Snort IPS

Snort IPS 機能は、Cisco 4000 シリーズサービス統合型ルータおよび Cisco クラウドサービスルータ 1000v シリーズのブランチオフィスで侵入防止システム（IPS）または侵入検知システム（IDS）を実現します。この機能は、オープンソースの Snort ソリューションを使用して IPS と IDS を有効にします。Snort IPS 機能は、Cisco IOS XE リリース 3.16.1S、3.17S、およびそれ以降のリリースで使用できます。

（注） 仮想ルーティングおよび転送（VRF）機能は、Cisco IOS XE Denali リリース16.3.1 以降のリリースの Snort IPS 設定に対応しています。

ここでは、その機能および動作の仕組みについて説明します。

Snort IPS の制約事項

Snort IPS 機能には、次のような制約事項が適用されます。

- Cisco 4000 シリーズ ISR でブーストライセンスを有効にした場合、Snort IPS の仮想サービスコンテナを設定できません。
- ゾーンベース型ファイアウォールの SYN クッキー機能と互換性がありません。

- ネットワークアドレス変換 64（NAT64）には対応しません。
- オープンソースの Snort での SNMP ポーリングには、SnortSnmp プラグインが必要となります。SnortSnmp プラグインが UTD にインストールされていないため、Snort IPS は SNMP ポーリング機能または MIB に対応しません。
- **IOS syslog はレートが制限されているため、Snort によって生成されたすべてのアラートが IOS Syslog で表示されない場合があります。ただし、外部ログサーバにエクスポートする場合は、すべての Syslog メッセージを表示できます。**

Snort IPS に関する情報

Snort IPS の概要

Snort IPS 機能は、Cisco 4000 シリーズサービス統合型ルータおよび Cisco クラウドサービスルータ 1000v シリーズのブランチオフィスで侵入防止システム（IPS）または侵入検知システム（IDS）を実現します。この機能は、Snort エンジンを使用して IPS および IDS 機能を実現します。

Snort は、リアルタイムでトラフィック分析を行い、IP ネットワークで脅威が検出されたときにアラートを生成するオープンソースのネットワーク IPS です。また、プロトコル分析、コンテンツ検索またはマッチングを実行し、バッファオーバーフロー、ステルスポートスキャンなどのさまざまな攻撃やプローブを検出することもできます。Snort エンジンは、Cisco 4000 シリーズサービス統合型ルータおよび Cisco クラウドサービスルータ 1000v シリーズで仮想コンテナサービスとして実行されます。

Snort IPS 機能は、IPS または IDS 機能を提供するネットワーク侵入検知および防止モードで動作します。ネットワーク侵入検知および防止モードでは、Snort は次のアクションを実行します。

- ネットワークトラフィックをモニタし、定義されたルールセットに照らしあわせて分析します。
- 攻撃の分類を行います。
- 一致したルールに照らしあわせてアクションを呼び出します。

要件に応じて、IPS または IDS モードで Snort を有効にできます。IDS モードでは、Snort はトラフィックを検査し、アラートを報告しますが、攻撃を防ぐためのアクションは実行しません。IPS モードでは、侵入検知に加えて、攻撃を防ぐためのアクションを実行します。

Snort IPS はトラフィックをモニタし、イベントを外部ログサーバまたは IOS syslog に報告します。IOS syslog へのロギングを有効にすると、ログメッセージが大量に発生する可能性があるため、パフォーマンスに影響する場合があります。Snort ログに対応する外部のサードパーティ製のモニタリングツールを、ログの収集と分析に使用できます。

Snort IPS 署名パッケージ

UTD OVA は、ルータのセキュリティライセンスに含まれています。デフォルトでは、ルータにはコミュニティ署名パッケージのみがロードされています。サブスクリプションには次の 2 つのタイプがあります。

- コミュニティ署名パッケージ
- サブスクライバベースの署名パッケージ

コミュニティ署名パッケージのルールセットは、脅威に対する限定的な防御を提供します。サブスクライバベースの署名パッケージのルールセットは、脅威に対する最良の防御を提供します。これには、エクスプロイトの前のカバレッジが含まれているため、セキュリティインシデントまたは新しい脅威のプロアクティブな検出に応じて、更新された署名に最速でアクセスできます。このサブスクリプションはシスコによって完全にサポートされており、パッケージは Cisco.com でアップデートされます。サブスクライバベースの署名パッケージは、ソフトウェアのダウンロードページからダウンロードできます。

ユーザがソフトウェアのダウンロードページから署名パッケージを手動でダウンロードする場合、パッケージのバージョンが Snort エンジンのバージョンと同じであることを確認する必要があります。たとえば、Snort エンジンのバージョンが 2982 の場合、ユーザは同じバージョンの署名パッケージをダウンロードする必要があります。バージョンが一致しないと、署名パッケージのアップデートは拒否され、失敗します。

（注） 署名パッケージがアップデートされると、データプレーンのフェールオープンまたはフェールクローズ設定に応じて、エンジンが再起動され、トラフィックが短時間中断されるか、もしくは検知がバイパスされます。

署名更新でサポートされる Cisco IOS XE のリリースおよび UTD パッケージの最小バージョン

次の表 1 に、Cisco IOS XE の最小リリースと、2020 年 1 月以降の署名パッケージのアップデートに対応する各 UTD パッケージのバージョンを示します。表に示されているものより前の Cisco IOS XE のリリースおよび各 UTD パッケージのバージョンには対応していません。表に記載されているものよりも新しい Cisco IOS XE のリリースおよび各 UTD パッケージのバージョンには、最初のリリースから対応しています。

表 *1 : UTD* 署名パッケージのアップデート対応バージョンのマトリックス

Cisco IOS XE リリース	UTD パッケージのバージョン
16.6.7	1.0.10_SV29111_XE_16_6
16.9.4	1.0.4_SV29111_XE_16_9

Cisco IOS XE リリース	UTD パッケージのバージョン
16.10.2	1.0.9_SV2.9.11.1_XE16.10

（注） UTD がオーバーサブスクライブされると、脅威防御チャネルの状態が緑と赤の間で変化します。UTD データプレーンは、フェールクローズが設定されている場合はそれ以降のすべてのパケットをドロップするか、フェールクローズが設定されていない場合は検査されてないパケットを転送します（デフォルト）。UTD サービスプレーンがオーバーサブスクリプションから回復すると、緑色のステータスで UTD データプレーンに応答します。

Snort IPS ソリューション

Snort IPS ソリューションは、次のエンティティで構成されています。

- Snort センサー：トラフィックをモニタして、設定されたセキュリティポリシー（署名、統計情報、プロトコル分析など）に基づいて異常を検出し、アラートサーバまたはレポートサーバにアラートメッセージを送信します。Snort センサーは、仮想コンテナサービスとしてルータに導入されます。
- 署名ストア：定期的に更新される Cisco 署名パッケージをホストします。これらの署名パッケージは、定期的にもしくはオンデマンドで Snort センサーにダウンロードされます。検証済みの署名パッケージは Cisco.com に掲載されます。設定に基づいて、署名パッケージを Cisco.com またはローカルサーバからダウンロードできます。

  次のドメインは、次の cisco.com から署名パッケージをダウンロードするプロセスにおいてルータによってアクセスされます。

  - api.cisco.com
  - apx.cisco.com
  - cloudsso.cisco.com
  - cloudsso-test.cisco.com
  - cloudsso-test3.cisco.com
  - cloudsso-test4.cisco.com
  - cloudsso-test5.cisco.com
  - cloudsso-test6.cisco.com
  - cloudsso.cisco.com
  - download-ssc.cisco.com
  - dl.cisco.com
  - resolver1.opendns.com
  - resolver2.opendns.com

（注） 署名パッケージを保持するためにローカルサーバから署名パッケージをダウンロードする場合は、HTTP のみに対応します。

Snort センサーが署名パッケージを取得するには、Cisco.com の認証情報を使用して、署名パッケージを Cisco.com からローカルサーバに手動でダウンロードする必要があります。

URL が IP アドレスとして指定されていない場合、Snort コンテナは（ルータに設定された DNS サーバ上で）ドメイン名ルックアップを実行して、Cisco.com によるまたはローカルサーバ上の自動署名更新の場所を解決します 。

- アラートまたはレポートサーバ：Snort センサーからアラートイベントを受信します。Snort センサーによって生成されたアラートイベントは、IOS syslog または外部 syslog サーバ、もしくは IOS syslog と外部 syslog サーバの両方に送信できます。Snort IPS ソリューションに付属している外部ログサーバはありません。
- 管理：Snort IPS ソリューションを管理します。管理は、IOS CLI を使用して設定します。Snort センサーには直接アクセスできず、すべての設定は IOS CLI を使用してのみ行えます。

Snort 仮想サービスインターフェイスの概要

Snort センサーは、ルータ上でサービスとして動作します。サービスコンテナは、仮想テクノロジーを使用して、アプリケーション用の Cisco デバイスにホスティング環境を提供します。

Snort トラフィック検査は、インターフェイス単位で、または対応しているすべてのインターフェイスでグローバルに有効にできます。検査対象のトラフィックは Snort センサーに転送され、再度投入されます。侵入検知システム（IDS）では、識別された脅威がログイベントとして報告され、許可されます。ただし、侵入防止システム（IPS）では、ログイベントとともに攻撃を防ぐためのアクションが実行されます。

Snort センサーには 2 つの VirtualPortGroup インターフェイスが必要です。最初の VirtualPortGroup インターフェイスは管理トラフィックに使用され、2 つ目は転送プレーンと Snort 仮想コンテナサービス間のデータトラフィックに使用されます。これらの VirtualPortGroup インターフェイスには、ゲスト IP アドレスを設定する必要があります。管理 VirtualPortGroup インターフェイスに割り当てられた IP サブネットは、署名サーバおよびアラート / 報告サーバと通信できる必要があります。

2 つ目の VirtualPortGroup インターフェイスの IP サブネットは、このインターフェイス上のトラフィックがルータ内部にあるため、カスタマーネットワーク上でルーティング可能であってはなりません。内部サブネットを外部に公開することはセキュリティ上のリスクとなります。2 つ目の VirtualPortGroup サブネットには 192.0.2.0/30 の IP アドレス範囲を使用することをお勧めします。192.0.2.0/24 のサブネットを使用することは、RFC 3330 で定義されています。

管理トラフィック用の **virtual-service** コマンドを使って管理インターフェイスを使用することもできます。管理インターフェイスを設定する場合、2 つの VirtualPortGroup インターフェイ

スが必要となります。ただし、最初の VirtualPortGroup インターフェイスには **guest ip address** を設定しないでください。

仮想サービスが実行されているルータと同じ管理ネットワークで、Snort 仮想コンテナサービスの IP アドレスを割り当てることができます。この設定は、syslog またはアップデートサーバが管理ネットワーク上にあり、他のインターフェイスからアクセスできない場合に役立ちます。

仮想サービスのリソースプロファイル

Snort IPS 仮想サービスは、低、中、高という 3 つのリソースプロファイルに対応しています。これらのプロファイルは、仮想サービスの実行に必要な CPU およびメモリリソースを表示します。これらのリソースプロファイルの 1 つを設定できます。リソースプロファイルの設定は任意です。プロファイルを設定しない場合、仮想サービスはデフォルトのリソースプロファイルでアクティブ化されます。次の表に、Cisco 4000 シリーズ ISR および Cisco クラウドサービスルータ 1000v シリーズのリソースプロファイルの詳細を示します。

プラットフォーム	プロファイル	仮想サービスのリソース要件		プラットフォーム要件
		システム CPU	メモリ	
Cisco 4321 ISR	デフォルト	50%	最小：1 GB（RAM） 最小：750 MB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
Cisco 4331 ISR	低（デフォルト）	25%	最小：1 GB（RAM） 最小：750 MB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	中	50%	最小：2 GB（RAM） 最小：1 GB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	高	75%	最小：4 GB（RAM） 最小：2 GB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）

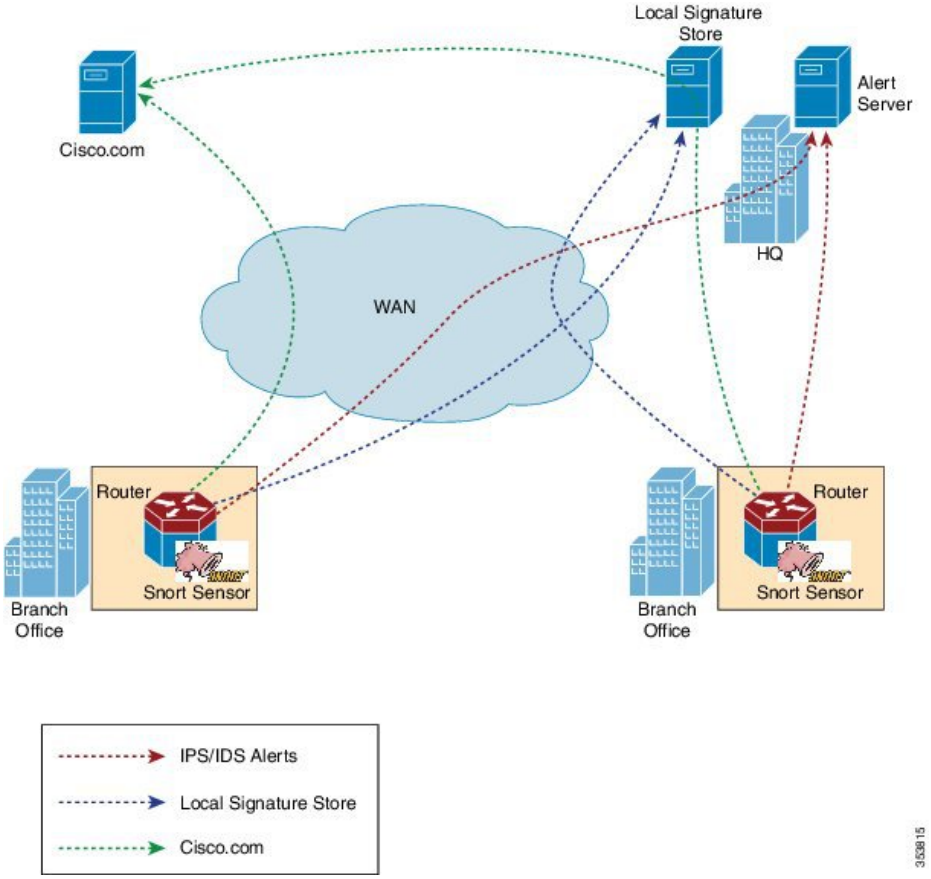
プラットフォーム	プロファイル	仮想サービスのリソース要件		プラットフォーム要件
		システム CPU	メモリ	
Cisco 4351 ISR	低（デフォルト）	25%	最小：1 GB（RAM） 最小：750 MB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	中	50%	最小：2 GB（RAM） 最小：1 GB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	高	75%	最小：4 GB（RAM） 最小：2 GB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
Cisco 4431 ISR	低（デフォルト）	25%	最小：1 GB（RAM） 最小：750 MB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	中	50%	最小：2 GB（RAM） 最小：1 GB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	高	75%	最小：4 GB（RAM） 最小：2 GB（ディスクまたはフラッシュ）	最小：12 GB（RAM） 最小：12 GB（ディスクまたはフラッシュ）

プラットフォーム	プロファイル	仮想サービスのリソース要件		プラットフォーム要件
		システム CPU	メモリ	
Cisco 4451 ISR	低（デフォルト）	25%	最小：1 GB（RAM） 最小：750 MB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	中	50%	最小：2 GB（RAM） 最小：1 GB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	高	75%	最小：4 GB（RAM） 最小：2 GB（ディスクまたはフラッシュ）	最小：12 GB（RAM） 最小：12 GB（ディスクまたはフラッシュ）
Cisco CSR 1000V	低（デフォルト）	25%	最小：1 GB（RAM） 最小：750 MB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	中	50%	最小：2 GB（RAM） 最小：1 GB（ディスクまたはフラッシュ）	最小：8 GB（RAM） 最小：8 GB（ディスクまたはフラッシュ）
	高	75%	最小：3 GB（RAM） 最小：2 GB（ディスクまたはフラッシュ）	最小：12 GB（RAM） 最小：12 GB（ディスクまたはフラッシュ）

Snort IPS の導入

次の図は、Snort IPS の導入概要を示しています。

図 *1 : Snort IPS* の展開概要



次の手順では、Snort IPS ソリューションの導入について説明します。

- Snort OVA ファイルを Cisco ルータにコピー、インストール、アクティブ化する。
- 署名パッケージを、Cisco.com または設定済みのローカルサーバから Cisco ルータにダウンロードする。
- ネットワーク侵入検知またはネットワーク防御機能を設定する。
- アラートおよびレポートサーバを、Snort センサーからアラートを受信するように設定する。

Snort IPS の導入方法

対応しているデバイスに Snort IPS を導入するには、次のタスクを実行します。

1. デバイスをプロビジョニングします。

   Snort IPS 機能をインストールするデバイスを特定します。

2. ライセンスを取得します。

Snort IPS 機能は、サービスを有効にするためにセキュリティライセンスを必要とするセキュリティパッケージでのみ使用できます。この機能は、Cisco IOS XE リリース 3.16.1S、3.17S、およびそれ以降のリリースで使用できます。

（注） ライセンスの取得については、シスコ サポートにお問い合わせください。

3. Snort OVA ファイルをインストールします。
4. VirtualPortGroup のインターフェイスおよび仮想サービスを設定します。
5. Snort 仮想コンテナサービスをアクティブにします。
6. Snort IPS または IDS のモードとポリシーを設定します。
7. 外部アラートおよびログサーバまたは IOS syslog、あるいはその両方へのイベントのレポートを設定します。
8. 署名の更新方法を設定します。
9. 署名を更新します。
10. IPS をグローバルに、または必要なインターフェイスで有効にします。

Snort OVA ファイルのインストール

OVA ファイルは、仮想マシンの圧縮された「インストール可能な」バージョンを含むオープン仮想アーカイブ（Open Virtualization Archive）です。Snort IPS は仮想コンテナサービスとして使用できます。この OVA ファイルをルータにダウンロードし、**virtual-service install** CLI を使用してサービスをインストールする必要があります。

サービス OVA ファイルは、ルータにインストールされている Cisco IOS XE リリースイメージには付属していません。ただし、OVA ファイルはルータのフラッシュに事前にインストールされている場合があります。

セキュリティライセンスが付属した Cisco IOS XE イメージを使用する必要があります。OVA ファイルのインストール中に、セキュリティライセンスがチェックされ、ライセンスが存在しない場合はエラーが報告されます。

手順の概要

1. **enable**
2. **virtual-service install name** *virtual-service-name* **package** *file-url* **media** *file-system*
3. **show virtual-service list**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**virtual-service install name** *virtual-service-name* **package** *file-url* **media** *file-system* 例： `Device# virtual-service install name UTDIPS package harddisk:utd-ips-v102.ova media harddisk:`	デバイスの仮想サービスコンテナにアプリケーションをインストールします。 • 名前の長さは 20 文字です。ハイフン（-）は有効な文字ではありません。 • インストールする OVA パッケージの完全パスを指定する必要があります。 （注） OVA のインストールは、ハードディスクとブートフラッシュの両方で行えますが、OVA をインストールするのに推奨されるファイルシステムはハードディスクです。
ステップ 3	**show virtual-service list** 例： `Device# show virtual-service list`	仮想サービスコンテナにインストールされているすべてのアプリケーションのインストールのステータスを表示します。

VirtualPortGroup のインターフェイスおよび仮想サービスの設定

2 つの VirtualPortGroup インターフェイスと両方のインターフェイスのゲスト IP アドレスを設定する必要があります。ただし、 **vnic management GigabitEthernet0** コマンドを使用して管理インターフェイスを設定する場合は、最初の VirtualPortGroup インターフェイスのゲスト IP アドレスを設定しないでください。

（注） データトラフィック用の VirtualPortGroup インターフェイスは、プライベートまたはルーティング不可の IP アドレスを使用する必要があります。このインターフェイスには、IP アドレスの範囲として 192.0.2.0 / 30を 使用することを推奨します。

（注） Cisco IOS ソフトウェアイメージを XE 3.x バージョンから XE 16.2.1 に、または XE 16.2.1 から XE 3.x バージョンに変更する前に、デバイス上の仮想サービスごとに **virtual-service uninstall name [name]** コマンドを使用して仮想サービスをアンインストールします。仮想サービスの 1 つが ISR-WAAS サービスであり、**service waas enable** コマンドを使用してインストールされている場合は、**service waas disable** コマンドを使用します。

Cisco IOS ソフトウェアイメージの新しいバージョンでデバイスをアップグレードした後、仮想サービスを再インストールします。ISR-WAAS の場合は **service wass enable** コマンドを使用し、その他の仮想サービスの場合は **virtual-service install name [name] package [.ova file]** コマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *VirtualPortGroup number*
4. **ip address** *ip-address mask*
5. **exit**
6. **interface** *type number*
7. **ip address** *ip-address mask*
8. **exit**
9. **virtual-service** *name*
10. **profile** *profile-name*
11. **vnic gateway VirtualPortGroup** *interface-number*
12. **guest ip address** *ip-address*
13. **exit**
14. **vnic gateway VirtualPortGroup** *interface-number*
15. **guest ip address** *ip-address*
16. **exit**
17. **vnic management GigabitEthernet0**
18. **guest ip address** *ip-address*
19. **exit**
20. **activate**
21. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *VirtualPortGroup number* 例： `Device(config)# interface VirtualPortGroup 0`	インターフェイスを設定し、インターフェイス設定モードを開始します。 • VirtualPortGroup インターフェイスを設定します。このインターフェイスは、管理インターフェイスの GigabitEthernet0 が使用されていない場合に管理トラフィックに対して使用されます。
ステップ 4	**ip address** *ip-address mask* 例：	インターフェイスのプライマリ IP アドレスを設定します。このインターフェイスは、署名アップデー

	コマンドまたはアクション	目的
	`Device(config-if)# ip address 10.1.1.1 255.255.255.252`	トサーバおよび外部ログサーバにルーティング可能である必要があります。
ステップ 5	**exit** 例： `Device(config-if)# exit`	インターフェイス設定モードを終了し、グローバル設定モードに戻ります。
ステップ 6	**interface** *type number* 例： `Device(config)# interface VirtualPortGroup 1`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。 • VirtualPortGroup インターフェイスを設定します。 • このインターフェイスは、データトラフィックに使用されます。
ステップ 7	**ip address** *ip-address mask* 例： `Device(config-if)# ip address 192.0.2.1 255.255.255.252`	インターフェイスのプライマリ IP アドレスを設定します。 • この IP アドレスは、外部ネットワークに対してルーティング不能である必要があります。 • IP アドレスは、推奨される 192.0.2.0/30 のサブネットから割り当てられます。
ステップ 8	**exit** 例： `Device(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了し、グローバルコンフィギュレーションモードに戻ります。
ステップ 9	**virtual-service** *name* 例： `Device(config)# virtual-service UTDIPS`	仮想コンテナサービスを設定し、仮想サービス設定モードに入ります。 • *name* 引数は、仮想コンテナサービスを識別するために使用される論理名です。

ステップ 10	**profile** *profile-name*
	例：
	`Device(config-virt-serv)#profile high`
	例：
	`Device(config-virt-serv)#profile multi-tenancy`
		（オプション）リソースプロファイルを設定します。リソースプロファイルを設定しない場合、仮想サービスはデフォルトのリソースプロファイルを使用してアクティブ化されます。オプションは、low、medium、high、および multi-tenancy です。（マルチテナントモードの場合（Cisco CSR 1000v のみ）、`profile multi-tenancy` コマンドを設定する必要があります。

ステップ 11	**vnic gateway VirtualPortGroup** *interface-number* 例： `Device(config-virt-serv)# vnic gateway VirtualPortGroup 0`	仮想コンテナサービスの仮想ネットワーク インターフェイス カード（vNIC）のゲートウェイ インターフェイスを作成し、vNIC ゲートウェイ インターフェイスを仮想ポートグループにマッピングして、仮想サービスの vNIC 設定モードに入ります。

	コマンドまたはアクション	目的
		• このコマンドで参照されるインターフェイスは、手順3で設定したインターフェイスである必要があります。このコマンドは、管理目的で使用されるインターフェイスをマッピングします。
ステップ 12	**guest ip address** *ip-address* 例： `Device(config-virt-serv-vnic)# guest ip address 10.1.1.2`	（オプション）vNIC ゲートウェイ インターフェイスのゲスト vNIC アドレスを設定します。 • （注） 手順 17 で指定した **vnic management gigabitethernet0** コマンドが設定されていない場合にのみこのコマンドを設定します。
ステップ 13	**exit** 例： `Device(config-virt-serv-vnic)# exit`	仮想サービスの vNIC 設定モードを終了し、仮想サービス設定モードに戻ります。
ステップ 14	**vnic gateway VirtualPortGroup** *interface-number* 例： `Device(config-virt-serv)# vnic gateway VirtualPortGroup 1`	仮想コンテナサービスの vNIC ゲートウェイ インターフェイスを作成し、vNIC ゲートウェイ インターフェイスを仮想ポートグループにマッピングして、仮想サービスの vNIC 設定モードに入ります。 • このコマンドで参照されるインターフェイスは、手順6で設定したインターフェイスである必要があります。このコマンドは、Snortがユーザトラフィックのモニタリングに使用する仮想コンテナサービスのインターフェイスをマッピングします。
ステップ 15	**guest ip address** *ip-address* 例： `Device(config-virt-serv-vnic)# guest ip address 192.0.2.2`	vNIC ゲートウェイ インターフェイスのゲスト vNIC アドレスを設定します。
ステップ 16	**exit** 例： `Device(config-virt-serv-vnic)# exit`	仮想サービスの vNIC 設定モードを終了し、仮想サービス設定モードに戻ります。
ステップ 17	**vnic management GigabitEthernet0** 例： `Device(config-virt-serv)# vnic management GigabitEthernet0`	（オプション）GigabitEthernet インターフェイスを vNIC 管理インターフェイスとして設定します。 • 管理インターフェイスは、VirtualPortGroup インターフェイスまたはGibugitEthernet0インターフェイスである必要があります。

	コマンドまたはアクション	目的
		• **vnic management GigabitEthernet0** コマンドを設定しない場合は、手順 12 で指定した **guest ip address** コマンドを設定する必要があります。
ステップ 18	**guest ip address** *ip-address* 例： `Device(config-virt-serv-vnic)# guest ip address 209.165.201.1`	（オプション）vNIC 管理インターフェイスのゲスト vNIC アドレスを設定します。このアドレスは、管理インターフェイスおよび GigabitEthernet0 設定と同じサブネット内にある必要があります。
ステップ 19	**exit** 例： `Device(config-virt-serv-vnic)# exit`	仮想サービスの vNIC 設定モードを終了し、仮想サービス設定モードに戻ります。
ステップ 20	**activate** 例： `Device(config-virt-serv)# activate`	仮想コンテナサービスにインストールされたアプリケーションをアクティブにします。
ステップ 21	**end** 例： `Device(config-virt-serv)# end`	仮想サービス設定モードを終了し、特権 EXEC モードに戻ります。

Snort IPS のグローバル設定

要件に基づいて、侵入防止システム（IPS）または侵入検知システム（IDS）の検査をグローバルレベルまたはインターフェイスで設定します。このタスクを実行して、デバイス上で IPS をグローバルに設定します。

（注） グローバルという用語は、対応しているすべてのインターフェイスで実行されている Snort IPS を意味します。

手順の概要

1. **enable**
2. **configure terminal**
3. **utd threat-inspection whitelist**
4. **generator id** *generator-id* **signature id** *signature-id* [**comment** *description*]
5. **exit**
6. **utd engine standard**
7. **logging** {**host** *hostname* | **syslog**}
8. **threat-inspection**

9. **threat** {**detection** | **protection** }
10. **policy** {**balanced** | **connectivity** | **security**}
11. **whitelist**
12. **signature update occur-at** {**daily** | **monthly** *day-of-month* | **weekly** *day-of-week*} *hour minute*
13. **signature update server** {**cisco** | **url** *url* } [**username** *username* [**password** *password*]]
14. **logging level** {**alert** | **crit** | **debug** | **emerg** | **err** | **info** | **notice** | **warning**}
15. **exit**
16. **utd**
17. **redirect interface virtualPortGroup** *interface-number*
18. **all-interfaces**
19. **engine standard**
20. **fail close**
21. **exit**
22. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル設定モードを開始します。
ステップ 3	**utd threat-inspection whitelist** 例： `Device(config)# utd threat-inspection whitelist`	（オプション）UTD 許可リストの設定モードを有効にします。
ステップ 4	**generator id** *generator-id* **signature id** *signature-id* [**comment** *description*] 例： `Device(config-utd-whitelist)# generator id 24 signature id 24245 comment traffic from branchoffice1`	署名 ID を許可リストに表示するように設定します。 • 署名 IDは、抑制する必要があるアラートからコピーできます。 • 複数の署名 ID を設定できます。 • 許可リストに追加する必要がある署名 ID ごとに、この手順を繰り返します。
ステップ 5	**exit** 例： `Device(config-utd-whitelist)# exit`	UTD 許可リストの設定モードを終了して、グローバル設定モードに戻ります。

	コマンドまたはアクション	目的
ステップ 6	**utd engine standard** 例： `Device(config)# utd engine standard`	統合脅威防御（UTD）標準エンジンを設定し、UTD 標準エンジンの設定モードに入ります。
ステップ 7	**logging** {**host** *hostname* | **syslog**} 例： `Device(config-utd-eng-std)# logging host syslog.yourcompany.com`	サーバへの緊急メッセージのロギングを有効にします。
ステップ 8	**threat-inspection** 例： `Device(config-utd-eng-std)# threat-inspection`	Snort エンジンの脅威検知を設定します。
ステップ 9	**threat** {**detection** | **protection** } 例： `Device(config-utd-eng-std-insp)# threat protection`	脅威検知または侵入防止システム（IPS）を Snort エンジンの動作モードとして設定します。 • デフォルトは**detection**です。 • 侵入検知システム（IDS）を設定するには、**detection** キーワードを設定します。
ステップ 10	**policy** {**balanced** | **connectivity** | **security**} 例： `Device(config-utd-eng-std-insp)# policy security`	Snort エンジンのセキュリティポリシーを設定します。 • デフォルトのポリシーオプションは **balanced** です。
ステップ 11	**whitelist** 例： `Device(config-utd-eng-std-insp)# whitelist`	（オプション）UTD エンジンで許可リストを有効にします。
ステップ 12	**signature update occur-at** {**daily** | **monthly** *day-of-month* | **weekly** *day-of-week*} *hour minute* 例： `Device(config-utd-eng-std-insp)# signature update occur-at daily 0 0`	署名の更新間隔パラメータを設定します。この設定をすることで、午前0時に署名の更新がトリガーされます。
ステップ 13	**signature update server** {**cisco** | **url** *url* } [**username** *username* [**password** *password*]] 例： `Device(config-utd-eng-std-insp)# signature update server cisco username abcd password cisco123`	署名更新サーバのパラメータを設定します。サーバの詳細で署名更新パラメータを指定する必要があります。署名の更新にCisco.comを使用する場合は、ユーザ名とパスワードを入力する必要があります。署名の更新にローカルサーバを使用する場合は、サーバ設定に基づいてユーザ名とパスワードを指定できます。

	コマンドまたはアクション	目的
ステップ 14	**logging level** {**alert** | **crit** | **debug** | **emerg** | **err** | **info** | **notice** | **warning**} 例： `Device(config-utd-eng-std-insp)# logging level emerg`	ログレベルを有効にします。
ステップ 15	**exit** 例： `Device(config-utd-eng-std-insp)# exit`	UTD標準エンジンの設定モードを終了して、グローバル設定モードに戻ります。
ステップ 16	**utd** 例： `Device(config)# utd`	統合脅威防御（UTD）を有効にし、UTD 設定モードに入ります。
ステップ 17	**redirect interface virtualPortGroup** *interface-number* 例： `Device(config-utd)# redirect interface virtualPortGroup 1`	（オプション）VirtualPortGroup インターフェイスにリダイレクトします。これはデータ トラフィック インターフェイスです。このインターフェイスを設定しない場合、インターフェイスは自動検出されます。
ステップ 18	**all-interfaces** 例： `Device(config-utd)# all-interfaces`	デバイスのすべてのレイヤ 3 インターフェイスで UTD を設定します。
ステップ 19	**engine standard** 例： `Device(config-utd)# engine standard`	統合脅威防御（UTD）エンジンを設定し、標準エンジンの設定モードに入ります。
ステップ 20	**fail close** 例： `Device(config-engine-std)# fail close`	（オプション）UTD エンジンに障害が発生した場合に行うアクションを定義します。デフォルトのオプションはフェールオープンです。フェールクローズオプションは、UTD エンジンに障害が発生した場合にすべての IPS または IDS トラフィックをドロップします。フェールオープンオプションを使用すると、UTD エンジンに障害が発生した場合にすべての IPS または IDS トラフィックを許可します。
ステップ 21	**exit** 例： `Device(config-eng-std)# exit`	標準エンジンの設定モードを終了して、グローバル設定モードに戻ります。
ステップ 22	**end** 例： `Device(config-utd)# end`	UTD 設定モードを終了して、グローバル設定モードに戻ります。

Snort IDS 検知のグローバル設定

要件に基づいて、侵入防止システム（IPS）または侵入検知システム（IDS）検査をグローバルレベルまたはインターフェイスレベルで設定します。インターフェイスごとにIDSを設定するには、次のタスクを実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **utd enable**
5. **exit**
6. 検査が必要なすべてのインターフェイスで、手順 3 ー 5 を繰り返します。
7. **utd threat-inspection whitelist**
8. **generator id** *generator-id* **signature id** *signature-id* [**comment** *description*]
9. **exit**
10. **utd engine standard**
11. **logging** {**host** *hostname* | **syslog**}
12. **threat-inspection**
13. **threat** {**detection** | **protection** }
14. **policy** {**balanced** | **connectivity** | **security**}
15. **whitelist**
16. **signature update occur-at** {**daily** | **monthly** *day-of-month* | **weekly** *day-of-week*} *hour minute*
17. **signature update server** {**cisco** | **url** *url*} [**username** *username* [**password** *password*]]
18. **logging level** {**alert** | **crit** | **debug** | **emerg** | **err** | **info** | **notice** | **warning**}
19. **exit**
20. **utd**
21. **redirect interface virtualPortGroup** *interface-number*
22. **engine standard**
23. **fail close**
24. **exit**
25. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**interface** *type number* 例： `Device(config)# interface gigabitethernet 0/0/0`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**utd enable** 例： `Device(config-if)# utd enable`	統合脅威防御（UTD）を有効にします。
ステップ 5	**exit** 例： `Device(config-if)# exit`	インターフェイス設定モードを終了し、グローバル設定モードに戻ります。
ステップ 6	検査が必要なすべてのインターフェイスで、手順 3 ー 5 を繰り返します。	–
ステップ 7	**utd threat-inspection whitelist** 例： `Device(config)# utd threat-inspection whitelist`	（オプション）UTD 許可リストの設定モードを有効にします。
ステップ 8	**generator id** *generator-id* **signature id** *signature-id* [**comment** *description*] 例： `Device(config-utd-whitelist)# generator id 24 signature id 24245 comment traffic from branchoffice1`	署名 ID を許可リストで表示するように設定します。 • 署名 IDは、抑制する必要があるアラートからコピーできます。 • 複数の署名 ID を設定できます。 • 許可リストに表示する必要がある署名 ID ごとに、この手順を繰り返します。
ステップ 9	**exit** 例： `Device(config-utd-whitelist)# exit`	UTD 許可リストの設定モードを終了して、グローバル設定モードに戻ります。
ステップ 10	**utd engine standard** 例： `Device(config)# utd engine standard`	統合脅威防御（UTD）標準エンジンを設定し、UTD 標準エンジンの設定モードに入ります。
ステップ 11	**logging** {**host** *hostname* | **syslog**} 例： `Device(config-utd-eng-std)# logging syslog`	IOSd syslog への重要なメッセージのロギングを有効にします。
ステップ 12	**threat-inspection** 例： `Device(config-utd-eng-std)# threat-inspection`	Snort エンジンの脅威検知を設定します。

	コマンドまたはアクション	目的
ステップ 13	**threat** {**detection** | **protection** } 例： `Device(config-utd-eng-std-insp)# threat detection`	脅威防止または侵入検知システム（IDS）を Snort センサーの動作モードとして設定します。 • 侵入防止システム（IPS）を設定するには、**protection** キーワードを設定します。
ステップ 14	**policy** {**balanced** | **connectivity** | **security**} 例： `Device(config-utd-eng-std-insp)# policy balanced`	Snort センサーのセキュリティポリシーを設定します。
ステップ 15	**whitelist** 例： `Device(config-utd-eng-std-insp)# whitelist`	（オプション）トラフィックの許可リストを有効にします。
ステップ 16	**signature update occur-at** {**daily** | **monthly** *day-of-month* | **weekly** *day-of-week*} *hour minute* 例： `Device(config-utd-eng-std-insp)# signature update occur-at daily 0 0`	署名の更新間隔パラメータを設定します。この設定をすることで、午前0時に署名の更新がトリガーされます。
ステップ 17	**signature update server** {**cisco** | **url** *url*} [**username** *username* [**password** *password*]] 例： `Device(config-utd-eng-std-insp)# signature update server cisco username abcd password cisco123`	署名更新サーバのパラメータを設定します。サーバの詳細で署名更新パラメータを指定する必要があります。署名の更新にCisco.comを使用する場合は、ユーザ名とパスワードを入力する必要があります。署名の更新にローカルサーバを使用する場合は、サーバ設定に基づいてユーザ名とパスワードを指定できます。
ステップ 18	**logging level** {**alert** | **crit** | **debug** | **emerg** | **err** | **info** | **notice** | **warning**} 例： `Device(config-utd-eng-std-insp)# logging level crit`	ログレベルを有効にします。
ステップ 19	**exit** 例： `Device(config-utd-eng-std-insp)# exit`	UTD標準エンジンの設定モードを終了して、グローバル設定モードに戻ります。
ステップ 20	**utd** 例： `Device(config)# utd`	統合脅威防御（UTD）を有効にし、UTD 設定モードに入ります。
ステップ 21	**redirect interface virtualPortGroup** *interface-number* 例：	（オプション）VirtualPortGroup インターフェイスにリダイレクトします。これはデータ トラフィック インターフェイスです。このインターフェイス

	コマンドまたはアクション	目的
	`Device(config-utd)# redirect interface virtualPortGroup 1`	を設定しない場合、インターフェイスは自動検出されます。
ステップ 22	**engine standard** 例： `Device(config-utd)# engine standard`	統合脅威防御（UTD）エンジンを設定し、標準エンジンの設定モードに入ります。
ステップ 23	**fail close** 例： `Device(config-engine-std)# fail close`	（オプション）UTD エンジンに障害が発生した場合に行うアクションを定義します。デフォルトのオプションはフェールオープンです。フェールクローズオプションは、UTD エンジンに障害が発生した場合にすべての IPS または IDS トラフィックをドロップします。フェールオープンオプションを使用すると、UTD エンジンに障害が発生した場合にすべての IPS または IDS トラフィックを許可します。
ステップ 24	**exit** 例： `Device(config-eng-std)# exit`	標準エンジンの設定モードを終了して、グローバル設定モードに戻ります。
ステップ 25	**end** 例： `Device(config-utd)# end`	設定モードを終了し、EXEC モードに戻ります。

アクティブな署名のリストの表示

アクティブな署名は、Snort IDS または IPS に脅威に対するアクションを実行するように指示するものです。トラフィックがアクティブな署名のいずれかと一致した場合、Snort コンテナは IDS モードでアラートをトリガーし、IPS モードでトラフィックをドロップします。

utd threat-inspection signature active-list write-to bootflash: file name コマンドは、アクティブな署名のリストと、アクティブな署名、ドロップ署名、およびアラート署名の合計数のサマリーを表示します。

コンテナの正常性をモニタリングするための Quality of Service（QoS）ポリシーの設定

コンテナの正常性をモニタリングする正常性プローブが高いトラフィックレートで影響を受けないように、Quality of Service（QoS）ポリシーを設定することをお勧めします。

手順の概要

1. **ip access-list extended** {acl-name | acl-number}

2. sequence-number permit protocol source *source-wildcard destination destination-wildcard* [precedence] [tos *tos* tos] [log] [time-range*time-range-name*] [fragments]
3. **exit**
4. class-map { [type inspect match-all] | [match-any] } *class-map-name*
5. match access-group { *access-group* | name *access-group-name*}
6. **exit**
7. policy-map *policy-map-name*
8. class {*class-name* | class-default
9. priority level *level*
10. **exit**
11. **interface** *type number*
12. service-policy [history | {output} *policy-map-name* | type control *control-policy-name*]
13. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**ip access-list extended** {acl-name | acl-number} 例： `Device(config)# ip access-list extended health_probes_accesslist`	拡張 ACL コンフィギュレーション モードを有効にします。CLI は拡張 ACL コンフィギュレーション モードを開始します。このモードでは、後続のすべてのコマンドが現在の拡張アクセスリストに適用されます。
ステップ 2	sequence-number permit protocol source *source-wildcard destination destination-wildcard* [precedence] [tos *tos* tos] [log] [time-range*time-range-name*] [fragments] 例： `Device(config-ext-nacl)# 10 permit udp any eq 3367 any eq 3367`	名前付き IP アクセス リスト モードで permit ステートメントを指定します。このアクセスリストでは、**permit** ステートメントを最初に使用していますが、必要なステートメントの順序に応じて、**deny** ステートメントが最初に使用される可能性もあります。
ステップ 3	**exit** 例： `Device(config-ext-nacl)# exit`	拡張 ACL コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 4	class-map { [type inspect match-all] | [match-any] } *class-map-name* 例： `Device(config)# class-map match-all health_probes_cmap`	作成するクラス マップの名前を指定し、QoS クラス マップ コンフィギュレーション モードを開始します。
ステップ 5	match access-group { *access-group* | name *access-group-name*} 例： `Device(config-cmap)# match access-group name health_probes_accesslist`	すべてのパケットに対して適切に一致する基準となる、クラスマップの一致基準を設定します。

	コマンドまたはアクション	目的
ステップ 6	**exit** 例： `Device(config-cmap)# exit`	クラス マップ コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 7	policy-map *policy-map-name* 例： `Device(config)# policy-map health_probes_pmap`	サービス ポリシーを指定するために 1 つ以上のインターフェイスに適用可能なポリシー マップを作成または修正し、QoS ポリシーマップ コンフィギュレーション モードを開始します。
ステップ 8	class {*class-name* | class-default 例： `Device(config-pmap)# class health_probes_cmap`	クラスのポリシーを設定する前に、ポリシーの作成/変更対象となるクラスの名前を指定するか、（一般に class-default クラスと呼ばれる）デフォルト クラスを指定してから、ポリシーマップ コンフィギュレーション モードを開始します。
ステップ 9	priority level *level* 例： `Device(config-pmap)# priority level 1`	指定されたプライオリティ レベルでトラフィック クラスにプライオリティを割り当てます。 • プライオリティ クラスに割り当てられた優先順位の値を入力します。有効な値は、1（高優先順位）または2（低優先順位）です。デフォルトは 1 です。
ステップ 10	**exit** 例： `Device(config-pmap)# exit`	ポリシーマップ コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 11	**interface** *type number* 例： `Device(config)# interface VirtualPortGroup 1`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。 • VirtualPortGroup インターフェイスを設定します。 • このインターフェイスは、データトラフィックに使用されます。
ステップ 12	service-policy [history | {output} *policy-map-name* | type control *control-policy-name*] 例： `Device(config-if)# service-policy output health_probes_pmap`	ポリシー マップをクラスに結合します。適用されるサービス ポリシー マップ（**policy-map** コマンドを使用して作成）の名前。名前には最大 40 文字までの英数字を指定できます。
ステップ 13	**end** 例： `Device(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

Snort IPS の設定例

例：VirtualPortGroup インターフェイスおよび仮想サービスの設定

```
Device# configure terminal
Device(config)# interface VirtualPortGroup 0
Device(config-if)# ip address 10.1.1.1 255.255.255.252
Device(config-if)# exit
Device(config)# interface VirtualPortGroup 1
Device(config-if)# ip address 192.0.2.1 255.255.255.252
Device(config-if)# exit
Device(config)# virtual-service UTDIPS
Device(config-virt-serv)# vnic gateway VirtualPortGroup 0
Device(config-virt-serv-vnic)# exit
Device(config-virt-serv)# vnic gateway VirtualPortGroup 1
Device(config-virt-serv-vnic)# guest ip address 192.0.2.2
Device(config-virt-serv-vnic)# exit
Device(config-virt-serv)# vnic management GigabitEthernet0
Device(config-virt-serv-vnic)# guest ip address 209.165.201.1
Device(config-virt-serv-vnic)# exit
Device(config-virt-serv)# activate
Device(config-virt-serv-vnic)# end
```

例：異なるリソースプロファイルの設定

```
Device# configure terminal
Device(config)# virtual-service UTDIPS
Device(config-virt-serv)# no activate
Device(config-virt-serv)# end
Device# virtual-service uninstall name UTDIPS
Device# configure terminal
Device(config)# virtual-service UTDIPS
Device(config-virt-serv)# profile medium
Device(config-virt-serv)# end
Device# virtual-service install name UTDIPS package:utd.ova
Device# configure terminal
Device(config)# virtual-service UTDIPS
Device(config-virt-serv)# activate
Device(config-virt-serv)# end
```

例：Snort IPS のグローバル設定

次に、デバイス上で侵入防止システム（IPS）をグローバルに設定する例を示します。

```
Device# configure terminal
Device(config)# utd engine standard
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-eng-std-insp)# threat protection
Device(config-utd-eng-std-insp)# policy security
```

```
Device(config-utd-eng-std)# exit
Device(config)# utd
Device(config-utd)# all-interfaces
Device(config-utd)# engine standard
Device(config-utd-whitelist)# end
Device#
```

例：インターフェイスごとの Snort IPS 検査の設定

次に、インターフェイスごとに Snort 侵入検知システム（IDS）を設定する例を示します。

```
Device# configure terminal
Device(config)# utd engine standard
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-eng-std-insp)# threat detection
Device(config-utd-eng-std-insp)# policy security
Device(config-utd-eng-std)# exit
Device(config)# utd
Device(config-utd)# engine standard
Device(config-eng-std)# exit
Device(config)# interface gigabitethernet 0/0/0
Device(config-if)# utd enable
Device(config-if)# exit
```

例：インバウンドインターフェイスとアウトバウンドインターフェイスの両方での VRF を使用した UTD の設定

```
Device# configure terminal
Device(config)# vrf definition VRF1
Device(config-vrf)# rd 100:1
Device(config-vrf)# route-target export 100:1
Device(config-vrf)# route-target import 100:1
Device(config-vrf)# route-target import 100:2
!
Device(config-vrf)# address-family ipv4
Device(config-vrf-af)# exit
!
Device(config-vrf)# address-family ipv6
Device(config-vrf-af)# exit
!
Device(config-vrf-af)# vrf definition VRF2
Device(config-vrf)# rd 100:2
Device(config-vrf)# route-target export 100:2
Device(config-vrf)# route-target import 100:2
Device(config-vrf)# route-target import 100:1
!
Device(config-vrf)# address-family ipv4
Device(config-vrf-af)# exit
!
Device(config-vrf)# address-family ipv6
Device(config-vrf-af)# exit
!
Device(config-vrf)# interface VirtualPortGroup0
Device(config-if)# ip address 192.0.2.1 255.255.255.252
```

```
Device(config-if)# no mop enabled
Device(config-if)# no mop sysid
!
Device(config-if)# interface VirtualPortGroup1
Device(config-if)# ip address 192.0.2.5 255.255.255.252
Device(config-if)# no mop enabled
Device(config-if)# no mop sysid
!
Device(config-if)# interface GigabitEthernet0/0/2
Device(config-if)# vrf forwarding VRF1
Device(config-if-vrf)# ip address 192.1.1.5 255.255.255.0
Device(config-if-vrf)# ipv6 address A000::1/64
!
Device(config-if)# interface GigabitEthernet0/0/3
Device(config-if)# vrf forwarding VRF2
Device(config-if-vrf)# ip address 192.1.1.5 255.255.255.0
Device(config-if-vrf)# ipv6 address B000::1/64
!
Device(config-if-vrf)# router bgp 100
Device(config-if-vrf)# bgp log-neighbor-changes
!
Device(config-vrf)# address-family ipv4 vrf VRF1
Device(config-vrf-af)# redistribute connected
Device(config-vrf-af)# redistribute static
Device(config-vrf-af)# exit
!
Device(config-vrf)# address-family ipv6 vrf VRF1
Device(config-vrf-af)# redistribute connected
Device(config-vrf-af)# redistribute static
Device(config-vrf-af)# exit
!
Device(config-vrf)# address-family ipv4 vrf VRF2
Device(config-vrf-af)# redistribute connected
Device(config-vrf-af)# redistribute static
Device(config-vrf-af)# exit
!
Device(config-vrf)# address-family ipv6 vrf VRF2
Device(config-vrf-af)# redistribute connected
Device(config-vrf-af)# redistribute static
Device(config-vrf-af)# exit
!
Device(config)# utd
Device(config-utd)# all-interfaces
Device(config-utd)# engine standard
Device(config-utd)# exit

Device(config)# utd engine standard
Device(config-utd-eng-std)# logging syslog
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-engstd-insp)# threat protection
Device(config-utd-engstd-insp)# policy security
Device(config-utd-engstd-insp)# exist
Device(config-utd-eng-std)# exit
!
Device(config)# virtual-service utd
Device(config-virt-serv)# profile low
Device(config-virt-serv)# vnic gateway VirtualPortGroup0
Device(config-virt-serv-vnic)# guest ip address 192.0.2.2
Device(config-virt-serv-vnic)# exit
Device(config-virt-serv)# vnic gateway VirtualPortGroup1
Device(config-virt-serv-vnic)# guest ip address 192.0.2.6
Device(config-virt-serv-vnic)# exit
Device(config-virt-serv)# activate
```

```
UTD Snort IPS Drop Log
============================
2016/06/13-14:32:09.524475 IST [**] [Instance_ID: 1] [**] Drop [**]
[1:30561:1] BLACKLIST DNS request for known malware
domain domai.ddns2.biz - Win.Trojan.Beebone [**]
[Classification: A Network Trojan was Detected]
[Priority: 1] [VRF_ID: 2] {UDP} 11.1.1.10:58016 -> 21.1.1.10:53
```

例：IOS Syslog のロギングの設定

次に、デバイスのログレベルを使用して IOS syslog のロギングを設定する例を示します。

```
Device# configure terminal
Device(config)# utd engine standard
Device(config-utd-eng-std)# logging syslog
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-engstd-insp)# logging level debug
Device(config-utd-eng-std-insp)# end
Device#
```

例：中央集中型ログサーバへのロギングの設定

次の例は、中央集中型ログサーバへのロギングの設定方法を示しています。

```
Device# configure terminal
Device(config)# utd engine standard
Device(config-utd-eng-std-insp)# logging host syslog.yourcompany.com
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-eng-std-insp)# logging level info
Device(config-utd-eng-std-insp)# end
Device#
```

例：Cisco サーバからの署名更新の設定

次の例は、Cisco サーバから署名の更新を設定する方法を示しています。

```
Device# configure terminal
Device(config)# utd engine standard
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-eng-std-insp)# signature update server cisco username CCOuser password
 passwd123
Device(config-utd-eng-std-insp)# end
Device#
```

（注） DNS が Cisco サーバから署名をダウンロードするように設定されていることを確認します。

例：ローカルサーバからの署名更新の設定

次の例は、ローカルサーバから署名の更新を設定する方法を示しています。

```
Device# configure terminal
Device(config)# utd engine standard
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-eng-std-insp)# signature update server url http://192.168.1.2/sig-1.pkg
Device(config-utd-eng-std-insp)# end
Device#
```

例：自動署名更新の設定

次の例は、サーバで自動署名更新を設定する方法を示しています。

```
Device# configure terminal
Device(config)# utd engine standard
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-eng-std-insp)# signature update occur-at daily 0 0
Device(config-utd-eng-std-insp)# signature update server cisco username abcd password
cisco123
Device(config-utd-eng-std-insp)# end
Device#
```

例：手動による署名の更新の実行

次の例は、さまざまな方法で手動で署名を更新する方法を示しています。

```
Device# utd threat-inspection signature update
```

既存のサーバ設定をダウンロードするか、既存のサーバ設定を使用して設定された明示的なサーバ情報を取得します。これらのコマンドにより、次の設定を使用して手動で署名更新が実行されます。

```
Device# show utd engine standard threat-inspection signature update status

Current signature package version: 2983.4.s
Current signature package name: UTD-STD-SIGNATURE-2983-4-S.pkg
Previous signature package version: 29.0.c
----------------------------------------
Last update status: Successful
----------------------------------------
Last successful update time: Mon Aug  7 02:02:32 2017 UTC
Last successful update method: Manual
Last successful update server: cisco
Last successful update speed: 3022328 bytes in 25 secs
----------------------------------------
Last failed update time: Mon Aug  7 01:53:21 2017 UTC
Last failed update method: Manual
Last failed update server: cisco
Last failed update reason: ('Connection aborted.', gaierror(-2, 'Name or service hnot
known'))
----------------------------------------
Last attempted update time: Mon Aug  7 02:02:32 2017 UTC
Last attempted update method: Manual
```

```
Last attempted update server: cisco
----------------------------------------
Total num of updates successful: 1
Num of attempts successful: 1
Num of attempts failed: 3
Total num of attempts: 4
----------------------------------------
Next update scheduled at: None
----------------------------------------
Current status: Idle
```

```
Device# utd threat-inspection signature update server cisco username ccouser password
passwd123
```

```
Device# utd threat-inspection signature update server url http://192.168.1.2/sig-1.pkg
```

例：署名許可リストの設定

次の例は、署名の許可リストを設定する方法を示しています。

```
Device# configure terminal
Device(config)# utd threat-inspection whitelist
Device(config-utd-whitelist)# utd-whitelist)# generator id 1 signature id 23456 comment
 "traffic from client x"
Device(config-utd-whitelist)# exit
Device(config)# utd engine standard
Device(config-utd-eng-std)# threat-inspection
Device(config-utd-eng-std-insp)# whitelist
Device(config-utd-eng-std-insp)# end
Device#
```

（注） 許可リストの署名 ID が設定されると、Snort はフローがアラートやドロップなしでデバイスを通過できるようにします。

アクティブな署名の表示例

例：接続ポリシーを使用したアクティブな署名の表示

```
Device# utd threat-inspection signature active-list write-to bootflash:siglist_connectivity
Device# more bootflash:siglist_connectivity
================================================================================
Signature Package Version: 2982.1.s
Signature Ruleset: Connectivity
Total no. of active signatures: 581
Total no. of drop signatures: 452
Total no. of alert signatures: 129

For more details of each signature please go to www.snort.org/rule_docs to lookup
================================================================================
List of Active Signatures:
```

```
-------------------------
<snipped>
```

例：バランスの取れたポリシーを使用したアクティブな署名の表示

```
Device# utd threat-inspection signature active-list write-to bootflash:siglist_balanced
Device# more bootflash:siglist_balanced
================================================================================
Signature Package Version: 2982.1.s
Signature Ruleset: Balanced
Total no. of active signatures: 7884
Total no. of drop signatures: 7389
Total no. of alert signatures: 495

For more details of each signature please go to www.snort.org/rule_docs to lookup
================================================================================

List of Active Signatures:
-------------------------
<snipped>
```

例：セキュリティポリシーを使用したアクティブな署名の表示

```
Device# utd threat-inspection signature active-list write-to bootflash:siglist_security
Device# more bootflash:siglist_security
================================================================================
Signature Package Version: 2982.1.s
Signature Ruleset: Security
Total no. of active signatures: 11224
Total no. of drop signatures: 10220
Total no. of alert signatures: 1004

For more details of each signature please go to www.snort.org/rule_docs to lookup
================================================================================

List of Active Signatures:
-------------------------
<snipped>
```

統合型 Snort IPS 設定の確認

次のコマンドを使用して、設定をトラブルシューティングします。

手順の概要

1. **enable**
2. **show virtual-service list**
3. **show virtual-service detail**
4. **show service-insertion type utd service-node-group**
5. **show service-insertion type utd service-context**
6. **show utd engine standard config**
7. **show utd engine standard status**
8. **show utd engine standard threat-inspection signature update status**

9. **show utd engine standard logging events**
10. **clear utd engine standard logging events**
11. **show platform hardware qfp active feature utd config**
12. **show platform software utd global**
13. **show platform software utd interfaces**
14. **show platform hardware qfp active feature utd stats**
15. **show utd engine standard statistics daq all**

手順の詳細

ステップ 1 enable

例：

```
Device> enable
```

特権 EXEC モードを有効にします。

- パスワードを入力します（要求された場合）。

ステップ 2 show virtual-service list

仮想サービスコンテナ上のすべてのアプリケーションのインストールのステータスを表示します。

例：

```
Device# show virtual-service list

Virtual Service List:


Name                     Status              Package Name
------------------------------------------------------------------------------
UTDIPS                   Activated           utdsnort.1_0_1_SV2982_XE_16_3.20160701_131509.ova
```

ステップ 3 show virtual-service detail

デバイスの仮想サービスコンテナにインストールされているアプリケーションによって使用されるリソースを表示します。

例：

```
Device# show virtual-service detail


Device#show virtual-service detail
Virtual service UTDIPS detail
  State                : Activated
  Owner                : IOSd
  Package information
    Name               : utdsnort.1_0_1_SV2982_XE_16_3.20160701_131509.ova
    Path               : bootflash:/utdsnort.1_0_1_SV2982_XE_16_3.20160701_131509.ova
    Application
      Name             : UTD-Snort-Feature
      Installed version : 1.0.1_SV2982_XE_16_3
      Description      : Unified Threat Defense
```

```
    Signing
      Key type          : Cisco development key
      Method            : SHA-1
    Licensing
      Name              : Not Available
      Version           : Not Available

  Detailed guest status

-------------------------------------------------------------------------
Process               Status            Uptime           # of restarts
-------------------------------------------------------------------------
climgr                 UP         0Y 0W 0D  0: 0:35        1
logger                 UP         0Y 0W 0D  0: 0: 4        0
snort_1                UP         0Y 0W 0D  0: 0: 4        0
Network stats:
eth0: RX  packets:43, TX  packets:6
eth1: RX  packets:8, TX  packets:6

Coredump file(s): lost+found

  Activated profile name: None
  Resource reservation
    Disk                : 736 MB
    Memory              : 1024 MB
    CPU                 : 25% system CPU

  Attached devices
    Type              Name        Alias
    ---------------------------------------------
    NIC               ieobc_1     ieobc
    NIC               dp_1_0      net2
    NIC               dp_1_1      net3
    NIC               mgmt_1      mgmt
    Disk              _rootfs
    Disk              /opt/var
    Disk              /opt/var/c
    Serial/shell                  serial0
    Serial/aux                    serial1
    Serial/Syslog                 serial2
    Serial/Trace                  serial3
    Watchdog          watchdog-2

  Network interfaces
    MAC address             Attached to interface
    ------------------------------------------------------
    54:0E:00:0B:0C:02       ieobc_1
    A4:4C:11:9E:13:8D       VirtualPortGroup0
    A4:4C:11:9E:13:8C       VirtualPortGroup1
    A4:4C:11:9E:13:8B       mgmt_1

  Guest interface
  ---
  Interface: eth2
  ip address: 48.0.0.2/24
Interface: eth1
  ip address: 47.0.0.2/24

  ---

  Guest routes
  ---
  Address/Mask                        Next Hop                          Intf.
-------------------------------------------------------------------------------
```

```
0.0.0.0/0                           48.0.0.1                        eth2
0.0.0.0/0                           47.0.0.1                        eth1

  ---

  Resource admission (without profile) : passed
    Disk space    : 710MB
    Memory        : 1024MB
    CPU           : 25% system CPU
    VCPUs         : Not specified
```

ステップ 4 show service-insertion type utd service-node-group

サービスノードグループのステータスを表示します。

例：

```
Device# show service-insertion type utd service-node-group

Service Node Group name : utd_sng_1
Service Context : utd/1
Member Service Node count : 1


Service Node (SN) : 30.30.30.2
Auto discovered : No
SN belongs to SNG : utd_sng_1
Current status of SN : Alive
Time current status was reached : Tue Jul 26 11:57:48 2016

Cluster protocol VPATH version : 1
Cluster protocol incarnation number : 1
Cluster protocol last sent sequence number : 1469514497
Cluster protocol last received sequence number: 1464
Cluster protocol last received ack number : 1469514496
```

ステップ 5 show service-insertion type utd service-context

AppNav およびサービスノードビューを表示します。

例：

```
Device# show service-insertion type utd service-context

Service Context : utd/1
Cluster protocol VPATH version : 1
Time service context was enabled : Tue Jul 26 11:57:47 2016
Current FSM state : Operational
Time FSM entered current state : Tue Jul 26 11:57:58 2016
Last FSM state : Converging
Time FSM entered last state : Tue Jul 26 11:57:47 2016
Cluster operational state : Operational

Stable AppNav controller View:
30.30.30.1

Stable SN View:
30.30.30.2

Current AppNav Controller View:
30.30.30.1

Current SN View:
```

```
30.30.30.2
```

ステップ 6 show utd engine standard config

統合脅威防御（UTD）の設定を表示します。

例：

```
Device# show utd engine standard config


UTD Engine Standard Configuration:
  Operation Mode : Intrusion Prevention
  Policy         : Security

Signature Update:
  Server    : cisco
  User Name : ccouser
  Password  : YEX^SH\fhdOeEGaOBIQAIcOVLgaVGf
  Occurs-at : weekly ;  Days:0 ; Hour: 23; Minute: 50

Logging:
  Server    :   IOS Syslog; 10.104.49.223
  Level     : debug

Whitelist Signature IDs:
  28878
```

ステップ 7 show utd engine standard status

UTD エンジンのステータスを表示します。

例：

```
Device# show utd engine standard status

Profile : High
System memory :
Usage : 8.00 %
Status : Green
Number of engines : 4

Engine Running CFT flows Health Reason
======================================================
Engine(#1): Yes 0 Green None
Engine(#2): Yes 0 Green None
Engine(#3): Yes 0 Green None
Engine(#4): Yes 0 Green None
======================================================

Overall system status: Green

Signature update status:
========================
Current signature package version: 2983.4.s
Last update status: Successful
Last successful update time: Mon Aug  7 02:02:32 2017 UTC
Last failed update time: Mon Aug  7 01:53:21 2017 UTC
Last failed update reason: ('Connection aborted.', gaierror(-2, 'Name or service not known'))
Next update scheduled at: None
Current status: Idle
```

ステップ 8 **show utd engine standard threat-inspection signature update status**

署名更新プロセスのステータスを表示します。

例：

```
Device# show utd engine standard threat-inspection signature update status

Current signature package version: 2983.4.s
Current signature package name: UTD-STD-SIGNATURE-2983-4-S.pkg
Previous signature package version: 29.0.c
----------------------------------------
Last update status: Successful
----------------------------------------
Last successful update time: Mon Aug  7 02:02:32 2017 UTC
Last successful update method: Manual
Last successful update server: cisco
Last successful update speed: 3022328 bytes in 25 secs
----------------------------------------
Last failed update time: Mon Aug  7 01:53:21 2017 UTC
Last failed update method: Manual
Last failed update server: cisco
Last failed update reason: ('Connection aborted.', gaierror(-2, 'Name or service hnot known'))
----------------------------------------
Last attempted update time: Mon Aug  7 02:02:32 2017 UTC
Last attempted update method: Manual
Last attempted update server: cisco
----------------------------------------
Total num of updates successful: 1
Num of attempts successful: 1
Num of attempts failed: 3
Total num of attempts: 4
----------------------------------------
Next update scheduled at: None
----------------------------------------
Current status: Idle
```

ステップ 9 **show utd engine standard logging events**

Snort センサーからのログイベントを表示します。

例：

```
Device# show utd engine standard logging events

2016/06/13-14:32:09.524475 IST [**] [Instance_ID: 1] [**] Drop [**] [1:30561:1]
BLACKLIST DNS request for known malware domain domai.ddns2.biz -
Win.Trojan.Beebone [**] [Classification: A Network Trojan was Detected] [Priority: 1]
[VRF_ID: 2] {UDP} 11.1.1.10:58016 -> 21.1.1.10:53
2016/06/13-14:32:21.524988 IST [**] [Instance_ID: 1] [**] Drop [**] [1:30561:1]
BLACKLIST DNS request for known malware domain domai.ddns2.biz -
Win.Trojan.Beebone [**] [Classification: A Network Trojan was Detected] [Priority: 1]
[VRF_ID: 2] {UDP} a000:0:0:0:0:0:0:10:59964 -> b000:0:0:0:0:0:0:10:53
```

ステップ 10 **clear utd engine standard logging events**

例：

```
Device# clear utd engine standard logging events
```

Snort センサーからのログイベントをクリアします。

ステップ 11 **show platform hardware qfp active feature utd config**

サービスノードの正常性に関する情報を表示します。

例：

```
Device# show platform hardware qfp active feature utd config

Global configuration
NAT64: disabled
SN threads: 12
CFT inst_id 0 feat id 1 fo id 1 chunk id 8
Context Id: 0, Name: Base Security Ctx
Ctx Flags: (0x60000)
Engine: Standard
SN Redirect Mode : Fail-open, Divert
Threat-inspection: Enabled, Mode: IDS
Domain Filtering : Not Enabled
URL Filtering : Not Enabled
SN Health: Green
```

ステップ 12 show platform software utd global

UTD が有効になっているインターフェイスを表示します。

例：

```
Device# show platform software utd global

UTD Global state
Engine               : Standard
Global Inspection    : Enabled
Operational Mode     : Intrusion Prevention
Fail Policy          : Fail-open
Container techonlogy : LXC
Redirect interface   : VirtualPortGroup1
UTD interfaces
All dataplane interfaces
```

ステップ 13 show platform software utd interfaces

すべてのインターフェイスに関する情報を表示します。

例：

```
Device# show platform software utd interfaces

UTD interfaces
All dataplane interfaces
```

ステップ 14 show platform hardware qfp active feature utd stats

データプレーンの UTD 統計情報を表示します。

例：

```
Device# show platform hardware qfp active feature utd stats

Security Context:    Id:0    Name: Base Security Ctx

Summary Statistics:
Pkts entered policy feature                         pkt                  228
```

```
                                                    byt                31083

Drop Statistics:

Service Node flagged flow for dropping                                    48
Service Node not healthy                                                  62

General Statistics:

Non Diverted Pkts to/from divert interface                             32913
Inspection skipped - UTD policy not applicable                         48892
Policy already inspected                                                2226
Pkts Skipped - L2 adjacency glean                                          1
Pkts Skipped - For Us                                                     67
Pkts Skipped - New pkt from RP                                           102
Response Packet Seen                                                     891
Feature memory allocations                                               891
Feature memory free                                                      891
Feature Object Delete                                                    863

Service Node Statistics:
SN Health: Green
SN down                                                                   85
SN health green                                                           47
SN health red                                                             13

Diversion Statistics
redirect                                                                2226
encaps                                                                  2226
decaps                                                                  2298
reinject                                                                2250
decaps: Could not locate flow                                             72
Redirect failed, SN unhealthy                                             62
Service Node requested flow bypass drop                                   48
```

ステップ 15　show utd engine standard statistics daq all

サービスプレーンのデータ収集（DAQ）の統計情報を表示します。

例：

```
Device# show utd engine standard statistics daq all

IOS-XE DAQ Counters(Engine #1):
--------------------------------
Frames received                       :0
Bytes received                        :0
RX frames released                    :0
Packets after vPath decap             :0
Bytes after vPath decap               :0
Packets before vPath decap            :0
Bytes before vPath decap              :0
Frames transmitted                    :0
Bytes transmitted                     :0

Memory allocation                     :2
Memory free                           :0
Merged packet buffer allocation       :0
Merged packet buffer free             :0

VPL buffer allocation                 :0
VPL buffer free                       :0
VPL buffer expand                     :0
```

```
VPL buffer merge                        :0
VPL buffer split                        :0
VPL packet incomplete                   :0

VPL API error                           :0
CFT API error                           :0
Internal error                          :0
External error                          :0
Memory error                            :0
Timer error                             :0

Kernel frames received                  :0
Kernel frames dropped                   :0

FO cached via timer                     :0
Cached fo used                          :0
Cached fo freed                         :0
FO not found                            :0
CFT full packets                        :0


VPL Stats(Engine #1):
------------------------
```

Cisco Prime CLI テンプレートを使用した Snort IPS の導入

Cisco Prime CLI テンプレートを使用して、Snort IPS 導入をプロビジョニングすることができます。Cisco Prime CLI テンプレートを使用すると、Snort IPS 導入を容易にプロビジョニングできます。Cisco Prime CLI テンプレートを Snort IPS 導入のプロビジョニングに使用するには、次の手順を実行します。

ステップ 1 システムで実行されている IOS XE バージョンに対応する Prime テンプレートをソフトウェアのダウンロードページからダウンロードします。

ステップ 2 このファイルが圧縮されている場合は解凍します。

ステップ 3 Prime から、[Configuration] > [Templates] > [Features and Technologies] の順に選び、[CLI Templates] を選択します。

ステップ 4 [Import] をクリックします。

ステップ 5 テンプレートのインポート先フォルダを選択し、[Select Templates] をクリックして、先ほどダウンロードしたテンプレートを選択してインポートします。

次の Snort IPS CLI テンプレートを使用できます。

- Copy OVA to Device：このテンプレートを使用して、Snort IPS OVA ファイルをルータのファイルシステムにコピーします。
- Delete OVA：このテンプレートを使用して、コピーした Snort IPS OVA ファイルをルータのファイルシステムから削除します。

- Dynamic NAT：ダイナミック NAT（ネットワークアドレス変換）が環境内で設定されており、Snort IPS 管理インターフェイス IP 用に変更する必要がある NAT 変換を選択するためにアクセスリストを使用する場合は、このテンプレートを使用します。
- Dynamic NAT Cleanup：このテンプレートを使用して、Snort IPS の NAT 設定を削除します。
- Dynamic PAT：環境内でダイナミック PAT（ポートアドレス変換）が設定されており、Snort IPS 管理インターフェイスIP用に変更する必要のある PAT 変換を選択するためにアクセスリストを使用する場合は、このテンプレートを使用します。
- Dynamic NAT Cleanup：このテンプレートを使用して、Snort IPS の PAT 設定を削除します。
- IP Unnumbered：このテンプレートを使用して、Snort IPS および IP 番号なしの導入に必要な仮想サービスを設定します。
- IP Unnumbered Cleanup：このテンプレートを使用して、IP 番号なしで設定された Snort IPS 管理インターフェイスを削除します。
- Management Interface：Snort IPS 管理トラフィックのルーティングにシステム管理インターフェイス（GigabitEthernet0 など）を使用する場合は、このテンプレートを使用します。
- Management Interface Cleanup：このテンプレートを使用して、Snort IPS 管理トラフィックをルーティングするために設定されたシステム管理インターフェイス（GigabitEthernet0 など）を削除します。
- Static NAT：このテンプレートを使用して、Snort IPS および既存の静的 NAT の導入に必要な仮想サービスを設定します。
- Static NAT Cleanup：このテンプレートを使用して、静的 NAT の導入で設定された Snort IPS を削除します。
- Upgrade OVA：このテンプレートを使用して、Snort IPS の OVA ファイルをアップグレードします。

IOx コンテナへの移行

ここでは、Cisco 1000 シリーズサービス統合型ルータ（ISR）での UTD 対応を拡張するための、Cisco IOx および IOx への UTD の移行について説明します。Cisco IOx では Cisco IOS と Linux OS が組み合わされており、安全性の高いネットワークを実現します。

Cisco IOx について

Cisco IOx は、さまざまな Cisco プラットフォームにおける各種アプリケーションに統一された一貫性のあるホスティング機能を提供するアプリケーション プラットフォームです。このプラットフォームは、ネットワーキング オペレーティング システム（Cisco IOS）とオープンソースのプラットフォーム（Linux）を統合し、ネットワーク上のカスタムアプリケーションとインターフェイスを実現します。

仮想サービス コンテナはデバイスの仮想化環境です。仮想マシン（VM）、仮想サービス、またはコンテナとも呼ばれます。仮想サービス コンテナ内にアプリケーションをインストールできます。このアプリケーションは、デバイスのオペレーティング システムの仮想サービス コンテナ内で稼動します。アプリケーションは、拡張子 .ova を持つ tar ファイルである Open Virtual Application（OVA）として提供されます。OVA パッケージは、コマンドラインのインターフェイスを介してデバイスにインストールされ、有効化されます。オープンフローの Cisco プラグインは、仮想サービスコンテナ内に導入できるアプリケーションの一例です。

UTD OVA をホストするために使用される仮想サービスコンテナのインフラストラクチャは、Cisco 1100 シリーズ ISR では対応していません。現在、UTD は両方のコンテナに対応しています。ただし、OVA コンテナ機能は Cisco IOS XE Gibrafilter 16.10 のリリースでは対応していますが、それ以降のリリースでは対応していません。

仮想サービスコンテナから IOx へのアップグレード

OVA ファイルは、仮想マシンの圧縮された「インストール可能な」バージョンを含むオープン仮想アーカイブ（Open Virtualization Archive）です。Snort IPS は仮想コンテナサービスとして使用できます。この OVA ファイルをデバイスにダウンロードし、**virtual-service install** CLI を使用してサービスをインストールする必要があります。

UTD IOx インフラストラクチャの場合、IOx ベースの OVA は IOx CLI コマンドを使用してインストールします。インストールする前に、グローバル設定モードで IOx 環境を開始します。

IOx ベースの OVA は TAR ファイルと呼ばれます。セキュリティライセンスが付属した Cisco IOS XE イメージを使用する必要があります。OVA ファイルのインストール中に、セキュリティライセンスがチェックされ、ライセンスが存在しない場合はエラーが報告されます。

仮想サービスから IOx コンテナにアップグレードするには、次の手順を実行します。

ステップ 1 **no activate**

例：

```
Device# configure terminal
Device (config)# virtual-service utd
Device (config-virt-serv)# no activate
Device (config-virt-serv)# exit
Device (config)# no virtual-service utd
```

仮想マネージャベースの仮想サービスのインスタンスを非アクティブにします。

ステップ 2 **show virtual-service list**

例：

```
Device# show virtual-service list
```

仮想サービスコンテナにインストールされているすべてのアプリケーションのステータスを表示します。仮想サービスインスタンスが非アクティブになっていることを確認します。

ステップ 3 **virtual-service uninstall name** *virtual-service instance*

例：

```
Device# virtual-service uninstall name utd
```

仮想マネージャベースの仮想サービスインスタンスをアンインストールします。**show virtual-service list** コマンドを実行したときに、仮想サービスインスタンスが表示されないことを確認します。

ステップ 4 **iox**

例：

```
Device# configure terminal
Device (config)# iox
Device (config)# end
```

IOx環境をグローバル設定モードで開始します。

ステップ 5 **app-hosting install appid** *name* **package** *bootflash:<tarfile>*

例：

```
Device# app-hosting install appid UTD package bootflash:utd.tar
Device#
```

Iox ベースの OVA tar ファイルをデバイスにコピーしてインストールします。

ステップ 6 **show app-hosting list**

例：

```
Device# show app-hosting list
App id                                   State
---------------------------------------------------------
UTD                                      DEPLOYED
Device#
```

インストールのステータスを表示します。アプリケーションが展開されていることを確認します。

ステップ 7 **app-hosting activate appid** *name*

例：

```
Device# app-hosting activate appid UTD
```

デバイス上の IOx ベースの TAR ファイルをアクティブにします。

ステップ 8 **show app-hosting list**

例：

```
Device# show app-hosting list
App id                                   State
---------------------------------------------------------
UTD                                      ACTIVATED

Device#
```

アクティベーションのステータスが表示されます。アプリケーションがアクティブになっていることを確認します。

ステップ 9 **app-hosting start appid** *name*

例：

```
Device# app-hosting start appid UTD
Device# show app-hosting list | in UTD
```

IOx ベースの OVA を開始します。

ステップ 10 **show app-hosting list**

例：

```
Example:
Device# show app-hosting list
App id                                   State
---------------------------------------------------------
UTD                                      RUNNING

Device#
```

開始のステータスを表示します。アプリケーションが実行されていることを確認します。

IOx の設定例

IOx の設定例を次に示します。

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# iox
Device(config)# interface VirtualPortGroup0
Device(config-if)# no shutdown
Device(config-if)# ip address 192.0.2.1 255.255.255.252
Device(config-if)# exit
Device(config)# interface VirtualPortGroup1
Device(config-if)# no shutdown
Device(config-if)# ip address 192.0.2.5 255.255.255.252
Device(config-if)# exit
Device(config)# app-hosting appid utd
Device(config-app-hosting)# app-vnic gateway0 virtualportgroup 0 guest-interface 0
Device(config-app-hosting-gateway0)# guest-ipaddress 192.0.2.2 netmask 255.255.255.252
Device(config-app-hosting-gateway0)# exit
Device(config-app-hosting)# app-vnic gateway1 virtualportgroup 1 guest-interface 1
Device(config-app-hosting-gateway1)# guest-ipaddress 192.0.2.6 netmask 255.255.255.252
Device(config-app-hosting-gateway1)# exit
Device(config-app-hosting)# app-resource package-profile custom
Device(config-app-hosting)# start
Device(config-app-hosting)# exit
Device(config)# exit
Device#
```

Snort IPS のトラブルシューティング

トラフィックが転送されない

問題 トラフィックは転送されません。

考えられる原因 仮想サービスがアクティブになっていない可能性があります。

解決法 **show virtual-service list** コマンドを使用して、仮想サービスがアクティブになっているかどうかを確認します。次に、コマンドの出力例を示します。

```
Device# show virtual-service list

Virtual Service List:


Name Status Package Name
--------------------------------------------------------------------------------
snort Activated utdsnort.1_0_1_SV2982_XE_16_3.20160701_131509.ova
```

考えられる原因 指定されたインターフェイスでは、統合脅威防御（UTD）が有効になっていない可能性があります。

解決法 **show platform software utd global** コマンドを使用して、インターフェイスで UTD が有効になっているかどうかを確認します。

```
Device# show platform software utd global

UTD Global state
Engine               : Standard
Global Inspection    : Disabled
Operational Mode     : Intrusion Prevention
Fail Policy          : Fail-open
Container techonlogy : LXC
Redirect interface   : VirtualPortGroup1
UTD interfaces
GigabitEthernet0/0/0
```

考えられる原因 サービスノードが正常に動作していない可能性があります。

解決法 **show platform hardware qfp active feature utd config** コマンドを使用して、サービスノードの状態が緑色かどうかを確認します。

```
Device# show platform hardware qfp active feature utd config

Global configuration
NAT64: disabled
SN threads: 12
CFT inst_id 0 feat id 0 fo id 0 chunk id 4
Context Id: 0, Name: Base Security Ctx
Ctx Flags: (0x60000)
Engine: Standard
SN Redirect Mode : Fail-open, Divert
Threat-inspection: Enabled, Mode: IDS
Domain Filtering : Not Enabled
URL Filtering : Not Enabled
SN Health: Green
```

考えられる原因 Snort プロセスがアクティブになっていない可能性があります。

解決法 **show virtual-service detail** コマンドを使用して、Snortプロセス が稼働しているかどうかを確認します。

```
Device# show virtual-service detail

Virtual service UTDIPS detail
  State                : Activated
  Owner                : IOSd
  Package information
```

```
    Name                : utdsnort.1_0_1_SV2982_XE_16_3.20160701_131509.ova
    Path                : bootflash:/utdsnort.1_0_1_SV2982_XE_16_3.20160701_131509.ova
    Application
      Name              : UTD-Snort-Feature
      Installed version : 1.0.1_SV2982_XE_16_3
      Description       : Unified Threat Defense
    Signing
      Key type          : Cisco development key
      Method            : SHA-1
    Licensing
      Name              : Not Available
      Version           : Not Available

  Detailed guest status

------------------------------------------------------------------------
Process               Status            Uptime           # of restarts
------------------------------------------------------------------------
climgr                 UP         0Y 0W 0D  0: 0:35        1
logger                 UP         0Y 0W 0D  0: 0: 4        0
snort_1                UP         0Y 0W 0D  0: 0: 4        0
Network stats:
eth0: RX  packets:43, TX  packets:6
eth1: RX  packets:8, TX  packets:6

Coredump file(s): lost+found

  Activated profile name: None
  Resource reservation
    Disk                : 736 MB
    Memory              : 1024 MB
    CPU                 : 25% system CPU

  Attached devices
    Type              Name        Alias
    ---------------------------------------------
    NIC               ieobc_1     ieobc
    NIC               dp_1_0      net2
    NIC               dp_1_1      net3
    NIC               mgmt_1      mgmt
    Disk              _rootfs
    Disk              /opt/var
    Disk              /opt/var/c
    Serial/shell                  serial0
    Serial/aux                    serial1
    Serial/Syslog                 serial2
    Serial/Trace                  serial3
    Watchdog          watchdog-2

  Network interfaces
    MAC address             Attached to interface
    ------------------------------------------------------
    54:0E:00:0B:0C:02       ieobc_1
    A4:4C:11:9E:13:8D       VirtualPortGroup0
    A4:4C:11:9E:13:8C       VirtualPortGroup1
    A4:4C:11:9E:13:8B       mgmt_1

  Guest interface
  ---
  Interface: eth2
  ip address: 48.0.0.2/24
Interface: eth1
  ip address: 47.0.0.2/24
```

```
  ---

  Guest routes
  ---
  Address/Mask                            Next Hop                              Intf.
------------------------------------------------------------------------------------
0.0.0.0/0                               48.0.0.1                              eth2
0.0.0.0/0                               47.0.0.1                              eth1

  ---

  Resource admission (without profile) : passed
    Disk space    : 710MB
    Memory        : 1024MB
    CPU           : 25% system CPU
    VCPUs         : Not specified
```

考えられる原因 AppNav トンネルがアクティブになっていない可能性があります。

解決法 **show service-insertion type utd service-node-group** および **show service-insertion type utd service-context** コマンドを使用して、AppNav トンネルがアクティブになっているかどうかを確認します。

解決法 次に、**show service-insertion type utd service-node-group** コマンドの出力例を示します。

```
Device# show service-insertion type utd service-node-group

Service Node Group name : utd_sng_1
Service Context : utd/1
Member Service Node count : 1


Service Node (SN) : 30.30.30.2
Auto discovered : No
SN belongs to SNG : utd_sng_1
Current status of SN : Alive
Time current status was reached : Tue Jul 26 11:57:48 2016

Cluster protocol VPATH version : 1
Cluster protocol incarnation number : 1
Cluster protocol last sent sequence number : 1469514497
Cluster protocol last received sequence number: 1464
Cluster protocol last received ack number : 1469514496
```

解決法 次に、**show service-insertion type utd service-context** コマンドの出力例を示します。

```
Device# show service-insertion type utd service-context

Service Context : utd/1
Cluster protocol VPATH version : 1
Time service context was enabled : Tue Jul 26 11:57:47 2016
Current FSM state : Operational
Time FSM entered current state : Tue Jul 26 11:57:58 2016
Last FSM state : Converging
Time FSM entered last state : Tue Jul 26 11:57:47 2016
Cluster operational state : Operational

Stable AppNav controller View:
30.30.30.1

Stable SN View:
```

```
30.30.30.2

Current AppNav Controller View:
30.30.30.1

Current SN View:
30.30.30.2
```

考えられる原因 トラフィックのステータスのデータプレーンUTD統計情報を確認します。トラフィックが転送されない場合、転送および拒否されたパケットの数はゼロになります。数値がゼロ以外の場合、トラフィック転送が行われており、Snort センサーはデータプレーンにパケットを再送信しています。

解決法 **show platform hardware qfp active feature utd stats** コマンドを使用してトラフィックのステータスを確認します。

```
Device# show platform hardware qfp active feature utd stats

Security Context:    Id:0    Name: Base Security Ctx

Summary Statistics:
Active Connections                                                    29
TCP Connections Created                                           712910
UDP Connections Created                                               80
Pkts entered policy feature                          pkt            3537977
                                                     byt          273232057
Pkts entered divert feature                          pkt            3229148
                                                     byt          249344841
Pkts slow path                                       pkt             712990
                                                     byt           45391747
Pkts Diverted                                        pkt            3224752
                                                     byt          249103697
Pkts Re-injected                                     pkt            3224746
                                                     byt          249103373
….
```

署名の更新が機能しない

問題 Cisco ボーダレスソフトウェア配布（BSD：Borderless Software Distribution）サーバからの署名更新が機能していません。

考えられる原因 さまざまな理由により署名の更新に失敗した可能性があります。最後に署名の更新に失敗した理由を確認します。

解決法 **show utd engine standard threat-inspection signature update status** コマンドを使用して、最後に署名の更新に失敗した理由を表示します。

```
Device# show utd eng standard threat-inspection signature update status
Current signature package version: 29.0.c
Current signature package name: default
Previous signature package version: None
----------------------------------------
Last update status: Failed
----------------------------------------
Last successful update time: None
```

```
Last successful update method: None
Last successful update server: None
Last successful update speed: None
----------------------------------------
Last failed update time: Thu Jan 11 13:34:36 2018 PST
Last failed update method: Manual
Last failed update server: http://172.27.57.252/UTD-STD-SIGNATURE-2983-1-S.pkg
Last failed update reason: [Errno 113] No route to host
----------------------------------------
Last attempted update time: Thu Jan 11 13:34:36 2018 PST
Last attempted update method: Manual
Last attempted update server: http://172.27.57.252/UTD-STD-SIGNATURE-2983-1-S.pkg
----------------------------------------
Total num of updates successful: 0
Num of attempts successful: 0
Num of attempts failed: 1
Total num of attempts: 1
----------------------------------------
Next update scheduled at: None
----------------------------------------
Current status: Idle
```

考えられる原因 ドメインネームシステム（DNS）が正しく設定されていません。

解決法 **show running-config | i name-server** コマンドを使用して、ネームサーバの詳細を表示します。

```
Device#  show run | i name-server

ip name-server 10.104.49.223
```

考えられる原因 システムエラー：ユーザ名とパスワードの組み合わせの処理に失敗しました。

解決法 署名パッケージのダウンロードに正しい認証情報を使用したことを確認します。

ローカルサーバからの署名の更新が機能しない

問題 ローカルサーバからの署名の更新が機能しない。

考えられる原因 最後の失敗の理由：無効なスキーム — HTTP または HTTPS のみに対応します。

解決法 ローカルダウンロード方式として HTTP またはセキュア HTTP（HTTPS）が指定されていることを確認します。

考えられる原因 最後の失敗の理由：名前またはサービスが不明です。

解決法 ローカルサーバに指定されたホスト名または IP アドレスが正しいことを確認します。

考えられる原因 最後の失敗の理由：認証情報が入力されていません。

解決法 ローカル HTTP または HTTPS サーバの認証情報が入力されていることを確認します。

考えられる原因 最後の失敗の理由：ファイルが見つかりません。

解決法 入力した署名ファイル名または URL が正しいことを確認します。

考えられる原因 最後の失敗の理由：ダウンロードが破損しています。

解決法

- 以前の署名のダウンロード時に署名更新の再試行でエラーが発生していないかどうかを確認します。
- 正しい署名パッケージが使用可能であることを確認します。

IOSd Syslog へのロギングが機能しない

問題 IOSd syslog へのロギングが機能しない。

考えられる原因 syslog へのロギングは、統合脅威防御（UTD）の設定では設定できません。

解決法 UTD 設定を表示し、syslog へのロギングが設定されていることを確認するには、**show utd engine standard config** コマンドを使用します。

```
Device# show utd engine standard config

UTD Engine Standard Configutation:
  Operation Mode : Intrusion Prevention
  Policy         : Security

Signature Update:
  Server    : cisco
  User Name : ccouser
  Password  : YEX^SH\fhdOeEGaOBIQAIcOVLgaVGf
  Occurs-at : weekly ;  Days:0 ; Hour: 23; Minute: 50

Logging:
  Server    :   IOS Syslog; 10.104.49.223
  Level     : debug

Whitelist Signature IDs:
  28878
```

解決法 UTD エンジンのイベントログを表示するには、次の **show utd engine standard logging events** コマンドを使用します。

```
Device# show utd engine standard logging events

2016/06/13-14:32:09.524475 IST [**] [Instance_ID: 1] [**] Drop [**] [1:30561:1]
BLACKLIST DNS request for known malware domain domai.ddns2.biz -
Win.Trojan.Beebone [**] [Classification: A Network Trojan was Detected]
[Priority: 1] [VRF_ID: 2] {UDP} 11.1.1.10:58016 -> 21.1.1.10:53
2016/06/13-14:32:21.524988 IST [**] [Instance_ID: 1] [**] Drop [**] [1:30561:1]
BLACKLIST DNS request for known malware domain domai.ddns2.biz -
Win.Trojan.Beebone [**] [Classification: A Network Trojan was Detected] [Priority: 1]
[VRF_ID: 2] {UDP} a000:0:0:0:0:0:0:10:59964 -> b000:0:0:0:0:0:0:10:53
```

外部サーバへのロギングが機能しない

問題 外部サーバへのロギングが機能していません。

考えられる原因 外部サーバで Syslog が実行されていない可能性があります。

解決法 syslog サーバが外部サーバで実行されているかどうかを確認します。ステータスを表示するには、外部サーバで次のコマンドを設定します。

```
ps -eaf | grep syslog

 root 2073 1 0 Apr12 ? 00:00:02 syslogd -r -m
```

考えられる原因 統合脅威防御（UTD）の Linux コンテナ（LXC：Linux Container）と外部サーバ間の接続が失われている可能性があります。

解決法 管理インターフェイスから外部 syslog サーバへの接続を確認します。

UTD 条件付きデバッグ

条件付きデバッグは、Unified Threat Defense のマルチテナントに対応しています。条件付きデバッグの設定方法の詳細については、以下を参照してください。

http://www.cisco.com/c/en/us/td/docs/routers/asr1000/troubleshooting/guide/Tbl-shooting-xe-3s-asr-1000-book.html#task_AC969BB06B414DCBBDEF7ADD29EF8131

Snort IPS に関するその他の参考資料

関連資料

関連項目	マニュアル タイトル
IOS コマンド	『Cisco IOS Master Command List, All Releases』[英語]
セキュリティ コマンド	• 『Cisco IOS Security Command Reference: Commands A to C』[英語] • 『Cisco IOS Security Command Reference: Commands D to L』[英語] • 『Cisco IOS Security Command Reference: Commands M to R』[英語] • 『Cisco IOS Security Command Reference: Commands S to Z』[英語]

シスコのテクニカル サポート

説明	リンク
シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンライン リソースを提供しています。 お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service（Field Notice からアクセス）、Cisco Technical Services Newsletter、Really Simple Syndication（RSS）フィードなどの各種サービスに加入できます。 シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。	http://www.cisco.com/support

Snort IPS の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *2 : Snort IPS* の機能情報

機能名	リリース	機能情報
Snort IPS	Cisco IOS XE 3.16.1S、3.17S 以降のリリース	Snort IPS 機能は、Cisco IOS XE ベースのプラットフォームのブランチオフィスにおける侵入防止システム（IPS：Intrusion Prevention System）および侵入検知システム（IDS）を有効にします。この機能は、オープンソースの Snort ソリューションを使用して IPS と IDS を有効にします。
Snort IPS での VRF 対応	Cisco IOS XE Denali 16.3.1	Snort IPS 設定で仮想フラグメンテーションの再構成（VFR：Virtual Fragmentation Reassembl）に対応。
Cisco クラウドサービスルータ 1000v シリーズで Snort IPS に対応	Cisco IOS XE Denali 16.3.1	Cisco クラウドサービスルータ 1000v シリーズは Snort IPS に対応します。

機能名	リリース	機能情報
16.4 リリースにおける UTD Snort IPS の機能拡張	Cisco IOS XE Everest 16.4.1	16.4 リリースにおける UTD Snort IPS の機能拡張には、アクティブな署名のリストを表示する機能が追加されています。
脅威検知アラートの可視性 UTD サービスの有用性の強化	Cisco IOS XE Fuji 16.8.1	この機能は、脅威検知アラートの概要を提供します。次のコマンドが導入されています。 • **show utd engine standard logging statistics threat-inspection** • **show utd engine standard logging statistics threat-inspection** *detail* 次のコマンドは、UTD サービスの有用性の強化の一環として変更されています。 • **show utd engine standard status** • **show utd engine standard threat-inspection signature update status**
IOX コンテナへの UTD（IPS および URL フィルタリング）の移行	Cisco IOS XE Gibraltar 16.10.1	UTD は、仮想サービスコンテナを OVA から IOx に移行することで、Cisco 1100 シリーズ ISR に対応します。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。