

`iptables --list INPUT --line-numbers -n`コマンドは、INPUTチェーンの中身を行番号と共に、名前解決をせずに表示させます。次の例のルールは、ホスト10.1.0.192からのトラフィックをすべてドロップし、ローカルサブネット(10.1.0.0/24)からのICMPエコーリクエスト(ping)、すべてのネットワーク(0.0.0.0/0)からのDNS、ローカルサブネット(10.1.0.0/24)からのHTTP/HTTPSトラフィックを許可します。

```
Chain INPUT (policy DROP)
# target prot opt source      destination
1 DROP   all  --  10.1.0.192  0.0.0.0/0
2 ACCEPT  icmp --  10.10.0.0/24 0.0.0.0/0  icmptype 8
3 ACCEPT  udp  --  0.0.0.0/0  0.0.0.0/0  udp dpt:53
4 ACCEPT  tcp  --  0.0.0.0/0  0.0.0.0/0  tcp dpt:53
5 ACCEPT  tcp  --  10.1.0.0/24 0.0.0.0/0  tcp dpt:80
6 ACCEPT  tcp  --  10.1.0.0/24 0.0.0.0/0  tcp dpt:443
7 ACCEPT  all  --  0.0.0.0/0  0.0.0.0/0  ctstate RELATED,ESTABLISHED
```

宛先0.0.0.0/0は「すべての場所」を意味します。`ctstate`ルールは、確立済みまたは関連するセッションの一部であるトラフィックを許可するステートフルルールです。確立済みの接続はすでに許可されているので、処理の必要性を減らし、トラフィックフローへの影響を最小限にします。

次のコマンドは、新しいルールを行番号2として挿入し、ローカルサブネットからSSHサーバーのTCPポート(22)へのトラフィックを許可します。

```
iptables -I INPUT 2 -p tcp -s 10.1.0.0/24 --dport 22
-j ACCEPT
```

異なるスイッチを使用することで、ルールの追加(-A)、削除(-D)または置換(-R)を行えます。

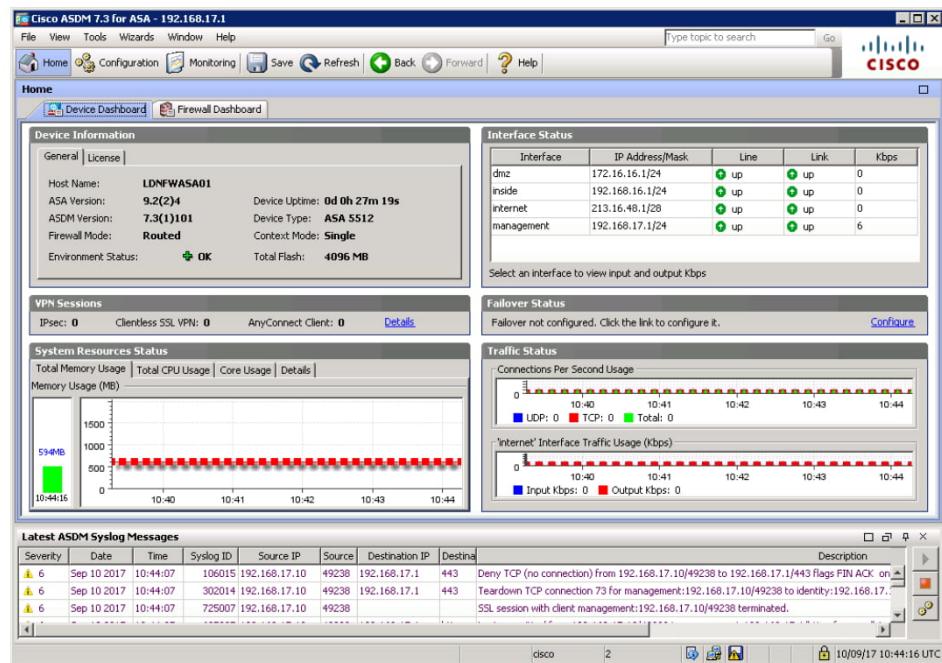
## ファイアウォールの実装

ファイアウォールは、ネットワーク上の場所や用途をカバーするために、ハードウェアまたはソフトウェアなど、どのように実装するかを検討する必要があります。ファイアウォールの中にはネットワークエッジやゾーン境界に配置するのに適したタイプもあれば、個々のホストを保護するよう設計されているタイプもあります。

### ファイアウォールアプライアンス

**アプライアンスファイアウォール**は、ネットワークゾーンに入りするトラフィックを監視するためにデプロイされる、スタンドアロンのハードウェアファイアウォールです。ファイアウォールアプライアンスは次の2つの方法でデプロイすることができます。

- ルーターとして（第3層） – ファイアウォールはサブネット間で転送を行います。ファイアウォール上の各インターフェイスは別々のサブネットに接続し、異なるセキュリティゾーンを表します。
- ブリッジとして（第2層） – ファイアウォールはルーターやスイッチなど、2つのノード間を行き来するトラフィックを検査します。これは透過モードとも呼ばれます。このファイアウォールにはIPインターフェイスがありません（構成管理を除く）。これは2つのノード間のイーサネットインターフェイスをブリッジします。このファイアウォールは第2層で転送を行いますが、あらゆる種類のパケットヘッダーを基にトラフィックの検査とフィルタリングを行えます。透過ファイアウォールの典型的な使用例として、サブネットを再構成して他のデバイスにIPアドレスを変更することなく、ファイアウォールをデプロイするというものがあります。



Cisco ASA (Adaptive Security Appliance)のASDM (Adaptive Security Device Manager)インターフェイス。(スクリーンショットはCiscoからの許可を得て使用。)

ルーターファイアウォールないしファイアウォールルーターAPライアンスは、ルーターのファームウェアの一部としてフィルタリング機能を実装したものです。ルーターAPライアンスの主たる目的はルーティングであり、ファイアウォールは二義的機能であるという点が違います。一例を挙げると、SOHO用のインターネットルーター/モデムにはファイアウォールが組み込まれています。

### アプリケーション型ファイアウォール

ファイアウォールはあらゆる種類のコンピューティングホスト上で、ソフトウェアとして動作させることもできます。アプリケーション型ファイアウォールにはいくつかのタイプがあります。

- ホスト型ファイアウォール**（またはパーソナルファイアウォール） – 単一のホスト上で動作するソフトウェアアプリケーションとして実装され、そのホストだけを保護します。パーソナルファイアウォールはパケットフィルタリングACLを実行するだけでなく、ソフトウェアプロセスがネットワークにアクセスするのを許可または拒否します。
- アプリケーションファイアウォール** – サーバー上で動作し、特定のアプリケーションだけを保護することを目的としたソフトウェアです（Webサーバーファイアウォールや、SQL Serverデータベースを保護することを目的としたファイアウォールなどがこれに該当します）。これはホスト型ファイアウォールの一種であり、通常はネットワークファイアウォールに追加する形でデプロイされます。
- NOS (Network Operating System)ファイアウォール** – WindowsやLinuxなどのネットワークサーバーOSで動作する、ソフトウェア型ファイアウォールです。このサーバーは、ネットワークセグメントのゲートウェイまたはプロキシとして機能します。

### プロキシとゲートウェイ

アプリケーションレイヤーのフィルタリングを行うファイアウォールは、プロキシとして実装されることがあります。ネットワークファイアウォールがトラフィックの許可とブロックのみを行うのに対し、**プロキシサーバー**はストアアンドフォワードモデルで動作します。プロキシは1つ1つのパケットを分解して分析を行い、それがルールに一致しているという条件の下に、そのパケットを再構築して転送します。



再構築の量はプロキシに依存します。プロキシの中にはIPとTCPヘッダーしか操作しないものもあります。アプリケーション対応のプロキシは、HTTPヘッダーを追加したり消去したりすることができます。ディープパケット検査プロキシは、HTTPペイロードからコンテンツを消去することができます。

## フォワードプロキシサーバー

フォワードプロキシはプロトコル固有のアウトバウンドトラフィックを提供します。一例を挙げると、LAN上のクライアントコンピューターがWebサイトに、またはインターネット上のセキュアなWebサイトに接続するのを可能にするWebプロキシをデプロイすることができます。これはアウトバウンドトラフィック用のTCPポート80と443にサービスを提供するフォワードプロキシです。

The screenshot shows the pfSense SquidGuard Blacklists configuration screen. The top navigation bar includes links for Package, SquidGuard, and Blacklists. Below the navigation is a toolbar with icons for refresh, search, and help. A sub-navigation bar at the top of the main content area includes General settings, Common ACL, Groups ACL, Target categories, Times, Rewrites, Blacklist (which is underlined), Log, and XMLRPC Sync. The main content area is titled "Blacklist Update". It features a progress bar indicating "0 %". Below the progress bar are three buttons: "Download" (green), "Cancel" (orange), and "Restore Default" (blue). A text input field below the buttons is labeled "Enter FTP or HTTP path to the blacklist archive here.".

pfSense上で動作しているSquidプロキシサーバー ([squid-cache.org](http://squid-cache.org))に透過プロキシの設定。このフィルターはACLと時間ベースの制限を適用し、ブラックリストを用いてURLへのアクセスを禁止することができます。（スクリーンショットはRubicon Communications, LLCからの許可を得て使用。）

プロキシの主な利点は、クライアントコンピュータがWebアクセスのために、境界ネットワーク上の指定されたポイントに接続することです。プロキシはDMZ内に配置することができます。これにより、ある程度のトラフィック管理とセキュリティがもたらされます。加えて、大半のWebプロキシサーバーはキャッシュエンジンを提供しており、頻繁にリクエストされるWebページをプロキシ上に保持し、後続のリクエストでそのページを再取得する必要はなくなります。

プロキシサーバーは、自身がサービスを提供しているアプリケーションを理解している必要があります。一例を挙げると、WebプロキシはHTTPやHTTPSコマンドを（場合によってはHTMLとスクриプトも）解析して修正できなければなりません。プロキシサーバーには特定のアプリケーション専用のものもあれば、多目的のものもあります。多目的プロキシは、HTTP、FTP、SMTPなど、さまざまなタイプのプロトコル用のフィルターと共に構成されます。

通常、プロキシサーバーは非透過型または透過型のいずれかに分類されます。

- 非透過プロキシ**は、クライアントがプロキシサーバーを使用するにあたり、そのアドレスとポート番号で構成されなければならないことを意味します。多くの場合、プロキシサーバーがクライアントの接続を受け入れるポートはport 8080.autoとして構成されます。
- 透過プロキシ（または強制プロキシないし傍受プロキシ）**は、クライアントを再構成することなくそのクライアントのトラフィックを傍受します。透過プロキシはスイッチやルーター、またはその他のインラインネットワークアプライアンスに実装されなければなりません。

**Transparent Proxy Settings**

**Transparent HTTP Proxy**

Enable transparent mode to forward all requests for destination port 80 to the proxy server.

**Transparent Proxy Interface(s)**

LAN  
WAN

The interface(s) the proxy server will transparently intercept requests on. Use CTRL + click to select multiple interfaces.

**Bypass Proxy for Private Address Destination**

Do not forward traffic to Private Address Space (RFC 1918) destinations.  
Destinations in Private Address Space ([RFC 1918](#)) are passed directly through the firewall, not through the proxy server.

**Bypass Proxy for These Source IPs**

Do not forward traffic from these source IPs, CIDR nets, hostnames, or aliases through the proxy server but let it pass directly through the firewall.  
Applies only to transparent mode. Separate entries by semi-colons ( ; )

**Bypass Proxy for These Destination IPs**

Do not proxy traffic going to these destination IPs, CIDR nets, hostnames, or aliases, but let it pass directly through the firewall.  
Applies only to transparent mode. Separate entries by semi-colons ( ; )

pfSense上で動作しているSquidプロキシサーバー ([squid-cache.org](http://squid-cache.org))に透過プロキシの設定。  
(スクリーンショットはRubicon Communications, LLC.からの許可を得て使用。)

いずれのタイプのプロキシも、アクセスを許可する前にユーザー認証を必要とする形で構成することができます。ユーザーにパスワードを入力させることなくこれを行うために、プロキシは多くの場合、SSOを使用できます。



プロキシ自動構成(PAC)スクリプトは、ユーザーによる介入を必要とせずにクライアントがプロキシ設定を構成するのを可能にします。WPAD (Web Proxy Autodiscovery)プロトコルは、ブラウザがPACファイルの場所を特定できるようにします。これは攻撃ベクトルとなる場合があります。ブラウザが認証を試みる際に、ローカルネットワーク上の悪意あるプロキシを用いてユーザーのハッシュを取得できるからです([nopsec.com/responder-beyond-wpad](http://nopsec.com/responder-beyond-wpad))。

## リバースプロキシサーバー

リバースプロキシサーバーは、プロトコル固有のインバウンドトラフィックのための機能を提供します。セキュリティ上の目的のために、外部ホストをWebサーバー、電子メールサーバー、VoIPサーバーなどのアプリケーションサーバーに直接接続できないようにしたいことがあります。その代わりに、リバースプロキシをネットワークエッジにデプロイし、パブリックネットワーク（インターネット）からのクライアントリクエストを傍受するよう構成することができます。プロキシはフィルタリングルールを適用し、それが許可されると、DMZ内のアプリケーションサーバー向けの適切なリクエストを生成します。それに加え、リバースプロキシサーバーの中には、各アプリケーション固有のロードバランシング、トラフィックの暗号化、キャッシュの処理をし、アプリケーションサーバーのオーバーヘッドを削減できるものもあります。

## アクセス制御リスト

ファイアウォールのアクセス制御リスト(ACL)は、最低限のアクセスという原則を基に構成されます。これは最小限の特権の原則と同じものであり、有効なネットワークサービスを運用するのに必要な最低限のトラフィックのみを許可し、それ以上は許可しないというものです。ファイアウォールのACL内のルールは上から下へと処理されます。トラフィックがルールの1つにマッチしていると、通過を許可されます。結果として、最も具体的なルールが1番上に置かれることになります。通常、最後のデフォルトルールはルールにマッチしなかったトラフィックをブロックするものです（**暗黙の拒否**）。ファイアウォールにデフォルトの「暗黙の拒否」ルールがない場合、「明示的にすべてを拒否する」というルールをACLの最後に手作業で加えることができます。

The screenshot shows the pfSense Firewall Rules configuration interface. The top navigation bar includes 'Firewall / Rules / WAN'. Below the navigation is a toolbar with 'Floating', 'WAN' (selected), and 'LAN' buttons. The main area is titled 'Rules (Drag to Change Order)' and lists the following rules:

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input checked="" type="checkbox"/> 0 / 0 B	*	Reserved Not assigned by IANA	*	*	*	*	*		Block bogon networks	
<input type="checkbox"/> <input checked="" type="checkbox"/> 0 / 0 B	IPv4 TCP	! 192.168.2.0/24	*	*	*	*	none			
<input type="checkbox"/> <input checked="" type="checkbox"/> 0 / 0 B	IPv4 TCP	*	*	80 (HTTP)	*	*	none			
<input type="checkbox"/> <input checked="" type="checkbox"/> 0 / 0 B	IPv4 TCP	*	*	443 (HTTPS)	*	*	none			
<input type="checkbox"/> <input checked="" type="checkbox"/> 0 / 0 B	IPv4 TCP	*	*	25 (SMTP)	*	*	none			

At the bottom of the interface are buttons for 'Add' (with up and down arrows), 'Delete', 'Save', and 'Separator'.

pfSense上で構成されたファイアウォールルールセットのサンプル。このルールセットはボゴンネットワーク (bogon networks: 本来インターネット上で使われることのないIPアドレス) と、特定のプライベートアドレス範囲からのすべてのトラフィックをブロックしますが、それ以外のあらゆるソースからのHTTP、HTTPS、またはSMTPトラフィックをすべて許可します。(スクリーンショットはRubicon Communications, LLCからの許可を得て使用。)

それぞれのルールは、しばしばタプルと呼ばれるいくつかのパラメータを基に、トラフィックをブロックするか拒否するかを規定することができます。各ルールをデータベースの1つの行として考えると、タプルは列にあたります。例えば前のスクリーンショットでは、タプルとしてProtocol、Source (address)、(Source) Port、Destination (address)、(Destination) Portがあります。

最も単純なパケットフィルタリングファイアウォールであっても、安全に構成するために複雑になる場合があります。フィルタリングのルールセットが何を行うべきかを説明する、明確に記述されたポリシーを作成すると共に、セットアップしたACLが意図した通りに機能することを保証するために、その構成を可能な限りテストすることが重要です。また、ACLに加えた変更をテストして文書化することも必要です。その他の基本的な原則には以下のものがあります。

- （なりすましたことが明らか）内部またはプライネートIPアドレスからのリクエストをブロックする。
- ICMP、DHCP、またはルーティングプロトコルのトラフィックなど、ローカルネットワークレベルでのみ機能すべきプロトコルからのリクエストをブロックする。
- ペネトレーションテストを行って構成がセキュアであることを確かめる。アクセスの試みをログに記録し、そのログを監視して疑わしいアクティビティを突き止める。
- 通常の手順を踏んで、ファイアウォールが動作しているハードウェアの安全を確保すると共に、管理インターフェイスを活用する。

## ネットワークアドレス変換

**ネットワークアドレス変換(NAT)**は、インターネットアクセスを必要とするホストのために、IPアドレスの枯渇を解決する手段として開発されました。通常、プライベートネットワークはプライベートアドレス指定スキームを使用して、各ホストにIPアドレスを割り当てています。それらのアドレスは、RFC 1918 ([tools.ietf.org/html/rfc1918](http://tools.ietf.org/html/rfc1918))で定義されたアドレスのプールの1つから、インターネット上でルーティング不可能なアドレスとして割り当てられます。

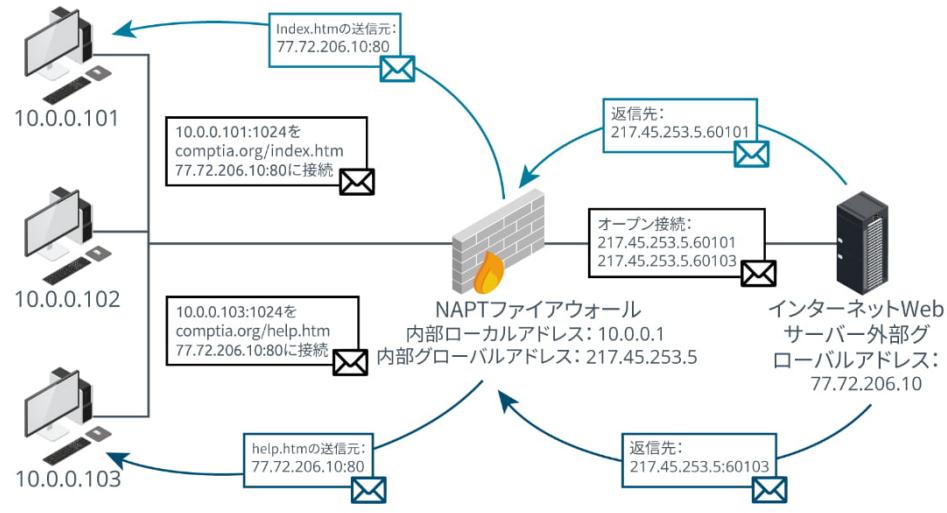
- 10.0.0.0 ~ 10.255.255.255 (クラスAプライベートアドレス範囲)

- 172.16.0.0 ~ 172.31.255.255 (クラスBプライベートアドレス範囲)
- 192.168.0.0 ~ 192.168.255.255 (クラスCプライベートアドレス範囲)

NATゲートウェイは、LAN上のホストが用いるプライベートアドレス指定スキームと、ネットワークエッジ上のルーター、ファイアウォール、またはプロキシサーバーが用いるパブリックアドレス付与スキームとの間で変換を行うサービスです。ネットワークエッジ上の明確に定義されたポイントでイングレス (ingress: 機器への入力) およびイーグレス (egress: 機器からの出力) トラフィックを管理できるという意味で、NATはセキュリティをもたらしますが、フィルタリング機能を実行するものではないと認識することが重要です。

NATにはいくつかのタイプがあります。

- 静的および動的ソースNAT – プライベート（「inside local」と呼ばれる）ネットワークアドレスと、パブリック（「inside global」と呼ばれる）アドレスとの間で1対1のマッピングを行います。これらのマッピングは静的に、または動的に割り当てることができます。
- NATオーバーロード/ネットワークアドレスポート変換(NAPT)/**ポートアドレス変換(PAT)** – 複数のプライベートIPアドレスを単一のパブリックアドレスにマッピングする手段を提供します。例えば、2つのホスト（192.168.0.101と192.168.0.103）が同時にWeb接続を開始します。NAPTサービスはこれらのリクエスト（192.168.0.101:1024と192.168.0.103:1024）に対し、2つの新しいポートマッピングを作成します。その後、プライベートIPをパブリックIPに置き換え、リクエストをパブリックインターネットに転送します。さらに、これらのポートを用いて返されたすべてのトラフィックに対して逆マッピングを行い、元のIPアドレスとポート番号を挿入して、内部ホストにパケットを転送します。



NATオーバーロード (画像提供: © 123RF.com)

- **宛先NAT/ポート転送** – ルーターのパブリックアドレスを用いてWebサービスを公開しますが、到着するリクエストを別のIPに転送します。ポート転送とは、ルーターがインターネットから特定のアプリケーション（例：HTTP/ポート80）向けのリクエストを受け取り、DMZまたはLAN上の指定されたホストとポートに送信することです。

The screenshot shows the 'Edit Redirect Entry' configuration page for a port forward rule. The rule is currently disabled. It specifies the WAN interface, TCP protocol, and a source port range from Any to Any. The destination is set to 10.1.0.10 on port HTTP. A descriptive note states: 'Publish web server'.

*pfSense*ファイアウォールアプライアンス上でポート転送を構成する – このルールは、アプライアンスの WANインターフェイス上で受信されたすべてのHTTPトラフィックを、LAN上の10.1.0.10ホストに転送します。  
(スクリーンショットは [pfsense.org](http://pfsense.org)からの許可を得て使用。)



より大規模なIPv6空間は、NATを不要にする良い例です。ホストはリンクローカルアドレスを用いて隣接するノードに接続できますが、ルーティングされたすべてのトラフィックはグローバルに一意のアドレスを使用する必要があります。IPv6において、どのホストとネットワークが到達可能であるかを管理するのは、ルーティングポリシーとファイアウォールフィルタリングです。とは言え、ネットワークエンジニアのフレックスの変換(NPTv6)や、IPv6アドレス同士(NAT66)またはIPv6アドレスとIPv4アドレス (NAT64とNAT46) の変換を行うメカニズムがあります。

## 仮想ファイアウォール

通常、仮想ファイアウォールはデータセンターやクラウドサービスの内部にデプロイされます。仮想ファイアウォールは、次の3つの方法で実装できます。

- ハイパー-/バイザ型 – フィルタリング機能がハイパー-/バイザまたはクラウドプロビジョニングソールに組み込まれていることを意味します。クラウドのWebアプリまたはアプリケーションプログラミングインターフェイス(API)を使用して、仮想ホストまたは仮想ネットワークに入りするトラフィック向けのアクセス制御リスト(ACL)を記述することができます。
- 仮想アプライアンス – WindowsやLinuxのゲストOSをデプロイするときと同じやり方で、仮想化を用いてベンダーのファイアウォールアプライアンスインスタンスをデプロイすることを指します。
- マルチコンテキスト – 複数の仮想ファイアウォールインスタンスが1つのハードウェアファイアウォールアプライアンス上で動作していることを指します。それぞれのコンテキストには個別のインターフェイスがあり、個別のフィルタリング機能を実行できます。

それらはゾーンベースのルーティングやフィルタリングを行う「通常の」ファイアウォールと同じようにデプロイできますが、仮想ファイアウォールの最も重要な役割は、イーストウェストのセキュリティと、ゼロトラストマイクロセグメンテーション設計のパラダイムをサポートすることです。トラフィックをファイアウォールアプライアンスにルーティングして戻すことを必要とせず、ホストからホストへ、または仮想ネットワーク間を行き来するトラフィックを検査することができます。

## オープンソースファイアウォールとプロプライエタリファイアウォール

ベンダー選定の際に暗黙の信頼に頼ることができない高セキュリティ環境では、ソースコードを検査する機能が必須となります。アプライアンス型、ソフトウェア型、仮想ファイアウォールの基盤となるコードは、オープンソースまたはプロプライエタリ、あるいはその中間として開発されます。

- 完全にプロプライエタリ – Cisco ASA、Juniper JunOS、PaloAlto PANOS、または BarracudaのWindowsベースアプライアンスなど、プロプライエタリとして実装されます。
- ほとんどプロプライエタリ – Linux kernelから開発されますが、プロプライエタリ機能が追加されています。例としてCheck Point IPSO、FortiGate FortiOS、Sonicwallがあります。GPLソースから開発されたすべてのコードが利用可能なはずですが、一般的に言って、これらの製品はベンダーとの商用契約から切り離して使用することができません。
- 完全オープンソース – これらはベンダーと無関係に使用できますが、通常ベンダーは商用アプライアンスとサポート契約も用意しています。例としてpfSenseとSmoothwallがあります。

コアアプライアンスのコードに加え、デプロイにあたって自身でインストールするか、またはサポートを受けるかどうかを決定するにあたっては、サポートへのアクセス、アップデートの可用性、シグネチャや脅威のフィードなど、サブスクリプションベース機能へのアクセスを考慮する必要があります。

# レビュー アク ティビティ：

## ファイアウォールとプロキシサーバー

次の質問にお答えください。

1. 次の記述は正しいですか、誤りですか？プロトコルスタックの最上層でデータを保護する際、アプリケーション型ファイアウォールには基本的なパケットフィルタリング機能が存在しない。
2. ホスト型のパーソナルソフトウェアファイアウォールと、ネットワークファイアウォールアプライアンスを区別するものは何ですか？
3. 次の記述は正しいですか、誤りですか？非透過プロキシをデプロイする際は、プロキシのアドレスとポートでクライアントを構成する必要がある。
4. 通常、ファイアウォールのデフォルトルールの目的は何ですか？
5. 次の記述は正しいですか、誤りですか？静的NATとは、単一のパブリック/外部IPアドレスを単一のプライベート/内部IPアドレスにマッピングすることを意味する。

# トピック10B

## ネットワークセキュリティ監視を実装する



### 対象試験範囲

3.3 与えられたシナリオに基づいて、セキュアなネットワークデザインを実装することができる。

侵入検知・防止システムは成熟したセキュリティ技術であり、企業ネットワークを監視するために幅広くデプロイされています。あなたが分析することになる監視データとアラートデータの大部分はこれらのシステムから生じるので、ネットワーク内の適切な場所にそれらをインストールし、正しく構成できることが重要です。

### ネットワーク型侵入検知システム

**侵入検知システム(IDS)**はソフトウェアツールを使用して、ネットワークトラフィックやシステムログおよびアプリケーションのログをリアルタイムで分析する手段です。**ネットワーク型IDS(NIDS)**は、センサーと呼ばれるパケットスニッファを介してトラフィックをキャプチャします。そのパケットを分析して、悪意のあるトラフィックを特定し、コンソールまたはダッシュボードにアラートを表示します。

**Snort** ([snort.org](http://snort.org))、Suricata ([suricata-ids.org](http://suricata-ids.org))、Zeek/Bro ([zeek.org](http://zeek.org))などのNIDSはパッシブ検知を実行します。トラフィックが検知シグネチャにマッチするとアラートを発するかログエントリを生成しますが、ソースホストをブロックすることはありません。このタイプのパッシブセンサーはトラフィックを遅らせることがなく、脅威アクターによって探知されることもありません。また監視しているネットワークセグメントのIPアドレスは有していません。

NIDSはホストとアプリケーションを識別してログに記録し、攻撃シグネチャ、パスワード推測の試み、ポートスキャン、ワーム、バックドアアプリケーション、悪意を持って生成されたパケットやセッション、ポリシー違反（例えば許可されていないポートまたはIPアドレスなど）を検知するために使用されます。ログの分析結果を用いてファイアウォールのルールセットを調整したり、疑わしいホストやプロセスをネットワークから除去またはブロックしたり、識別した脅威を軽減する追加のセキュリティ管理をデプロイしたりすることができます。

The screenshot shows the Kibana Discover interface. On the left is a sidebar with navigation links: Discover, Visualize, Dashboard, Timelion, Dev Tools, Management, Squert, Logout, and Collapse. The main area has tabs for Time and \_source. The Time tab shows a single event from March 16th, 2020, at 13:57:40.947. The \_source tab displays the raw log data:

```

destination_ip: 195.2.253.92 message:
[1:2003380:12] ET USER_AGENTS Suspicious User-Agent -
Possible Trojan Downloader (ver18/ver19 etc)
[Classification: A Network Trojan was detected]
[Priority: 1]: <siem-eth1-1> {TCP} 192.168.3.35:1037

```

Below the log are two tabs: Table and JSON. The JSON tab is selected, showing the document structure. At the top right are buttons for "View surrounding documents" and "View single document". Below the JSON tab, there are search and filter controls for each field.

Security Onion上のKibanaアプリ内にあるSnortによって生成された侵入検知アラートの表示。  
(Security Onion [securityonion.net](http://securityonion.net)のスクリーンショット)

## ポートミラーとTAP

通常パケットキャプチャセンサーは、ファイアウォールの内側または特に重要なサーバーの近くに設置されます。これは一般的に、ファイアウォールの通過に成功した悪意のあるトラフィックを識別することが狙いです。単一のIDSが大量のログデータやアラートデータを生み出すこともあるため、これらのデータを適切に管理するリソースが必要で、ネットワークのいたるところにさまざまなセンサーを単に設置するだけでは済みません。ネットワークサイズやリソースに応じて、主要な資産またはネットワーク経路の監視のために配置するセンサーの数は1個または数個です。

ネットワーク内の適切な場所にセンサーを接続するにあたっては、主な選択肢が3つあります。

- SPAN (switched port analyzer)/ミラーポート** – これは、センサーをスイッチ上の特別に設定されたポートに接続し、指名されたアクセスポート（または他のすべてのポート）宛てのフレームのコピーを受信するようにします。この方法は完全に信頼できるわけではありません。エラーがあるフレームはミラーリングされず、また負荷が重いとフレームがドロップされることもあります。
- パッシブテストアクセスポイント(TAP)** – これは箱形のデバイスで、ネットワークの入出力ケーブルを接続するポートと、ケーブルからの信号を物理的に監視ポートにコピーするコイルまたは光学スプリッタを備えています。銅線ケーブル用と光ファイバーケーブル用のタイプがあります。SPANと違って論理的な判断は行われないため、破損や不正の有無にかかわらず監視ポートはすべてのフレームを受信し、コピー作業が負荷の影響を受けることはありません。
- アクティブTAP** – これは電源に接続されるデバイスで、状況によって必要となる信号の再生を行います（こちらも銅線用と光ファイバーケーブル用の両タイプがあります）。銅線を介して送信されたギガビット単位の信号は、パッシブTAPで監視するにはあまりに複雑であり、またファイバーリンクの中には、光学スプリッティングによって悪影響を受けるタイプもあります。このTAPは電源を得て作動するので、電源喪失の際にリンクの単一障害点となります。アクティブTAPをデプロイする際は、内部バッテリーを備えたモデルを使用するか、UPSに接続することが重要です。

通常、TAPは2つのストリームを出し、全二重リンク（チャネルの1つはアップストリーム用、もう1つはダウンストリーム用）を監視します。それとは別にアグリゲーションTAPがあり、単一のチャネルへのストリームを再構築しますが、非常に負荷が重い状況ではフレームをドロップすることがあります。

## ネットワーク型侵入防止システム

IDSのパッシブ機能と比較した場合、**侵入防止システム(IPS)**は、それとマッチするすべてのネットワーク脅威に対してアクティブに反応することができます。典型的な防止手段の1つに、TCPセッションを終了させ、攻撃側のホストにTCPリセットパケットを送信するというものがあります。別の選択肢として、IPSが一時的なフィルターをファイアウォールに適用し、脅威アクターのIPアドレスをブロックする（回避する）というものがあります。その他の高度な手段には、攻撃側ホストへの帯域幅を絞る、複雑なファイアウォールフィルターを適用する、さらには疑わしいパケットを改変して無害にすることといったものがあります。最後に、アプライアンスがスクリプトやサードパーティのプログラムを実行し、IPSソフトウェア自体がサポートしていない他のアクションを行うこともあります。

IPSの中には、インラインのワイヤースピード・アンチウイルススキャンを行うものもあります。そのルールセットを構成することで、URLをブロックする、キーワードごとのブロックリストまたは許可リストを適用する、もしくは時間ベースのアクセス制限を適用するなど、ユーザーコンテンツのフィルタリングを提供できます。

IPSアプライアンスはファイアウォールのように、2つのネットワークゾーンを隔てる境界に設置します。プロキシサーバーの場合と同じく、アプライアンスはネットワークと「インライン」であり、これはすべてのトラフィックがそれらを通過することを意味します（また、耐障害メカニズムがない場合は単一障害点となります）。このことは、そうしたアプライアンスは高帯域に対応し、ネットワークの遅延を避けるために各パケットを極めて迅速に処理できる必要がある、ということを意味します。

## シグネチャ型検知

IDSにおいて、疑わしいトラフィックを識別する目的で、センサーによってキャプチャされたトラフィックのスキャンと解釈を行う構成部品を分析エンジンと言います。分析エンジンはある特定のイベントをどのように分類するかを決定します。典型的な選択肢としては無視、ログへの記録のみ、アラート、およびブロック(IPS)があります。分析エンジンは、その意志決定プロセスを推進するために用いられる、一連のルールでプログラムされます。ルールセットを構成するいくつかの方法があります。

**シグネチャ型検知**（またはパターンマッチング）は、エンジンに攻撃パターンまたはシグネチャのデータベースを搭載することを意味します。トラフィックがパターンにマッチすると、エンジンはインシデントを生成します。

```
GNU nano 2.5.3          File: downloaded.rules

#
# ----- Begin ET-emerging-activex Rules Category ----- #
#
# -- Begin GID:1 Based Rules -- #

#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Internet Explorer Plugin.ocx Heap Overflow$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX winhlp32 ActiveX control attack - phase 1$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX winhlp32 ActiveX control attack - phase 2$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX winhlp32 ActiveX control attack - phase 3$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX MciWndx ActiveX Control"; flow:from_serv$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX COM Object Instantiation Memory Corrupti$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX JuniperSetup Control Buffer Overflow"; f$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Danim.dll and Dxtnsft.dll COM Objects"; $"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Wmmzfxa.dll Control Buffer Overflow"; f$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft COM Object Instantiation Mem$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft Multimedia Controls - ActiveX$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft Multimedia Controls - ActiveX$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft Multimedia Controls - ActiveX$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft WMIScriptutils.WMIOBJECTBroker$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft VsmIDE.DTE object call CSLID";$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft DExplore.AppObj.8.0 object cal$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft VisualStudio.DTE.8.0 object ca$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft Microsoft.DbgClr.DTE.8.0 objec$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft VsaIDE.DTE object call CSLID";$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft Business Object Factory object$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft Outlook Data Object object cal$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Microsoft Outlook.Application object cal$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX ACTIVEX Possible Microsoft IE Install En$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Possible Microsoft IE Install Engine Ins$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Possible Microsoft IE Shell.Application $"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX ACTIVEX Possible Microsoft IE Shell.App$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX NCTAudioFile2 ActiveX SetFormatLikeSampi$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Possible Microsoft Internet Explorer ADO$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Sony ImageStation (SonyISUpload.cab 1.0.$"
#alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET ACTIVEX Citrix Presentation Server Client WFICA.$"

[ Read 27185 lines (Warning: No write permission) ]
```

オープンソースの*Emerging Threats*コミュニティフィードが提供するSnortルールファイル。

侵入検知を支えるシグネチャとルール（しばしばプラグインまたはフィードと呼ばれます）は定期的に更新し、最新の脅威タイプに対する保護を行うようにする必要があります。商用ソフトウェアで更新入手するには、有償のサブスクリプションが必要となります。理想的にはHTTPSなどのセキュアな接続手段を用いて、正当なレポジトリからのみ更新を行うよう、ソフトウェアを構成することが重要です。

## 振る舞い検知と異常検知

**振る舞い検知**は、エンジンが「正常な」ベースラインのトラフィックまたはイベントを認識できるようになっていることを意味します。このベースラインからの逸脱（定義済みの許容レベルから外れたもの）があれば、インシデントが生成されます。この考え方方は、ゼロデイ攻撃、インサイダーの脅威、および単一のシグネチャがあるその他の悪意あるアクティビティを、ソフトウェアが識別できるというものです。

歴史的に見ると、このタイプの検知はネットワーク動作異常検出(NBAD)製品によって提供されていました。NBADエンジンは**ヒューリスティクス**（「経験から学ぶ」の意）を用いることで、正常なベースラインのトラフィックがどのようなものかに関する統計モデルを生成します。また、1日の異なる時間帯のネットワーク使用をモデル化するために、いくつかのプロファイルを作成することもあります。このことは、「正常」な状態まで統計モデルを改良するまでの間に、システムが誤検知（フォールスピジティブ）や検知漏れ（フォールスネガティブ）を発生させることを意味します。**誤検知**とは正当な行動がアラートを発生させることであり、一方の**検知漏れ**とは悪意あるアクティビティにアラートが発せられないことです。

NBAD製品は比較的洗練されていませんが、最近の製品では機械学習が採用されており、より生産的になっています。Gartner社の市場分析([gartner.com/en/documents/3917096/market-guide-for-user-and-entity-behavior-analytics](http://gartner.com/en/documents/3917096/market-guide-for-user-and-entity-behavior-analytics))によって特定された通り、機械学習を活用した振る舞い検知製品は次の2つの一般的なクラスに分かれます。

- ユーザーおよびエンティティの行動分析(UEBA) – これらの製品は複数の侵入検知やログソースからのインジケーターをスキャンすることで、異常を識別します。多くの場合、セキュリティ情報イベント管理(SIEM)プラットフォームが組み込まれています。
- ネットワークトラフィック分析(NTA) – これらの製品は、複数のネットワークやログのデータソースではなく、ネットワークストリームにのみ分析手法を適用するという点で、IDSやNBADにより近い存在です。

多くの場合、振る舞い検知や異常検知は、（エンジンが異常な行動を検知するという点で）同じことを意味するものとして捉えられます。また異常検知は、プロトコルの不正使用性を特に探すという意味として捉えられることもあります。例えば、エンジンがパケットヘッダーまたはあるセッション内のパケットのやり取りをRFC標準に照らしてチェックし、それが厳格なRFC適合性からの逸脱があれば、アラートを生成することができます。

## 次世代ファイアウォールとコンテンツフィルター

侵入検知は本来スタンドアロンのソフトウェアまたはアプライアンスとして生み出されたのですが、その機能はすぐさま新世代のファイアウォールに組み込まれました。オリジナルの**次世代ファイアウォール(NGFW)**は、Palo Alto社によって2010年という早い時期にリリースされました。この製品は、アプリケーション対応のフィルタリングと、ユーザー・アカウントベースのフィルタリング機能、および侵入防止システム(IPS)として機能する能力を組み合わせたものです。このアプローチはすぐさま競合製品に取り入れられました。その後の世代のファイアウォールには、クラウド検査などの機能が追加され、さまざまなセキュリティ技術を組み込んでいます。

## 統合脅威管理(UTM)

**統合脅威管理(UTM)**は、さまざまなタイプのセキュリティ管理（ファイアウォール、アンチマルウェア、ネットワーク侵入防止、スパムフィルタリング、コンテンツフィルタリング、データ喪失の防止、VPN、クラウドアクセスゲートウェイ）を1つのアプライアンスに集中させたセキュリティ製品を指します。このことは、単一のコンソールからそれらのコントロールを監視・管理できることを意味します。ですが、UTMにも欠点はあります。防御が単一のシステムに統合されることを意味します。

とで、ネットワーク全体に影響を及ぼす单一障害点が生じる可能性があります。個別のセキュリティシステムなら、たとえ機能停止しても、特定の攻撃手段による侵害のみに留まるでしょう。さらにUTMシステムでは、ネットワークアクティビティが多すぎると、レイテンシーの問題が発生する可能性があります。またUTMは、単一の専用セキュリティ機能を有するソフトウェアやデバイスに比べて、良好に機能しないこともあります。



ある意味で、NGFWとUTMはマーケティング用語に過ぎません。UTMは、「何でもできる」タンキー（すぐに使える）ソリューションとみなされる一方、NGFWはエンタープライズ製品であり、機能は少なく（つまりモジュール化が進んでいる）、構成もより複雑ですが、パフォーマンスは優れています。実装の決定をNGFWかUTMかの選択として提示しようとするのではなく、特定の製品機能に焦点を当てる方が役立ちます。

## コンテンツ/URLフィルター

ファイアウォールは高負荷に耐えなければならず、過負荷はレイテンシーを増大させたり、さらには機能停止を引き起こすこともあります。アプリケーション対応のNGFWやUTMソリューションは複雑性が高く、エッジデバイスとしての適格性に欠けることがあります。高い機密性と完全性を提供する一方、低いスループットによって可用性が減少するからです。これに対する解決策の1つに、サーバートラフィック向けのセキュリティソリューションを、ユーザートラフィック向けのものと別に扱う、というものがあります。ユーザートラフィックとは、ローカルネットワーククライアントによって行われるWebブラウジング、ソーシャルネットワーク、電子メール、動画/VoIP接続を指します。

結果として、ステートフルまたはNGFWファイアウォールがアプリケーションサーバートラフィック向けにデプロイされる場合、ユーザートラフィックをフィルタリングする役割は、別のアプライアンスまたはプロキシホストによって実行されます。コンテンツフィルターは、コンテンツブラックリストに載っているUniform Resource Locator (URL)をブロックする、またはブラウジングに時間ベースの制限をかけるなど、ユーザーに焦点を当てた多数のフィルタリングルールを適用するよう設計されます。現在のことろ、コンテンツフィルターは通常、セキュアWebゲートウェイ(SWG)と呼ばれる製品群として実装されています。SWGはフィルタリングに加えて脅威分析を行い、また多くの場合データ損失防止(DLP)とクラウドアクセスセキュリティプロトコル(CASB)の機能を組み込むことで、マルウェアのコマンドやコントロール、データ流出など、ありとあらゆる未承認の外部漏出の脅威から保護しています。

## ホスト型侵入検知システム

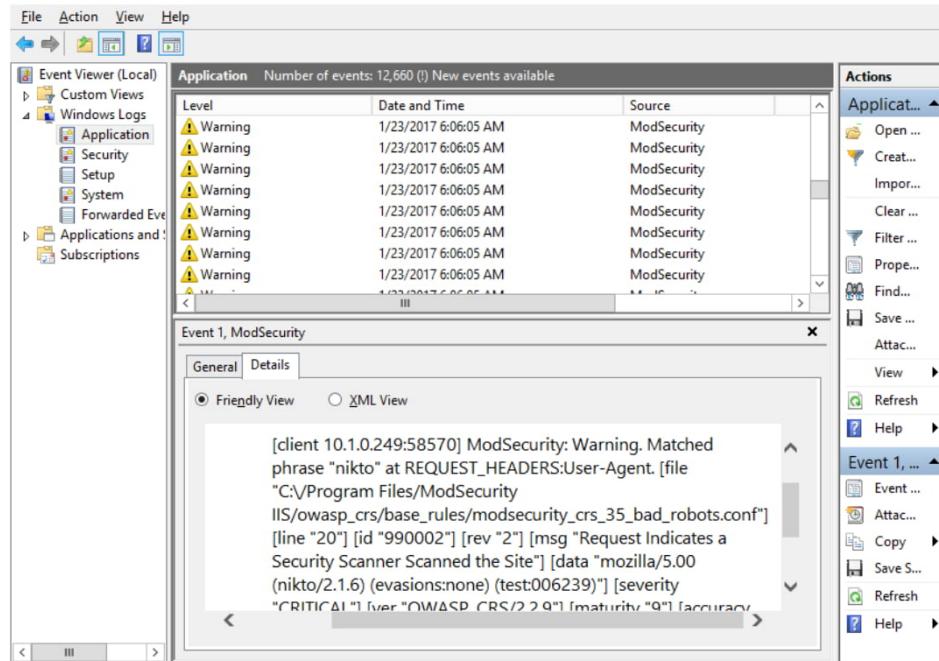
**ホスト型IDS (HIDS)**は、サーバー、ルーター、またはファイアウォールなど、単一のホストからの情報をキャプチャします。各クライアントワークステーション上にHIDSを構成している組織もあります。HIDSはさまざまな機能を持つさまざまな形式で利用できます。核となる能力はログファイルをキャプチャし分析することですが、より洗練されたシステムではOSカーネルファイルの監視、ポートとネットワークインターフェイスの監視、またはHTTPやFTPなど特定のアプリケーションが生成したデータとログの処理も可能になっています。

HIDSソフトウェアは、アンチマルウェアスキャナーと同様のアウトプットを生成します。ソフトウェアが脅威を検知しても、そのイベントをログに記録するか、アラートを表示するだけのこともあります。そのログは、どのプロセスがイベントを開始したかや、ホスト上のどのリソースが影響を受けたかを示すはずです。そのログを用いることで、疑わしいプロセスが許可されたものか、またはホストから除去すべきものかどうかを調査することができます。

HIDSの中核をなす機能の1つに、**ファイル完全性の監視(FIM)**があります。これはスタンダードアロン機能として実装されることもあります。ソフトウェアが正当なソース（Windowsの場合はコード署名を用いているソース、Linuxの場合はセキュアなリポジトリを用いているソース）からインストールされた際、OSのパッケージマネージャーは1つ1つの実行可能ファイルのシグネチャまたはフィンガープリントをチェックし、問題があればユーザーに通知します。FIMソフトウェアは主要なシステムファイルを監査し、それらが許可されたバージョンとマッチしていることを確認します。WindowsにおいてはWindows File Protectionサービスが自動的に実行され、System File Checker (sfc)ツールを手動で使用してOSシステムファイルを検証することができます。Tripwire ([tripwire.com](http://tripwire.com))とOSSEC ([ossec.net](http://ossec.net))は、幅広い範囲のアプリケーションを保護するオプションを備えたマルチプラットフォームの一例です。

## Webアプリケーションファイアウォール

Webアプリケーションファイアウォール(WAF)は、Webサーバーとそのバックエンドデータベースで実行されるソフトウェアを、コードインジェクション攻撃やDoS攻撃から保護するために特別に設計されたファイアウォールです。WAFはアプリケーション対応の処理ルールを使用することでトラフィックをフィルターし、各アプリケーション固有の侵入検知を実行します。WAFは既知の攻撃のシグチャと共にプログラムされ、パターンマッチングを用いて疑わしいコードを含むリクエストをブロックします。WAFからの出力はログに記述され、そのログを検査することで、Webアプリケーションがどの脅威の標的になり得るかを判断することができます。



このIISサーバーにModSecurity WAFをインストールすることで、スキャニングの試みが検知され、Applicationイベントとしてログに記録されました。見て分かるように、デフォルトのルールセットは多数のイベントを生成します。(スクリーンショットはMicrosoftからの許可を得て使用。)

WAFはアプライアンスとしてデプロイされることもありますが、Webサーバープラットフォーム向けのプラグインソフトウェアとしてデプロイされることもあります。WAF製品の例には次のようなものがあります。

- ModSecurity ([modsecurity.org](http://modsecurity.org))はTrustwaveがスポンサーとなっているオープンソースWAFで、Apache、nginx、IIS用です。
- NAXSI ([github.com/nbs-system/naxsi](https://github.com/nbs-system/naxsi))は、nginx Webサーバーソフトウェア用のオープンソースモジュールです。
- Imperva ([imperva.com](http://imperva.com))は商用Webセキュリティ製品で、特にデータセンターに焦点を当てています。Impervaは同社のSecureSphereアプライアンスを通じてWAF、DDoS、データベースのセキュリティを市場展開しています。

# レビュー アク ティビティ： ネットワークセキュリティ監視

次の質問にお答えください。

1. 同じスイッチ上のホスト間を行き来しているトラフィックを監視する最善の選択肢は何ですか？
2. シグネチャ型の監視ソフトウェアでは、どのような種類のメンテナンスを行う必要がありますか？
3. 振る舞い検知で侵入防止システムをデプロイする場合、主なリスクは何ですか？
4. Windowsのシステムファイルがファイル完全性のチェックに失敗した場合、マルウェアの感染を疑うべきですか？
5. WAFとは何ですか？

# トピック10C

## SIEMの使用を要約する



### 対象試験範囲

- 1.7 セキュリティ評価で使用する手法を要約することができる。
- 3.3 与えられたシナリオに基づいて、セキュアなネットワークデザインを実装することができる。
- 4.1 与えられたシナリオに基づいて、適切なツールを利用して組織のセキュリティを評価することができる。

ネットワーク、ホスト、データを保護するためにデプロイできる、さまざまなタイプのセキュリティ管理があります。それら管理のすべてに共通していることの1つに、ログデータとアラートを生成するという点があります。そのアウトプットをレビューすることは、情報セキュリティ管理における主要な課題の1つです。あなたはセキュリティのプロフェッショナルとして、システムの説明、インストール、構成を行い、ロギングとイベントを管理できるようになる必要があります。

### 監視サービス

セキュリティ評価とインシデント対応はいずれも、ホストやネットワークの状態インジケーター、および監査情報のリアルタイムの監視を必要とします。

### パケットキャプチャ

ネットワークセンサー / スニッファからキャプチャされたデータと、ネットフローソースからキャプチャされたデータはいずれも、帯域幅やプロトコルの使用状況に関する統計上の要約を提供し、詳細なフレーム分析を可能にします。

### ネットワークモニター

ネットワークトラフィックの監視と異なり、**ネットワークモニター**はスイッチ、アクセスポイント、ルーター、ファイアウォール、サーバーなど、ネットワークアプライアンスに関する情報を収集します。これは、CPU/メモリの負荷状況、ステートテーブル、ディスク容量、ファンの回転速度/温度、ネットワーキングの使用率/エラー統計などを監視するために使用されます。もう1つの重要な機能として、可用性を示すハートビートメッセージがあります。このデータは、SNMP (Simple Network Management Protocol : 簡易ネットワーク管理プロトコル) またはプロプライエタリな管理システムを用いて収集されることがあります。ネットワーク監視は、可用性を高めるだけでなく、ある種の攻撃を示す異常な状態を発見することもあります。

### ログ

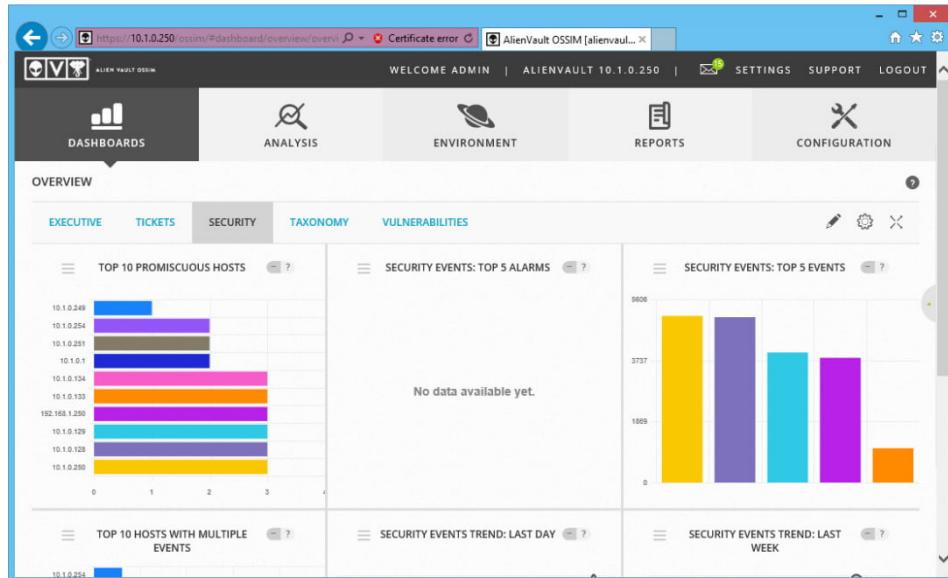
ログはセキュリティ情報の最も価値あるソースの1つです。システムログを用いて可用性の問題を診断することができます。セキュリティログは、リソースまたは特権の許可された使用と不正な使用を記録することができます。ログはアクションの監査記録として機能すると共に、(定期的に監視されている場合は) 侵入の試みを警告します。ログのレビューはセキュリティ保護にとって不可欠な手順の一部です。重大なインシデントの後にログを参照するだけでは、脅威と脆弱性を初期のうちに突き止め、事前にに対応する機会を失っています。



通常、ログは1つのアクションをある特定のユーザーに関連付けます。ユーザーがログオンの詳細をシェアしないことが重要なのは、それが理由の1つです。ユーザー アカウントが侵害されると、ログ内のイベントを実際の脅威アクターに結びつける方法がなくなります。

## セキュリティ情報イベント管理

セキュリティデータ入力の管理を支援し、レポートとアラートの提供を目的として設計したソフトウェアは、**セキュリティ情報イベント管理(Security Information and Event Management, SIEM)**と表現されることがあります。SIEMツールの核となる機能は、トラフィックのデータとログを集約することです。WindowsおよびLinuxベースのホストからのログに加え、スイッチ、ルーター、ファイアウォール、IDSセンサー、脆弱性スキャナー、マルウェアスキャナー、データ損失防止(DLP)システム、データベースが含まれることもあります。



**OSSIM SIEMダッシュボード** – 設定可能なダッシュボードは、ネットワークのセキュリティ指標のハイレベルなステータスの表示を提供します。(スクリーンショットはAT&T Cybersecurityからの許可を得て使用。)

## ログ収集

SIEMが行う最初のタスクは、さまざまなソースからのデータ入力を収集することです。ログ収集には主に3つのタイプがあります。

- エージェント型 – このアプローチでは、各ホストにエージェントサービスをインストールする必要があります。ホストでイベントが発生すると、ロギングデータはそのホストでfiltrating、集約、正規化され、SIEMサーバーに送信されて分析と保存が行われます。
- リスナー / コレクター** – エージェントをインストールするのではなく、syslogやSNMPなどのプロトコルを使用してSIEMサーバーにアップデートを送信するようにホストを設定することができます。それぞれのログ/監視ソースを解析し正規化するプロセスは管理サーバーで実行されます。
- センサー – SIEMはログデータに加え、スニッファからのパケットキャプチャやトラフィックフローのデータを集めることもあります。

*pfSenseセキュリティアプライアンス用のログ解析プラグインを有効化して、ファイアウォールのイベントをSIEMにインポートできるようにします。(スクリーンショットはAT&T Cybersecurityからの許可を得て使用。)*

## ログ集約

収集と違い、集約は異なるソースからのデータを正規化することで、一貫性を保ち、検索可能にすることを指します。SIEMソフトウェアにはコネクターまたはプラグインが備わっており、異なるタイプのシステムからのデータを解釈（または解析）し、ベンダーによる実装の違いに対処します。通常、解析はそれぞれのログファイルフォーマットに合わせた正規表現を用いて行われ、SIEMのレポートと分析ツール内の標準フィールドにマッピングできる属性とコンテンツを識別します。もう1つの重要な機能として、日付/時間帯の違いを単一のタイムラインに標準化することができます。

## 分析とレポートのレビュー

収集と集約がインプットを生み出す一方、SIEMはレポートにも使用できます。SIEMの重要な機能であり、基本的なログ管理との大きな違いは「相関」です。これは、SIEMソフトウェアが、個々のイベントやデータポイント（観測値）をリスクの意味のある指標、すなわち侵害インジケーター（IOC）に結びつけることができるることを意味します。そして相関を利用して、アラートシステムを駆動させることができます。これらのレポートはSIEMダッシュボードから閲覧することができます。

基本的な相関は、単純な「If ... Then」タイプのルールを用いて表現されます。しかし、多くのSIEMソリューションは自動分析の基礎として人工知能（AI）や機械学習を用いています。

## ユーザーおよびエンティティの行動分析

ユーザーおよびエンティティの行動分析（UEBA）ソリューションは、悪意のある行動をベースラインとの比較によって識別するのをサポートします。その名が示す通り、分析ソフトウェアはさまざまなデバイスやクラウドサービスをまたがるユーザー アカウントの行動を追跡します。エンティティはクライアントワークステーションや仮想サーバーインスタンスなどのマシンアカウント、およびモノのインターネット（IoT）デバイスなどの組み込みハードウェアを指します。ベースラインを決定して誤検知を減らすのが複雑であることは、UEBAソリューションがAIや機械学習に強く依存していることを意味します。その例としてMicrosoftのAdvanced Threat Analytics ([docs.microsoft.com/en-us/advanced-threat-analytics/what-is-ata](https://docs.microsoft.com/en-us/advanced-threat-analytics/what-is-ata))やSplunk UEBA ([splunk.com/en\\_us/software/user-behavior-analytics.html](https://splunk.com/en_us/software/user-behavior-analytics.html))があります。

## センチメント分析

機械学習による行動分析の最大の課題の1つは、意図を特定することです。自然言語による発言の文脈や解釈を機械が判断することは、多くの進歩が見られるとはいえ、極めて困難です。この分野における総合的な取り組みは**センチメント分析**またはエモーションAIと呼ばれます。センチメント分析の典型的な使用例として、不満を持った顧客が、ひどいサービスを受けたとTwitterに書き込むといった、ブランド関連の「インシデント」を確かめるためにソーシャルメディアをモニターする、といったことが挙げられます。セキュリティの観点から言えば、脅威インテリジェンスを収集し、攻撃を仕掛けられる前に外部またはインサイダーの脅威を突き止めるべく、センチメント分析を使用できます。

## セキュリティのオーケストレーション、自動化、レスポンス

SOAR (Security Orchestration, Automation, and Response)は、アナリストの対処能力をはるかに超えるアラートの量という問題へのソリューションとして設計されたものです。SOARはスタンドアロン技術として実装されることもあり、これは次世代SIEMとよく呼ばれます。SOARの基本は、組織が保存しているセキュリティと脅威インテリジェンスをスキャンし、機械学習またはディープラーニングを用いてそれを分析した上で、そのデータを使用して自動化を行い、インシデント対応と脅威ハンティングを推進するワークフローを目指してデータの強化を図ることです。

## ファイル操作

SIEMがログの収集とレビューに関する数多くの機能を自動化できる一方で、Linuxのコマンドラインを用いて手作業でデータを用意しなければならない場合もあります。

### catコマンド

Linuxの**catコマンド**を用いることで、1つまたは複数のファイルの中身を見るることができます。例えば、連続した2つのログファイルの中身全体を見たい場合、次のコマンドを実行します。

```
cat -n access.log access2.log
```

-nスイッチは行番号を追加します。端末にてなく新規ファイルに出力したい場合は、次のコマンドを実行します。

```
cat -n access.log access2.log > access_cat.log
```

### headコマンドとtailコマンド

**headコマンド**と**tailコマンド**は、指定したファイルの最初と最後の10行をそれぞれ出力します。このデフォルト値を調整し、-nスイッチを使用してそれより多い（または少ない）行数を出力することもできます。例えば、次のコマンドはログファイル内の最新のエントリ20件を表示させます。

```
tail /var/log/messages -n 20
```

### loggerコマンド

**loggerコマンド**は、ローカルシステムログまたはリモートsyslogサーバーに入力を書き込みます([linux.die.net/man/1/logger](http://linux.die.net/man/1/logger))。スクリプト内でこのコマンドを使用すればどんなテキスト文字列でも書き込むことができ、また-fオプションを用いて別のファイルの中身を書き込むこともできます。さらに、このコマンドをバッククオート (`) で囲むことにより、コマンドの出力を書き込むこともできます。次のコマンドはローカルマシンの名前をテキスト「up」と共に、10.1.0.242のsyslogサーバーに書き込みます。

```
logger -n 10.1.0.242 `hostname` up
```