす場合がある Unicast Reverse Path Forwarding（URPF）チェック失敗が発生するときのような同様に適用することができます。

基本的な IPv6 ルーティングを解決して下さい

IPv6 が使用されるときルーティング プロトコルにおけるトラブルシューティング 手順は IPv4 が使用されるとき本質的に同じそれらです。 **Debug および Show コマンド**、またパケットキャプチャの使用は、ルーティング プロトコルが期待どおりに動作しないという理由を確認する試みと役立ちます。

IPv6 のためのルーティング プロトコル デバッグ

このセクションは IPv6 に有用な debug コマンドを提供します。

グローバル な IPv6 ルーティング デバッグ

IPv6 ルーティング テーブル変更すべてを解決するためにデバッグを**ルーティングするデバッグ IPv6 を**使用できます:

```
ASAv# clear ipv6 ospf 1 proc

Reset OSPF process? [no]: yes
ASAv# IPv6RT0: ospfv3 1, Route update to STANDBY with epoch: 2 for
2001:aaaa:aaaa:aaaa::/64
IPv6RT0: ospfv3 1, Delete 2001:aaaa:aaaa:aaaa::/64 from table
IPv6RT0: ospfv3 1, Delete backup for fd02::/64
IPv6RT0: ospfv3 1, Route update to STANDBY with epoch: 2 for ::/0
IPv6RT0: ospfv3 1, Delete ::/0 from table
IPv6RT0: ospfv3 1, ipv6_route_add_core for 2001:aaaa:aaaa:aaaa::/64 [110/10],
next-hop :: nh_source :: via interface outside route-type 2
IPv6RT0: ospfv3 1, Add 2001:aaaa:aaaa:aaaa::/64 to table
IPv6RT0: ospfv3 1, Added next-hop :: over outside for 2001:aaaa:aaaa:aaaa::/64,
[110/10]
IPv6RT0: ospfv3 1, ipv6_route_add_core Route update to STANDBY with epoch: 2 for
2001:aaaa:aaaa:aaaa::/64
IPv6RT0: ipv6_route_add_core: input add 2001:aaaa:aaaa:aaaa::/64
IPv6RT0: ipv6_route_add_core: output add 2001:aaaa:aaaa:aaaa::/64
IPv6RT0: ospfv3 1, ipv6_route_add_core for fd02::/64 [110/10], next-hop ::
nh_source :: via interface outside route-type 2
IPv6RT0: ospfv3 1, ipv6_route_add_core for ::/0 [110/1], next-hop
fe80::c671:feff:fe93:b516
nh_source fe80::c671:feff:fe93:b516 via interface outside route-type 16
IPv6RT0: ospfv3 1, Add ::/0 to table
IPv6RT0: ospfv3 1, Added next-hop fe80::c671:feff:fe93:b516 over outside for ::/0,
[110/1]
IPv6RT0: ospfv3 1, ipv6_route_add_core Route update to STANDBY with epoch: 2 for ::/0
IPv6RT0: ipv6_route_add_core: input add ::/0
IPv6RT0: ipv6_route_add_core: output add ::/0
IPv6RT0: ospfv3 1, ipv6_route_add_core for 2001:aaaa:aaaa:aaaa::/64 [110/10],
next-hop :: nh_source :: via interface outside route-type 2
IPv6RT0: ospfv3 1, Route add 2001:aaaa:aaaa:aaaa::/64 [owner]
IPv6RT0: ospfv3 1, ipv6_route_add_core Route update to STANDBY with epoch: 2 for
2001:aaaa:aaaa:aaaa::/64
IPv6RT0: ipv6_route_add_core: input add 2001:aaaa:aaaa:aaaa::/64
IPv6RT0: ipv6_route_add_core: output add 2001:aaaa:aaaa:aaaa::/64
IPv6RT0: ospfv3 1, ipv6_route_add_core for fd02::/64 [110/10], next-hop ::
nh_source :: via interface outside route-type 2
IPv6RT0: ospfv3 1, Reuse backup for fd02::/64, distance 110
IPv6RT0: ospfv3 1, ipv6_route_add_core for ::/0 [110/1], next-hop
fe80::c671:feff:fe93:b516 nh_source fe80::c671:feff:fe93:b516 via interface outside
route-type 16
```

```
IPv6RT0: ospfv3 1, Route add ::/0 [owner]
IPv6RT0: ospfv3 1, ipv6_route_add_core Route update to STANDBY with epoch: 2 for ::/0
IPv6RT0: ipv6_route_add_core: input add ::/0
IPv6RT0: ipv6_route_add_core: output add ::/0
```

OSPFv3 デバッグ

OSPFv3 問題を解決するためにデバッグ IPv6 ospf コマンドを使用できます:

```
ASAv# debug ipv6 ospf ?

adj OSPF adjacency events
database-timer OSPF database timer
events OSPF events
flood OSPF flooding
graceful-restart OSPF Graceful Restart processing
hello OSPF hello events
ipsec OSPF ipsec events
lsa-generation OSPF lsa generation
lsdb OSPF database modifications
packet OSPF packets
retransmission OSPF retransmission events
spf OSPF spf
```

OSPFv3 プロセスが再起動した後有効に なるデバッグすべてのための出力例はここにあります:

```
ASAv# clear ipv6 ospf 1
OSPFv3: rcv. v:3 t:1 l:44 rid:192.168.128.115
aid:0.0.0.0 chk:a9ac inst:0 from outside
OSPFv3: Rcv hello from 192.168.128.115 area 0 from outside fe80::217:fff:fe17:af80
interface ID 142
OSPFv3: End of hello processingpr
OSPFv3: rcv. v:3 t:1 l:44 rid:14.38.104.1
aid:0.0.0.0 chk:bbf3 inst:0 from outside
OSPFv3: Rcv hello from 14.38.104.1 area 0 from outside fe80::c671:feff:fe93:b516
interface ID 14
OSPFv3: End of hello processingo
ASAv# clear ipv6 ospf 1 process

Reset OSPF process? [no]: yes
ASAv#
OSPFv3: Flushing External Links
Insert LSA 0 adv_rtr 172.16.118.1, type 0x4005 in maxage
OSPFv3: Add Type 0x4005 LSA ID 0.0.0.0 Adv rtr 172.16.118.1 Seq 80000029 to outside
14.38.104.1 retransmission list
....

!--- The neighbor goes down:

OSPFv3: Neighbor change Event on interface outside
OSPFv3: DR/BDR election on outside
OSPFv3: Elect BDR 14.38.104.1
OSPFv3: Elect DR 192.168.128.115
OSPFv3: Schedule Router LSA area: 0, flag: Change
OSPFv3: Schedule Router LSA area: 0, flag: Change
OSPFv3: Schedule Prefix DR LSA intf outside
OSPFv3: Schedule Prefix Stub LSA area 0
OSPFv3: 14.38.104.1 address fe80::c671:feff:fe93:b516 on outside is dead, state DOWN
....

!--- The neighbor resumes the exchange:

OSPFv3: Rcv DBD from 14.38.104.1 on outside seq 0xd09 opt 0x0013 flag 0x7 len 28
 mtu 1500 state EXSTART
```

```
OSPFv3: First DBD and we are not SLAVE
OSPFv3: rcv. v:3 t:2 l:168 rid:14.38.104.1
     aid:0.0.0.0 chk:5aa3 inst:0 from outside
OSPFv3: Rcv DBD from 14.38.104.1 on outside seq 0x914 opt 0x0013 flag 0x2 len 168
 mtu 1500 state EXSTART
OSPFv3: NBR Negotiation Done. We are the MASTER
OSPFv3: outside Nbr 14.38.104.1: Summary list built, size 0
OSPFv3: Send DBD to 14.38.104.1 on outside seq 0x915 opt 0x0013 flag 0x1 len 28
OSPFv3: rcv. v:3 t:2 l:28 rid:192.168.128.115
     aid:0.0.0.0 chk:295c inst:0 from outside
OSPFv3: Rcv DBD from 192.168.128.115 on outside seq 0xfeb opt 0x0013 flag 0x7 len 28
 mtu 1500 state EXSTART
OSPFv3: NBR Negotiation Done. We are the SLAVE
OSPFv3: outside Nbr 192.168.128.115: Summary list built, size 0
OSPFv3: Send DBD to 192.168.128.115 on outside seq 0xfeb opt 0x0013 flag 0x0 len 28
OSPFv3: rcv. v:3 t:2 l:28 rid:14.38.104.1
     aid:0.0.0.0 chk:8d74 inst:0 from outside
OSPFv3: Rcv DBD from 14.38.104.1 on outside seq 0x915 opt 0x0013 flag 0x0 len 28
 mtu 1500 state EXCHANGE
....

!--- The routing is re-added to the OSPFv3 neighbor list:

OSPFv3: Add Router 14.38.104.1 via fe80::c671:feff:fe93:b516, metric: 10
 Router LSA 14.38.104.1/0, 1 links
   Link 0, int 14, nbr 192.168.128.115, nbr int 142, type 2, cost 1
     Ignore newdist 11 olddist 10
```

Enhanced Interior Gateway Routing Protocol (EIGRP)

ASA の EIGRP は IPv6 の使用をサポートしません。 *CLI 本 1* の EIGRP セクションのためのガイドラインを参照して下さい*: Cisco ASA シリーズ 操作全般 CLI コンフィギュレーション ガイド*、詳細については *9.4*。

ボーダー ゲートウェイ プロトコル (BGP)

この **debug コマンドは** IPv6 が使用されるとき BGP を解決するために使用することができます:

```
ASAv# debug ip bgp ipv6 unicast ?

X:X:X:X::X IPv6 BGP neighbor address
keepalives BGP keepalives
updates BGP updates
<cr>
```

IPv6 のための役に立つ show コマンド

IPv6 問題を解決するためにこれらの **show コマンドを**使用できます:

- **show ipv6 route**
- **show ipv6 interface brief**
- **IPv6 ospf <process ID> を示して下さい**
- **IPv6 トラフィックを示して下さい**
- **IPv6 ネイバーを示して下さい**

- IPv6 icmp を示して下さい

IPv6 のパケット トレーサー

IPv4 のと同様に ASA の IPv6 と組み込みパケット トレーサー機能性を使用できます。 OSPF によって 881 インターフェイスから学習されるデフォルト ルートが付いているインターネットにある 5555::1 で Webサーバに接続するように試みる fd03::2 で内部ホストを模倣するためにパケット トレーサー機能性が使用される例はここにあります:

```
ASAv# packet-tracer input inside tcp fd03::2 10000 5555::1 80 detailed

Phase: 1
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7fffd59ca0f0, priority=1, domain=permit, deny=false
       hits=2734, user_data=0x0, cs_id=0x0, l3_type=0xdd86
       src mac=0000.0000.0000, mask=0000.0000.0000
       dst mac=0000.0000.0000, mask=0100.0000.0000
       input_ifc=inside, output_ifc=any

Phase: 2
Type: ROUTE-LOOKUP
Subtype: Resolve Egress Interface
Result: ALLOW
Config:
Additional Information:
found next-hop fe80::c671:feff:fe93:b516 using egress ifc  outside

Phase: 3
Type: NAT
Subtype: per-session
Result: ALLOW
Config:
Additional Information:
 Forward Flow based lookup yields rule:
 in  id=0x7fffd589cc30, priority=1, domain=nat-per-session, deny=true
       hits=1166, user_data=0x0, cs_id=0x0, reverse, use_real_addr, flags=0x0,
protocol=6
       src ip/id=::/0, port=0, tag=any
       dst ip/id=::/0, port=0, tag=any
       input_ifc=any, output_ifc=any

<<truncated output>>

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

ASAv#
```

出力 MAC アドレスが 881 インターフェイスのリンク ローカル アドレスであることに注意して下さい。 ルータ 使用 リンク ローカル IPv6 を、多くのダイナミック ルーティング プロトコルのために、以前に述べられるように隣接関係を確立するために当たります。

IPv6-Related ASA デバッグの完全なリスト

IPv6 問題を解決するために使用できるデバッグはここにあります:

```
ASAv# debug ipv6 ?

dhcp IPv6 generic dhcp protocol debugging
dhcprelay IPv6 dhcp relay debugging
icmp ICMPv6 debugging
interface IPv6 interface debugging
mld IPv6 Multicast Listener Discovery debugging
nd IPv6 Neighbor Discovery debugging
ospf OSPF information
packet IPv6 packet debugging
routing IPv6 routing table debugging
```

よくある IPv6-Related 問題

このセクションはもっとも一般的な IPv6-related 問題を解決する方法を記述します。

不適当に設定されたサブネット

多くの IPv6 TAC ケースは IPv6 がどのようにについての機能するか、または管理者が原因で IPv4-specific プロセスの使用の IPv6 を設定するように試みるかナレッジの一般の欠如が生成された原因です。

たとえば、TAC は管理者 インターネットサービスプロバイダー （ ISP ）によって IPv6 アドレスの \56 ブロックが割り当てられたケースを見ました。 管理者は ASA outside インターフェイスにそしてアドレスおよび完全な \56 サブネットを割り当て、内部サーバのために使用するために内部範囲を選択します。 ただし、IPv6 と、内部ホストすべてはまたルーティング可能 な IPv6 アドレスを使用し IPv6 アドレスブロックは必要に応じて分割されたより小さいサブネットであるはずです。 このシナリオでは、と同時に割り当てられた \56 ブロックの一部多くの \64 サブネットを作成できます。

> **ヒント**： その他の情報に関しては RFC 4291 を参照して下さい。

修正された EUI 64 エンコード

ASA は修正された EUI-64-encoded IPv6 アドレスを必要とするために設定することができます。 EUI は、RFC 4291 によって、ホストがそれ自身にユニークな 64 ビット IPv6 インターフェイス識別子 (EUI-64) を割り当てるようにします。 この機能は IPv6 アドレス 指定のために DHCP を利用するために要件を取除くので、IPv4 上の長所です。

IPv6 enforce-eui64 nameif コマンドによってこの機能拡張を必要とするために ASA が設定される場合ローカルサブネットの他のホストから多分多くの Neighbor Discovery 要請およびアドバタイ

ズメントを廃棄します。

> **ヒント**： 詳細については、知識 IPv6 EUI-64 ビットアドレス Ciscoサポート コミュニティ資料を参照して下さい。

クライアントは一時 IPv6 アドレスをデフォルトで使用します

デフォルトで、多くのクライアント オペレーティング システム（OS）は、Microsoft Windows バージョン 7 および 8 のような、Macintosh OS X IPv6 ステートレス アドレスの自動設定メカニズム（SLAAC）によって拡張プライバシーのためにおよび Linuxベース システム、自動割り当て一時 IPv6 アドレスを使用します。

Cisco TAC はホストが一時アドレスおよびない静的割り当て済みアドレスからのトラフィックを生成するのでこれが環境で予想外の問題を引き起こしたいくつかのケースを見ました。 その結果、ACL によりおよびホスト ルーティングは廃棄されるか、または不適当にルーティングされるようになるにトラフィックを引き起こすかもしれませんホスト通信が失敗します。

この状況を当たるために使用する 2 つのメソッドがあります。 動作はクライアント システムでそれぞれディセーブルにすることができますまたは ASA および Cisco IOS ® ルータのこの動作をディセーブルにすることができます。 ASA かルータ側で、この動作を引き起こす ルータ アドバタイズメント（RA） Message フラグを修正して下さい。

個々のクライアント システムのこの動作をディセーブルにするために次の セクションを参照して下さい。

Microsoft Windows

Microsoft Windows システムのこの動作をディセーブルにするためにこれらのステップを完了して下さい:

1. Microsoft Windows では、高いコマンド プロンプト（管理者として実行）を開いて下さい。
2. ランダム IP アドレス世代別機能をディセーブルにするためにこのコマンドを入力し次に『Enter』 を押して下さい:
   ```
   netsh interface ipv6 set global randomizeidentifiers=disabled
   ```
3. Microsoft Windows を EUI-64 規格を使用するために強制するためにこのコマンドを入力して下さい:
   ```
   netsh interface ipv6 set privacy state=disabled
   ```
4. 変更を加えるためにマシンをリブートして下さい。

Macintosh OS X

ターミナルでは、次の再度ブートするまでのホストの IPv6 SLAAC をディセーブルにするためにこのコマンドを入力します:

```
sudo sysctl -w net.inet6.ip6.use_tempaddr=0
```

設定パーマネントを作るために、このコマンドを入力して下さい:

```
sudo sh -c 'echo net.inet6.ip6.use_tempaddr=0 >> /etc/sysctl.conf'
```

Linux

ターミナル シェルでは、このコマンドを入力して下さい:

```
sysctl -w net.ipv6.conf.all.use_tempaddr=0
```

ASA からの SLAAC をグローバルにディセーブルにして下さい

第 2 方式はこの動作を当たるために使用する SLAAC の使用を引き起こす、ASA からクライアントに送られる RA メッセージを修正することです。 RA メッセージを修正するために、*インターフェイス設定モード*からこのコマンドを入力して下さい:

```
ASAv(config)# interface gigabitEthernet 1/1
ASAv(config-if)# ipv6 nd prefix 2001::db8/32 300 300 no-autoconfig
```

このコマンドは Aビット フラグが設定 されない、クライアントは一時 IPv6 アドレスを生成しませんように ASA によって送信 される RA メッセージを修正し。

> **ヒント**： その他の情報に関しては RFC 4941 を参照して下さい。

IPv6 FAQ

このセクションは IPv6 の使用に関していくつかの FAQ を記述します。

インターフェイス、同時に両方の IPv4 のためのトラフィックおよび同じの IPv6 を渡すことができますか。

はい。 インターフェイスの IPv6 を単に有効に し、インターフェイスに IPv4 および IPv6 アドレスを両方割り当てて下さい両方のトラフィック の 種類を同時に処理します。

IPv6 を適用でき、同じへの IPv4 ACL はインターフェイスしますか。

ASA バージョンでこれをバージョン 9.0(1)より先にすることができます。 ASA バージョン 9.0(1)現在で、ASA のすべての ACL は*統一されます*、つまり ACL が同じ ACL の IPv4 および IPv6 両方エントリの組合せをサポートすることを意味します。

ASA バージョン 9.0(1) および それ 以降では、ACL は単に一緒にマージされ、単一の、統一された ACL は **access-group コマンド**によってインターフェイスに適用されます。

ASA は IPv6 のための QoS をサポートしますか。

はい。 ASA は IPv4 とすること IPv6 のためのポリシングおよびプライオリティ キューイングを同じようにサポートします。

ASA バージョン 9.0(1)現在で、ASA のすべての ACL は*統一されます*、つまり ACL が同じ ACL の IPv4 および IPv6 両方エントリの組合せをサポートすることを意味します。 その結果、class-map で制定されるどの Qos コマンドでも ACL と一致する IPv4 および IPv6 両方トラフィックの処置をとります。

IPv6 と NAT を使用する必要がありますか。

NAT が ASA の IPv6 のために設定することができるが IPv6 の NAT の使用は非常に落胆させ、不必要、利用可能 な、グローバルにルーティングできる IPv6 アドレスの近い無限量を指定されてです。

NAT が IPv6 シナリオに必要となる場合 *CLI 本 2 の* IPv6 NAT ガイドライン セクションでそれを設定する、方法についての詳細を見つけることができます*: Cisco ASA シリーズ ファイアウォール CLI コンフィギュレーション ガイド、9.4*。

> **注:** IPv6 の NAT を設定するとき考慮する必要がある制限およびいくつかのガイドラインがあります。

show failover コマンド出力のリンク ローカル IPv6 アドレスを見る理由

IPv6 では、ND は L2 アドレス リゾリューションを行うためにリンク ローカル アドレスを使用します。 従って、**show failover コマンド**出力の監視されたインターフェイスのための IPv6 アドレスはインターフェイスで設定されるないグローバル な IPv6 アドレスおよびリンク ローカル アドレスを示します。 これは正常な動作です。

既知の警告/機能拡張要求

いくつかの既知の警告は IPv6 の使用に関してここにあります:

- Cisco バグ ID CSCtn09836 âÂ Â *ASA 8.x キャプチャ「一致する」句は IPv6 トラフィックをつかまえません*
- Cisco バグ ID CSCuq85949 âÂ Â *ENH: WCCP のための ASA IPv6 サポート*
- Cisco バグ ID CSCut78380 âÂ Â *ASA IPv6 ECMP ルーティングはロード バランス トラフィック*

関連情報

- RFC 2460 âÂ Â インターネット プロトコル、バージョン 6 （IPv6）仕様
- RFC 4291 âÂ Â IP バージョン 6（IPv6）アドレッシング アーキテクチャ
- IP バージョン 6(IPv6) のための RFC 4861 âÂ Â Neighbor Discovery
- CLI 本 1: Cisco ASA シリーズ 操作全般 CLI コンフィギュレーション ガイド、9.4 âÂ Â IPv6
- IPv4+IPv6 を介して ASA コンフィギュレーションに対する AnyConnect SSL
- テクニカル サポートとドキュメント - Cisco Systems