

レビュー アク ティビティ： ソーシャルエンジニアリングの手法

次の質問にお答えください。

- ヘルプデスクに電話があり、発信者はeコマースWebサイトに接続して注文ステータスを確認できないと述べています。ユーザー名とパスワードも要請しています。ユーザーは有効な顧客の会社名を示しましたが、顧客データベースに連絡先としてリストされていません。ユーザーは正しい企業コードや顧客IDは知りません。これは、ソーシャルエンジニアリングの試みでしょうか。それともこれは誤認警報でしょうか。
- ある購買部長が、ベンダーのWebサイトで製品のリストを閲覧していたところ、コンピューターにある数千個のファイルがウイルスに感染していることがマルウェア対策ソフトウェアで検知されたことを知らせるウインドウが開きました。公式に見えるウインドウの手順には、ユーザーにリンクをクリックして、こうした感染を取り除くソフトウェアをインストールするべきだと記されています。これはどのような種類のソーシャルエンジニアリングの試みですか。それともこれは誤認警報でしょうか。
- あるCEOは電話で、市場調査データをすぐに自身の電子メールアドレスに転送するよう要求しています。あなたは声でCEOを認識しましたが、正式な要請フォームが記入されておらず、サードパーティの電子メールの使用は禁止されています。CEOは、通常はフォームに記入し、例外は認められるべきではないが、出席しているコンファレンスの円卓会議用にデータが至急必要だと言っています。これはどのような種類のソーシャルエンジニアリングの手法ですか。それともこれは誤認警報でしょうか。
- あなたの会社では、多くの有名クライアントのマーケティングデータとプライベート情報を管理しています。あなたは内定者に対して一般公開日を開催しています。ソーシャルエンジニアリング攻撃の可能性を念頭に置いて、従業員はゲストが社内を案内される際にどのような予防措置を講じる必要がありますか。

トピック4B

マルウェアベースの攻撃のインジケーターを分析する



対象試験範囲

1.2 与えられたシナリオに基づいて、可能性あるインジケーターを分析して攻撃のタイプを特定することができる

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 (Cuckooのみ)

今日コンピューターで最も普及している脅威の1つは、悪意のあるコードです。あなたはセキュリティの専門家として、システムに感染する望ましくないソフトウェアを扱った経験があるでしょう。さまざまなタイプのマルウェアを分類し、感染のサインを特定することで、侵害されたシステムの修復を準備し、あるいはそもそもマルウェアが実行されないようにします。

マルウェア分類

コンピューターネットワークに対して行われる侵入の試みの多くは、悪意のあるソフトウェアまたはマルウェアの使用に依存しています。マルウェアは通常、システム所有者の観点から、何か悪いことをするソフトウェアとして単に定義されます。マルウェアには多くの種類がありますが、厳密に分類されていないため、一部の定義が重複したり、はっきりしていません。トロイの木馬、ウイルス、ワームなど一部のマルウェア分類では、マルウェアによって使用されるベクトルに重点を置いています。ベクトルとは、マルウェアがコンピューター上で実行され、ほかのネットワークホストに拡散する可能性のある方法です。マルウェア分類におけるもう1つの複雑な要因は、そのインストールがユーザーによって期待または許容される度合いで。次のカテゴリでは、ベクトルに従っていくつかのマルウェアを説明しています。

- ウイルスとワーム — これらは最初のマルウェアの種類の一部を表しており、別のプロセスの実行可能コード内に隠されることにより、ユーザーの許可なく拡散します。
- **トロイの木馬** — 正当に見えるソフトウェアのインストーラーパッケージ内に隠されたマルウェアです。このタイプのマルウェアでは、インストールにいかなる種類の許可も求めず、密かに動作するように設計されています。
- **潜在的に望ましくないプログラム(PUP)/潜在的に望ましくないアプリケーション(PUA)** — ユーザーが選択したパッケージと一緒にインストールされたソフトウェア、または新しいコンピューターシステムにバンドルされたソフトウェア。PUPの存在は、トロイの木馬とは異なり、自動的に悪意のあるものとは見なされません。積極的同意または意図的に混乱させるライセンス契約から同意なく、インストールされた可能性があります。このタイプのソフトウェアは、マルウェアではなくグレイウェアと説明される場合があります。

その他の分類は、マルウェアによって提供されるペイロードに基づきます。ペイロードは、ホスト上で単に複製または永続化する以外に、マルウェアによって実行されるアクションです。ペイロード分類の例には、スパイウェア、ルートキット、リモートアクセス型トロイの木馬(RAT)、ランサムウェアがあります。



ベクトル別マルウェア分類。

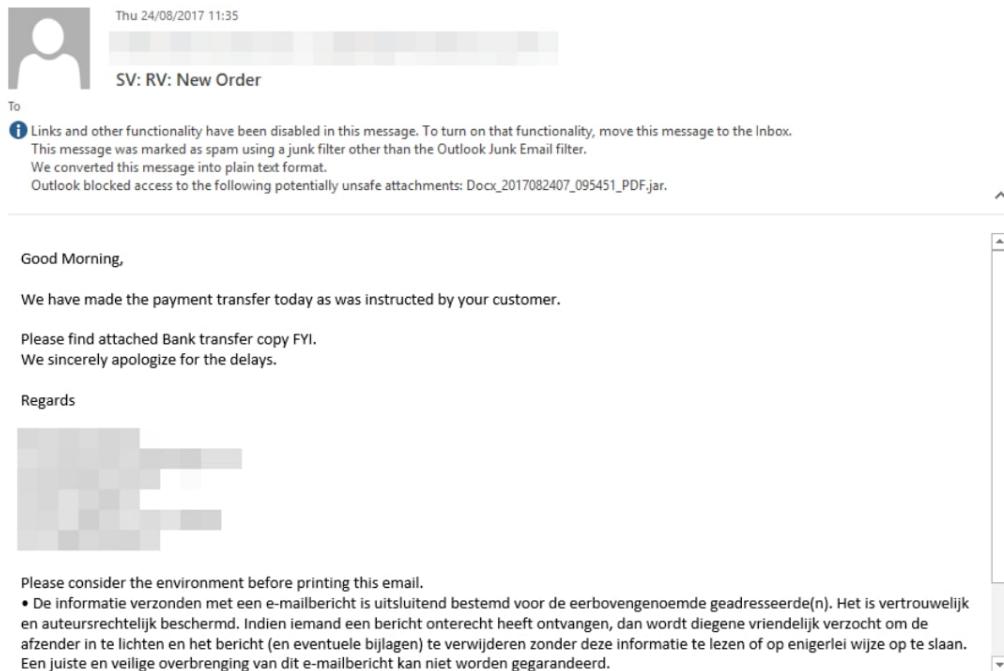
コンピューターウィルス

コンピューター **ウイルス** は通常、実行可能アプリケーションまたはプログラムコードに「感染」することにより、コンピューター間で複製・拡散するように設計されたマルウェアの一種です。ウイルスにはいくつかの異なる種類があり、これは通常、感染するさまざまなファイルやメディアの種類によって分類されます。

- 非常駐/ファイルインフェクター — ウィルスはホストで実行可能なファイル内に格納され、ホストプロセスで実行されます。ウィルスは永続ストレージで別のプロセスイメージに感染し、別のペイロードアクションの実行を試みます。その後、制御をホストプログラムに戻します。
- メモリ常駐 — ホストファイルが実行されると、ウィルスはメモリ内で自身に対する新しいプロセスを作成します。悪意のあるプロセスは、ホストプロセスが終了してもメモリに留まります。
- ブート — ウィルスコードは、ディスクブートセクタや固定ディスクやUSBメディアのパーティションテーブルに書き込まれ、OSの起動時やメディアがコンピューターに接続されたときに、メモリ常駐プロセスとして実行されます。
- スクリプトおよびマクロウイルス — マルウェアでは、PowerShell、Windows Management Instrumentation (WMI)、JavaScript、Visual Basic for Applications (VBA)コードが有効になっているMicrosoft Officeドキュメント、またはJavaScriptが有効になっているPDFドキュメントなどのOSやブラウザ向けのローカルスクリプティングエンジンで利用できるプログラミング機能を使用します。

またマルチパートイト(*multipartite*)という用語は、複数のベクトルを使用するウィルスに使用され、ポリモーフィック(*polymorphic*)はコードを動的に変更または難読化して検知を回避できるウィルスに使用されます。

これらすべての種類のウィルスに共通するのは、ホストファイルやメディアに感染する必要があるという点です。感染したファイルは、ディスクやネットワーク、電子メールの添付ファイルや、ソーシャルメディアの投稿、Webサイトからのダウンロードとしてなど、あらゆる通常の手段によって配信される可能性があります。



Outlookのメールフィルターによって検知された安全でない添付ファイル「二重」ファイル拡張子は、メッセージテキストで英語とドイツ語の両方を使用することで、まだ警告を受けていないユーザーをだますための単純な試みです。（スクリーンショットはMicrosoftからの許可を得て使用。）

コンピューターワームとファイルレスマルウェア

コンピューターワームは、メモリ常駐のマルウェアで、ユーザーが介入することなく実行でき、ネットワークリソースを通じて複製できます。ウイルスは、感染した実行可能プロセスをダウンロードして実行する、感染したUSBスティックを使用する、感染したWordドキュメントをマクロを有効にして開くなどのアクションをユーザーが実行する場合のみに実行されます。一方でワームは、ユーザーがWebサイトをブラウズしたり、脆弱なサーバーアプリケーションを実行したり、感染したファイル共有に接続する場合に、プロセスの脆弱性を悪用して実行できます。例えば、Code-Redワームは、バッファーオーバーフローの脆弱性を介して初期バージョンのMicrosoft's IIS Webサーバーソフトウェアに感染することができました。そしてランダムに生成されたIP範囲をスキャンして、他の脆弱なIISサーバーへの感染を試みました(caida.org/research/security/code-red)。

最初のタイプのコンピューターワームの主な影響は、ワームが自己複製するに従い、ネットワークの帯域幅が急速に消費されることです。ワームはまた、（サービス拒否攻撃を実行して）オペレーティングシステムやサーバーアプリケーションをクラッシュさせることもあります。さらにウイルスと同様に、ワームは他の悪意のあるアクションを実行できる可能性があるペイロードを格納することができます。

Confickerワームでは、リモートコード実行とメモリ常駐マルウェアで非常に強力な攻撃を実行する可能性を示しました(secureworks.com/research/downadup-removal)。マルウェアが犯罪目的で開発され続け、セキュリティソフトウェアで静的な脅威をより適切に検知し、ブロックできるようになるにつれ、マルウェアのコードと手法はより高度になっています。最近よく耳にするファイルレスという用語は、こうした最新のマルウェアを指します。ファイルレスは明確な分類ではありませんが、一般的な動作と手法をまとめて説明しています。

- ファイルレスマルウェアでは、ディスクにコードを書き込みません。このマルウェアではメモリ常駐手法を使用し、ホストプロセスやダイナミックリンクライブラリ(DLL)内、またはスクリプティングホスト内で独自のプロセスを実行します。ですが、これはディスクのアクティビティがないことを意味しているわけではありません。マルウェアは、持続性を維持するためにレジス

トリ値を変更する可能性があります（ホストコンピューターが再起動する時に実行します）。また、マルウェアの最初の実行はダウンロードしたスクリプト、ファイル添付、またはトロイの木馬ソフトウェアパッケージを実行するユーザーによって異なります。

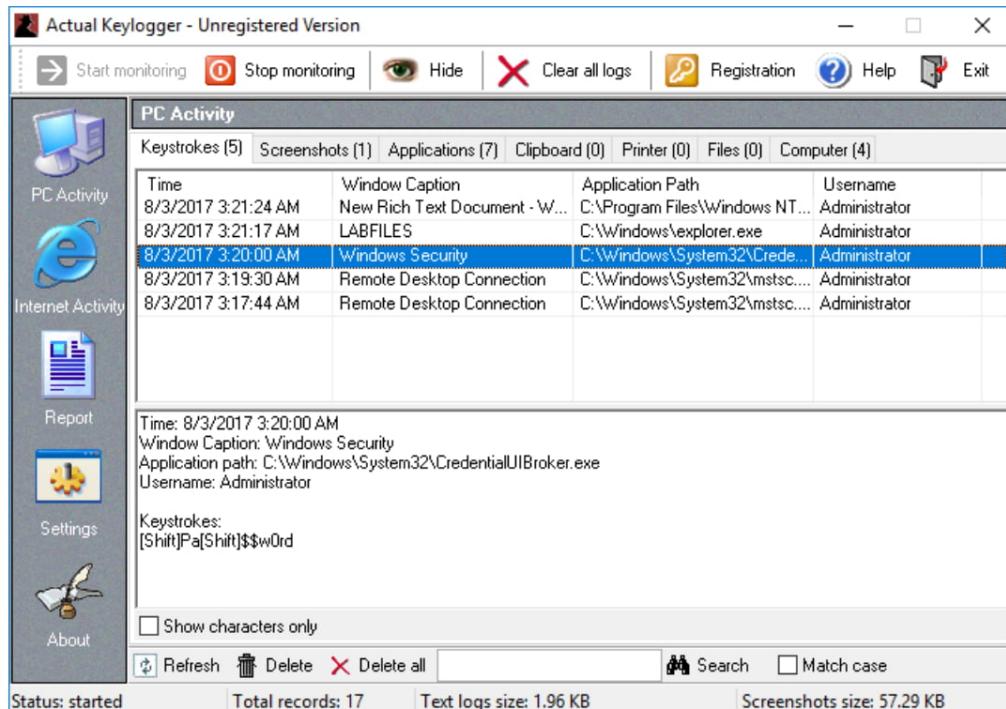
- ファイルレスマルウェアでは、軽量の**シェルコード**を使用して、ホストのバックドアメカニズムを実現します。シェルコードは、スキャナーによる検知を回避するために難読化された形式で簡単に再コンパイルできます。その後、追加パッケージやペイロードをダウンロードして、アクターの目的に対するアクションが達成できるようになります。またこれらのパッケージは、自動検知を回避するためにひそかに難読化、ストリーミング、コンパイルできます。
- ファイルレスマルウェアでは、検知を回避するためにコンパイルした実行可能ファイルではなく、「自給自足」手法を使用する可能性があります。これは、マルウェアコードが特にPowerShellやWindows Management Instrumentation (WMI)などの正当なシステムスクリプティングツールを使用して、ペイロードアクションを実行することを意味します。十分な権限で実行できる場合、こうした環境では、脅威アクターがスキャンを実行し、設定を再構成して、データを盗み出すために必要なすべてのツールが提供されます。

高度標的型脅威(APT)と高度揮発性脅威(Advanced Volatile Threat, AVT)という用語を使用して、この一般的なクラスの最新のファイルレス/自給自足型マルウェアを説明できます。別の役立つ分類は、低観測性(LOC)攻撃です(mcafee.com/enterprise/en-us/security-awareness/ransomware/what-is-fileless-malware.html)。正確な分類は、脅威アクターがさまざまなコーディングのトリックを使用して侵入を実行でき、検知を回避するための戦術、手法、手順が継続的に進化しているという認識ほど重要ではありません。

スパイウェアとキーロガー

初期のウイルスとワームは複製が可能という有害な可能性に重点を置いていました。ですがこのソフトウェアによって利益を生み出す使用法が明らかになると、侵入、詐欺、データの窃盗を容易にするように設計されたペイロードでコード化されるようになりました。さまざまな種類の望ましくないコードとマルウェアが、ある程度の監視を実行します。

- **トラッキングCookie** — Cookieは、マルウェアではなく平文のファイルですが、ブラウザの設定でサードパーティのCookieが許可されている場合は、アクセスしたページ、検索クエリ、ブラウザのメタデータ、IPアドレスを記録するために使用できます。トラッキングCookieは、多くのWebサイトに組み込まれる広告や分析ウィジェットによって作成されます。
- **アドウェア** — これは、トラッキングCookieの許可、デフォルト検索プロバイダーの変更、起動時にスポンサーのページを開く、ブックマークの追加など、ブラウザの再構成を実行するPUP/グレイウェアのクラスです。アドウェアは、プログラムとして、またはブラウザの拡張機能/プラグインとしてインストールされる可能性があります。
- **スパイウェア** — これはアドウェアのようなトラッキングを実行できるマルウェアですが、ローカルアプリケーションアクティビティの監視、スクリーンショットの撮影、マイクやWebカメラなどの記録デバイスの有効化も実行できます。他のスパイウェアの手法としては、ファーミングサイトに対するDNSリダイレクトの実行があります。
- **キーロガー**は、キーストロークを記録することで機密情報を積極的に盗もうとするスパイウェアです。通常脅威アクターは、パスワードまたはクレジットカードデータを発見することを望んでいます。



*Actual Keylogger*はバックグラウンドで実行して、さまざまな種類のコンピューターアクティビティ（プログラムの起動/終了、Webサイトの閲覧、キーストロークの記録、スクリーンショットのキャプチャなど）を監視できるWindowsソフトウェアです。（スクリーンショットはActualKeylogger.comからの許可を得て使用。）



キーロガーはソフトウェアとして実装されるだけではありません。悪意のあるスクリプトは、キー操作をサードパーティのWebサイトに送信できます。また、キーボードとポートの間に改ざんしたUSBアダプターを挿入し、そこにキー操作をキャプチャするハードウェアデバイスもあります。そのようなデバイスでは、データをローカルで保存したり、Wi-Fi接続可能でデータを隠れアクセスポイントに送信したりできます。その他の攻撃には、キープレステータを記録するワイヤレススニッファーやオーバーレイATM PINパッドなどがあります。

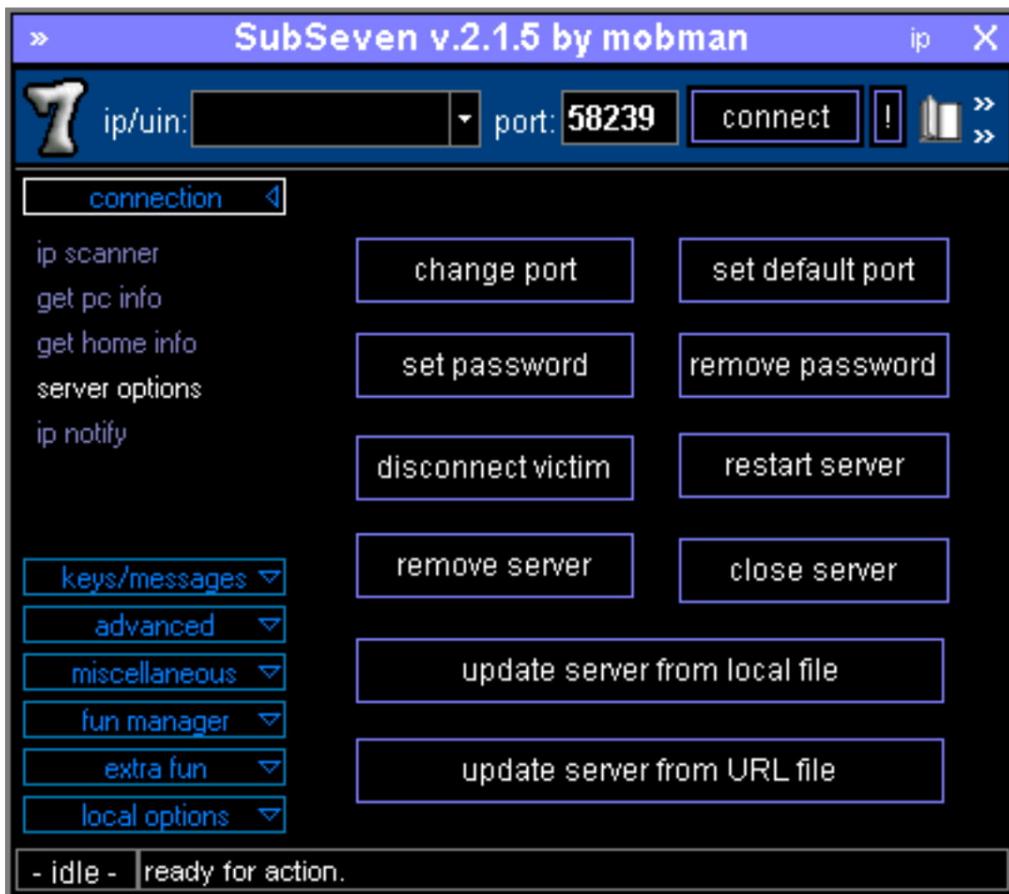
バックドアおよびリモートアクセス型トロイの木馬

通常の認証方法を回避し、リモートユーザーに管理制御を与える、あらゆるタイプのホストへのアクセス方法は、**バックドア**と呼ばれることがあります。リモートアクセス型トロイの木馬(RAT)は、正規のリモートコントロールプログラムの機能を模倣するバックドアマルウェアですが、特に密かに動作するように設計されています。RATがインストールされると、脅威アクターはホストにアクセスし、ファイルをアップロードしたり、ソフトウェアをインストールしたり、「自給自足」手法を使用してさらなる侵害を実行できます。



この場合、RATはリモート管理ツールの略語でもあります。悪意のある制御下にあるホストは、ゾンビと呼ばれことがあります。

侵害されたホストは、1つまたは複数のポートでインストールできます。ポートとは、自動化されたスクリプトまたは一種の悪意のあるアクティビティを実行するツールです。同じマルウェアのインスタンスの制御下にあるポートのグループは、運用者のプログラムによって**ポートネット**として操作できます。ポートネットは、分散型サービス拒否(DDoS)攻撃のトリガー、スパムキャンペーンの開始、または暗号マイニングの実行などさまざまな悪意のある目的に使用できます。



SubSeven RAT。(スクリーンショットはCCAS4.0 InternationalのWikimedia Commonsの許可を得て使用。)

バックドアをスタンドアロン型の侵入メカニズムとして使用する場合や、ボットを管理するために使用する場合であっても、脅威アクターは、侵害されたホストから**コマンド&コントロール(C2またはC&C)** ホストやネットワークへの接続を確立する必要があります。通常このネットワーク接続は、RAT、バックドア、ボットの存在を特定する最適な方法です。検知とフィルタリングを回避するための**隠れチャネル**として、C&Cネットワークの実装方法はたくさんあります。歴史的に、**インターネットリレーチャット(IRC)**プロトコルが一般的でした。最近の方法では、HTTPSやDNSトライフィックに埋め込まれたコマンドシーケンスを使用する可能性が高くなっています。



バックドアは、マルウェアによる感染以外の方法で作成できます。プログラマーは、テスト用や開発用のソフトウェアアプリケーションにバックドアを作成できます。バックドアはその後アプリケーション展開時に削除されません。またバックドアは、不正なユーザーにアクセスを許可するソフトウェアまたはハードウェアの設定ミスによっても作成されます。例として、ルーターをデフォルトの管理パスワードで構成したままにする、リモートデスクトップ接続を安全でないパスワードで構成したままにする、モデムを開いたままにしてダイヤルアップ接続を受信するなどが挙げられます。

ルートキット

Windowsの場合、マルウェアはローカル管理者権限で手動でのみインストールできます。つまりユーザーは認証情報を入力するか、ユーザーアカウント制御(UAC)プロンプトを受け入れるために、インストーラーパッケージに精通している必要があります。Windowsでは、管理者権限の悪用からシステムの保護を試みます。重要なプロセスは、高レベルの権限(SYSTEM)で実行されます。結果として通常のソフトウェアと同じ方法でインストールされたトロイの木馬は、その存在を完全に隠すことはできず、実行中のプロセスまたはサービスとして表示されます。多くの

場合、プロセスイメージ名は検知を回避するために本物の実行可能ファイルまたはライブラリに似せてあります。例えばトロイの木馬は「rundll32.exe」を装って、「rundll1132.exe」というファイル名を使用する可能性があります。トロイの木馬の場合、持続性を確保（コンピューターが再起動する際に実行）するには、レジストリエントリを使用するか、サービスとして自身を作成する必要がありますが、通常は簡単に検知されてしまいます。

マルウェアをペイロードとして供給し、重大な脆弱性を突いてエクスプロイト攻撃を行う場合、SYSTEM権限を使用して認証の必要なく実行できる可能性があります。またマルウェアでは、エクスプロイトを使用して、インストール後に権限をエスカレーションさせる可能性があります。このレベルの権限を使用して実行するマルウェアは、**ルートキット**と呼ばれます。この用語は、ルートとして実行されるプロセスがファイルシステムのルートから以下のすべてに無制限にアクセスできるUNIX/Linuxに由来します。

理論上は、ルートキットが変更できないシステムはありません。実際には、Windowsでは他のメカニズムを使用して、コード署名などのカーネルプロセスの誤用を防ぎます(microsoft.com/security/blog/2017/10/23/hardening-the-system-and-maintaining-integrity-with-windows-defender-system-guard)。結果として、ルートキットができることは、敵の能力と努力のレベルに大きく依存します。ルートキットを扱う際は、マルウェアがシステムファイルとプログラミングインターフェイスを侵害して、WindowsのExplorer、taskmgr、tasklist、Linuxのpsやtopなどのローカルシェルプロセスに加えて、netstatなどのポートスキャントールが、マルウェアの存在を表示しなくなる（少なくとも、感染したマシンから実行される場合）ようにする可能性があることに注意する必要があります。また、ルートキットにはシステムログを消去し、さらにその存在を隠すツールが含まれます(microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Win32%2fCutwail)。



ソフトウェアプロセスは、いくつかある「リング」の1つで実行できます。リング0は最も特権があるため（ハードウェアへの直接アクセスを提供）、カーネルプロセス専用にしてください。リング3は、ユーザーモードプロセスが実行される場所です。ドライバーとI/Oプロセスは、リング1またはリング2で実行できます。またこのアーキテクチャは、仮想化を使用することにより複雑になる可能性があります。

ファームウェア（コンピューターファームウェア、またはあらゆる種類のアダプターカード、ハードドライブ、リムーバブルドライブ、周辺機器のファームウェアのいずれか）に常駐できるルートキットの例もあります。これはドライブをフォーマットして、OSを再インストールしてルートキットを削除しようと試みても、できません。例えば、米国情報局では、Apple Macbookラップトップでファームウェアを対象となるDarkMatterとQuarkMatter EFIのルートキットを開発しました(pcworld.com/article/3179348/after-cia-leak-intel-security-releases-detection-tool-for-efi-rootkits.html)。

ランサムウェア、クリプトマルウェア、論理爆弾

ランサムウェアは、被害者から金銭を奪おうとする種類のマルウェアです。あるタイプのランサムウェアは、Windowsのライセンス認証を再実行することを要求したり、コンピューターが児童ポルノの表示やテロに使われたために警察によってロックされたと警告するなどといった脅迫メッセージを表示します。また、別のシェルプログラムをインストールしてファイルシステムへのアクセスを明らかにブロックするケースもありますが、このような攻撃は通常、比較的容易に修正できます。



WannaCryランサムウェア。(画像提供：Wikimedia Commons)

ランサムウェアのクリプトマルウェアクラスとして、固定ドライブやリムーバブルドライブ、ネットワークドライブ上のデータファイルを暗号化しようとするものもあります。攻撃が成功すると、ユーザーは脅威アクターが所有する秘密暗号化の鍵を取得しなければファイルにアクセスできなくなります。このような攻撃が成功した場合、暗号化されたファイルの最新バックアップを保有していない限り、解決は極めて困難です。この例としてクリプトロッカーがあります。これは暗号化するファイルを検索し、特定のカウントダウン時間の前に被害者に支払いを済ませるように促すトロイの木馬です。この後マルウェアでは復号を可能にする鍵を破壊します。

ランサムウェアは、電信送金や暗号通貨で身代金を支払うよう要求したり、プレミアム料金の電話サービスに登録させるなどの方法で、脅威アクターの身元を明かさず、また警察に逆探知されることもなく金銭の強奪を可能にします。

別の種類のクリプトマルウェアでは、ホストのリソースをハイジャックして、暗号通貨マイニングを実行します。これはクリプトマイニングまたはクリプトジャックと呼ばれます。暗号通貨内のコインの総数は、新しいデジタルコインを作成するために必要な計算が難しいため、制限されています。結果として、新しいコインは非常に価値が高くなる可能性がありますが、コインを発見するには膨大なコンピューティングリソースが必要になります。多くの場合、クリプトジャックはボットネットを通じて実行されます。

一部の種類のマルウェアは自動的にトリガーされません。システムに感染すると、事前に構成された日時（時限爆弾）またはシステムやユーザーイベント（論理爆弾）を待ちます。また論理爆弾は、マルウェアコードである必要はありません。典型的な例として不満を持つシステム管理者が挙げられ、自分のアカウントが削除または無効化された場合に実行されるスクリプトトラップを残します。ウイルス対策ソフトウェアでは、この種の悪意のあるスクリプトやプログラムを検知する可能性がほとんどありません。このタイプのトラップは地雷と呼ばれます。

マルウェアインジケーター

マルウェアの種類の範囲を考えると、多くの潜在的なインジケーターがあります。ある種のマルウェアでは、ブラウザの設定を調整したり、身代金の通知を表示するなど、明らかな変更を表示します。マルウェアが密に動作するように設計されている場合、インジケーターがプロセス、ファイルシステム、ネットワークの動作の詳細分析を要求することがあります。

ウイルス対策の通知

ほとんどのホストは、ある種のウイルス対策(A-V)ソフトウェアを実行しているはずです。ウイルス対策モニターが引き続き一般的ですが、これらのスイートはエンドポイント保護プラットフォーム(EPP)または次世代ウイルス対策として考えられています。これらはタイプに関係なくシグネチャによってマルウェアを検知しますが、検知率は製品ごとに非常に大きく異なる可能性があります。また多くのスイートは、ユーザーとエンティティの行動分析(UEBA)で統合され、AIに裏付けられた分析を使用して、マルウェアのシグネチャマッチングを迂回した脅威アクターの動作を検知します。

サンドボックスの実行

疑わしいコードがエンドポイント保護によって検知されない場合は、サンドボックス環境で分析してみてください。サンドボックスは、ホストから完全に隔離されて構成されたシステムであるため、マルウェアは「ブレークアウト」できません。サンドボックスは、ファイルシステムとレジストリの変更、ネットワークアクティビティを記録するように設計されます。Cuckooは、ターンキーサンドボックスソリューションを提供するためのパッケージ化されたソフトウェアです(cuckoosandbox.org)。

リソースの消費

異常なリソースの消費は、パフォーマンスマニテー、タスクマネージャー、`top` Linuxユーティリティを使用して検知できます。過剰で継続的なCPUの使用率、メモリリーク、ディスクの読み取り/書き込み、ディスクスペースの使用率などのインジケーターはマルウェアの兆候の可能性がありますが、他にもさまざまなパフォーマンスやシステムの安定性の問題によって引き起こされる可能性があります。またこのような動作を示すのは、非常に粗悪に記述されたマルウェアまたは集中的な操作を実行するマルウェア（ボットネットDDoS、クリプトジャック、クリプトランサムウェアなど）だけです。リソースの消費は、感染の確固たる証拠ではなく、システムを調査する理由になります。

ファイルシステム

ファイルレスマルウェアは確かに蔓延していますが、ファイルシステムの変更や異常分析は依然として必要になります。マルウェアコードがディスクに保存されていなくても、マルウェアはファイルシステムやレジストリとやり取りする可能性が高く、その存在は動作によって明らかになります。コンピューターのファイルシステムでは、ファイル作成日やアクセス日、変更日について有用なメタデータが大量に保存されます。こうしたメタデータを分析し、疑わしい一時ファイルを確認することで、ホストやそのファイルに形跡が残されたインシデントのイベントのタイムラインを確立できるようになります。

プロセス分析

シェルコードは簡単に難読化できるため、多くの場合シグネチャベースのウイルス対策製品を回避できます。脅威ハントティングとセキュリティのモニタリングでは、動作ベースの手法を使用して感染を特定する必要があります。これは、ホストのシステムメモリで実行されるプロセスを密接に分析することを意味します。異常プロセス行動分析を効果的に実行するには、システムでの「通常状態」を設定し、感染した可能性のあるシステムでの逸脱を特定してください。また、適切な分析ツールを使用する必要があります。Sysinternals (docs.microsoft.com/en-us/sysinternals)はWindowsの問題のトラブルシューティングをサポートするための一連のツールで、その多くはセキュリティ問題の調査向けです。Sysinternalsのツールであるプロセスエクスプローラーは、タスクマネージャーの機能強化されたバージョンになります。各プロセスに関する追加情報を確認し、親/子関係でどのようにプロセスが作成されたかをより具体的に理解できます。

この例の場合、Metasploit Frameworkは、取得したハッシュを渡すことで得た権限を使用して、リモートで実行されたPowerShellプロンプト経由でアクセスを得るために使用されます。この攻撃は、Sysinternals PsExecユーティリティを利用して、リモートマシンのAdmin\$共有フォルダにサービス実行ファイルをドロップします。この攻撃のバリエーションでは、サービスはPowerShellを起動します。プロセスエクスプローラーでpowershell.exeイメージにポイントすると、起動したプロセスのパラメータが表示されます。この場合、これを起動するために使用されるコマンドは、PowerShellの一般的な使用法ではありません。長い文字列がありますが、これはBase64で表示されるバイナリコードです。スクリプトはこれを新しいDLLに挿入し、メモリにのみ保存します。

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name	User Name
System Idle Process		0 K	4 K	0		NT AUTHORITY\SYSTEM	
System	3.50	108 K	180 K	4		NT AUTHORITY\SYSTEM	
csrss.exe	0.71	1,716 K	2,796 K	416	Client Server Runtime Process	Microsoft Corpor...	NT AUTHORITY\SYSTEM
csrss.exe		1,284 K	2,348 K	480	Client Server Runtime Process	Microsoft Corpor...	NT AUTHORITY\SYSTEM
wininit.exe		772 K	2,276 K	488	Windows Start-Up Application	Microsoft Corpor...	NT AUTHORITY\SYSTEM
winlogon.exe		1,564 K	2,596 K	532	Windows Log-on Application	Microsoft Corpor...	NT AUTHORITY\SYSTEM
csrss.exe	0.12	1,636 K	18,036 K	2384	Client Server Runtime Process	Microsoft Corpor...	NT AUTHORITY\SYSTEM
winlogon.exe		1,220 K	4,700 K	2688	Windows Log-on Application	Microsoft Corpor...	NT AUTHORITY\SYSTEM
explorer.exe	0.35	62,420 K	127,868 K	11944	Windows Explorer	Microsoft Corpor...	classroom\Administrator
processh64.exe	10.64	18,864 K	37,108 K	35760	Sysinternals Process Explorer	Sysinternals - ww...	classroom\Administrator
cmd.exe		1,480 K	2,248 K	46816	Windows Command Processor	Microsoft Corpor...	classroom\Administrator
Procmon.exe		2,024 K	10,448 K	109844	Process Monitor	Sysinternals - ww...	classroom\Administrator
powershell.exe	0.07	41,288 K	43,508 K	112120	Windows PowerShell	Microsoft Corpor...	NT AUTHORITY\SYSTEM

Name	Command Line:
System.M.	"C:\Windows\syswow64\WindowsPowerShell\v1.0\powershell.exe" -nop -w hidden & \$s=New-Object IO.Memor...
Microsoft.	yStream,[Convert]::FromBase64String(H4sIAjXNDgFgaCA7VWa2+bSBTSnErD6yBK0O3EaJVkHYwfM2zQzqjNzQAg...
Microsoft.	OMP TAODH Hst99LzYk7Ibd7e5qkS3mc>...</pre>
Microsoft.	oo3EfJWWk0ludh5hGz70zVnrHJBK7qVLB_EoSEt4zShJfb5j44DE5P0yf4c1XZV2Sr9xUozfY5brpVvYC_Yn0C13m+tzB2eRv...
Microsoft.	Vr7CwUGZ2DmDvdW0_SgEPfB75igIKGw5LP2mTPArNI0jDAvCaX10kfQGSo9HLMXBVnpvgzlaq3zop+05NRSzWgZO3o...
Microsoft.	7U5G7kyM5ZV/H-kylC8zQnDC9.fv3r.zChGe2uywnBoHUy3bQjhKk0e0K3d26WlkzYDQserfbuolTos6kacbAdDaTSu16eL...
Microsoft.	1lg5v6Sfrnq9GLf3DA150Lfbvrw/DU5tSdgVvOlsm93p94mzn53TiUcqjg8HFKnkjTyFvTEY2R73kphNoDGFmfIK50GFGrxyJ...
System.Xr.	AsS9PfXbu20imdfLaXMuTFygL4EogJm1R-D2ZGjyE2dkhDw2vloMEDIZPCoHfvutg9640R3G14ScrSMIVMscaSRTAjbllCULzKZ...
System.Tr.	KQvn.3KL+GaKRPUwYkolpup.L0mj0724l4gYBEOP2NtSQx.SwDoyz1qEu0tUX9Ymf5tShamDea+bDS1Ab1kxElsikEUQzezJQKx...
System.M.	YRRRhkJATLbWp3GPYhkfNc2GK+8SVX0daSH2n6wyVAo6901Fgj3FRlmwaC6gRGckFav5LKHt1lg+qFZOclK0lam2Fpnys.FmHT...
System.Di.	s5D15Cq36SKTb-HbItLAACd1sxDD5rkU9MSMWCrhKe0haCZ2jEzHS0Ba2FaDbJvxH9Mlg+o17cT7vWP9FCQk.Rhmb6nf9Xrxh3...
System.Da.	PLbgarbYlLoSHM9u18bqHe9W/g1g7zUu6G1xaSSWZ7TjdVH7uSp+nnmbvY17Wkz91voneuf+jZ1/XyDu2PWIdarYH7ejjtj7WVvm...
System.Co.	smbrngXdhHR1eK8l+4nNmJr+rf1k8ferHc7v0zY2BUDc4gbmmt0NTHe96VVpx80FaIUp2P+0fog5GRVr74m5027/FtucBh...
System.Co.	MH/bTqg+0tRga61sfmxG./spnLVy5uqtW7cDR6uOoy8XHrbMKfbR1E70x6B6at+yR/C/X9Sz9gPq8nbmh0Rlk31KdX/WRRxxv9Qb...
System.Co.	hEGKErhLQx9jU-vn+Ofaq3q1+eEcduox74bciaTuBBDPHbH3UV39wmnf3LMzx8y1oH20vJ48DiwlnbA9p8Wek2cZwEmAHDDU...
System.ni.	CluOvwuJNx1yGmYel/FftgsQRYXBwRVWaByxop2s0G9LWMwyu9l/g-QbQf0o8WZLZ4N1zqXwynd1BjQAB+9Kg9Enki6Bcez...
ierutil.dll	qq1aCa156aNTj2x+yyZdr5Ycly9ntsifZn3dk2x3VLR1KybgajdU4nZv+9jm6drAC/377B9GfuL2V/Cu1bx+LVS1BD/wf4...
wininet.dll	PBGMFBxhYUHE22F+LbUOS2kvul2oK90L7Jd5mKwwF8Y/wBMOGckAAA-);:EX(New-Object IO.StreamReader(Ne...
msvcr.dll	w-Object: IO.Compression.Gzip Stream(\$s,[IO.Compression.CompressionMode]::Decompress))).ReadToEnd();
Path:	4 ...
	5 ...
	C:\Windows\SysWOW64\WindowsPowerShell\v1.0\powershell.exe

PsExecを使用して、メモリに常駐するシェルコードを作成するPowerShellスクリプトを呼び出す方法を確認。(スクリーンショット：プロセスエクスプローラー docs.microsoft.com/en-us/sysinternals)

この種の動作は、マルウェアがサンドボックスで実行される場合にリアルタイムでのみ観察できます。脅威の探索と自動検出ツールは、System Monitor (github.com/SwiftOnSecurity/sysmon-config)が提供するような詳細なログを使用して、悪意のあるプロセスの動作を記録し特定することができます。

プロセスがファイルシステムとどのようにやり取りするのかを確認するとともに、ネットワークのアクティビティは、マルウェアを特定する最も信頼できる方法の1つです。脅威データを使用して、既知の不正なIPアドレスやドメインへの接続を関連付けることができますが、マルウェアはfast-fluxやドメイン生成アルゴリズム(DGA)などの手法を活用して、絶えず変化するエンドポイントへの接続を試みる可能性があります。またソーシャルメディアやクラウドサービスを使用して正規のトラフィックにまぎれ込もうとする可能性があります。

レビュー アク ティビティ：

マルウェアベースの攻撃のインジケーター

次の質問にお答えください。

- あなたは、あるユーザーのワークステーションのトラブルシューティングを行っています。コンピューターの画面にはすべてのファイルが暗号化されていることを示すアプリのウィンドウが表示されています。アプリのウィンドウでは、データを回復する場合は、匿名で支払いをするように要求しています。コンピューターが感染したのはどのタイプのマルウェアですか。
- あなたは、小さな旅行代理店のCEOにさまざまなウイルス対策製品を勧めています。CEOは、トロイの木馬が近年のコンピューターセキュリティの最大の脅威であると耳にしているため、困惑しています。どのような説明をすることができますか。
- あなたは、自身の脅威プラットフォームに登録する会社CEOのためにセキュリティ認識に関するブログを書いています。マルウェアのリスクを分類し、特定する方法が、バックドアとトロイの木馬で異なるのはなぜですか。
- あなたはビジネス電子メール詐欺(BEC)のインシデントを調査しています。開発者の電子メールアカウントがWebメールを通じてリモートでアクセスされたのです。開発者のワークステーションを調査すると、悪意のあるプロセスを示すものはありませんでしたが、背面にあるポートの1つに不明なUSB拡張デバイスが接続されました。これは攻撃ベクターの可能性が高いですか。その場合、どのようなマルウェアが実装される可能性がありますか。
- あるユーザーのコンピューターのパフォーマンスが非常に遅くなっています。調査を行ったところ、n0tepad.exeというプロセスが80～90%のレートでCPUを使用していました。これには、継続的な小量のディスクの読み取りと一時フォルダへの書き込みが伴います。マルウェアの感染を疑うべきですか。またそれには特定のクラスがありますか。
- Cuckooはマルウェアの種類ですか。それともセキュリティ製品ですか。

レッスン4

概要

脅威アクターが侵入を成功させるため使用するソーシャルエンジニアリングとマルウェアベースの方法を特定できる必要があります。

ソーシャルエンジニアリングとマルウェアの特定のガイドライン

ソーシャルエンジニアリングとマルウェア攻撃からセキュリティシステムを保護するためにセキュリティ評価を使用する際は、次のガイドラインに従ってください。

- トレーニングプログラムや教育プログラムを使用し、ソーシャルエンジニアリングがどのように有効であるか(権威、威圧、コンセンサス、希少性、親密性、信頼、緊急性)を従業員に認識させる。
- ソーシャルエンジニアが情報を引き出したり、不正アクセスを取得したりするのを妨げるポリシーと手順を使用する。
- ドメイン名の確認や疑わしいメッセージの特定など、ユーザーがフィッシングとファーミングの試行を認識できるように指導する。
- トレーニングプログラムと教育プログラムを使用して、従業員がマルウェアの脅威の種類(トロイの木馬、PUP、スパイウェア、バックドア、ボット、ルートキット、ランサムウェア)とマルウェアが実行される可能性のあるベクトルを認識できるようにする。
- セキュリティフィルターと権限の制限を使用して、ユーザーによる感染したファイルやスクриプトの実行を制限する。
- ファイルレスマルウェアをより効果的に検知できる行動ベースのエンドポイント保護スイートの実装を検討する。
- 分析ツールでサンドボックスを設定し、疑いのあるプロセス動作を調査することを検討する。
- 脅威データフィードを使用して、コマンドとコントロールネットワークの特定をサポートすることを検討する。

レッスン5

暗号コンセプトの基本を要約する

レッスン概要

評価および監視アクティビティでは、潜在的な攻撃ベクトルの特定や悪意のあるアクティビティの検出に脅威インテリジェンスを利用します。サイバーセキュリティ保護機能は、機密性、完全性、可用性の特性による安全なIT処理システムを構築することを目的にしています。セキュリティシステムの多くは全面的または部分的に暗号アルゴリズムに依存しています。

暗号システムは、権限のある人物だけが復号できる方法で、データを暗号化します。暗号技術は、多くのセキュリティシステムを実装または設定する際の基盤となるものです。情報セキュリティの専門家として、暗号アルゴリズムのコンセプトと、セキュアプロトコルおよびサービスへの実装についてよく理解する必要があります。全てのセキュリティ担当者は、さまざまなタイプの暗号アルゴリズムを比較し、データの機密性、完全性、可用性を適用する場合の使用方法を理解し、発生し得る脆弱性を説明できる必要があります。この項目の技術を確実に理解できること、暗号システムの重要性の説明や、任意のセキュリティ目標を満たす技術の選択が可能になります。

レッスンの目的

このレッスンの内容は、以下のとおりです。

- 暗号アルゴリズムの比較対照。
- 暗号動作モードの概要。
- 暗号のユースケースと脆弱性の概要。
- その他の暗号化技術の概要。

トピック5A

暗号アルゴリズムを比較対照する



対象試験範囲

2.1 エンタープライズ環境におけるセキュリティコンセプトの重要性を説明することができる
(ハッシュ化のみ)

2.8 暗号化コンセプトの基本を要約することができる

暗号アルゴリズムは、データの暗号化または復号の際に実行する特別な演算です。最新の暗号システムでは、データの暗号化や復号に対称暗号方式（共通鍵暗号方式）および非対称暗号方式（公開鍵暗号方式）を使用します。これらの暗号アルゴリズムの種類と同様に、多くのセキュリティ管理で、ハッシュ関数が重要な役割を果たしています。さまざまなユースケースのセキュリティ管理を行う上で、こういった暗号アルゴリズムの種類およびハッシュ関数の特徴を比較対比できることが重要です。

暗号のコンセプト

暗号技術（文字通り「秘密の文書」）は数千年にわたって存在してきました。これは、符号化することで情報の安全を守るための技術です。これは、隠ぺいによるセキュリティとは真逆のコンセプトです。隠ぺいによるセキュリティとは、隠すことで秘密を守ることです。これは一般的に、コンピューターネットワーク上では不可能（あるいは少なくとも、ハイリスク）だと考えられています。暗号技術を使用すれば、第三者が秘密の存在を知っても、適切な鍵を入手しない限り、それがどんなものかを知ることはできないため、問題になりません。

以下の用語は、暗号技術を論じる際に使用します。

- **平文（プレーンテキスト）**（またはクリアテキスト）—暗号化されていないメッセージ。
- **暗号アルゴリズム**—暗号化されたメッセージ。
- **暗号アルゴリズム**—メッセージの暗号化/復号に使用するプロセス（またはアルゴリズム）。
- **暗号解読**—暗号システムを破る技術。

暗号技術と暗号システムに対する攻撃を論じる際には、キャラクターを使って攻撃プロセスに関するさまざまな役割を説明します。主なキャラクターは次のとおりです：

- アリス—本物のメッセージの送信者。
- ボブ—メッセージの本来の受信者。
- マロリー—さまざまな方法でメッセージの改竄を試みる悪意のある脅威アクター。

暗号アルゴリズムには、機密性、完全性、可用性、否認防止というセキュリティ特性を保証するため、それぞれ異なる役割を持つ3つのタイプがあります。これらは、ハッシュアルゴリズムと、対称型と非対称型の2種類の暗号アルゴリズムです。

ハッシュ化アルゴリズム

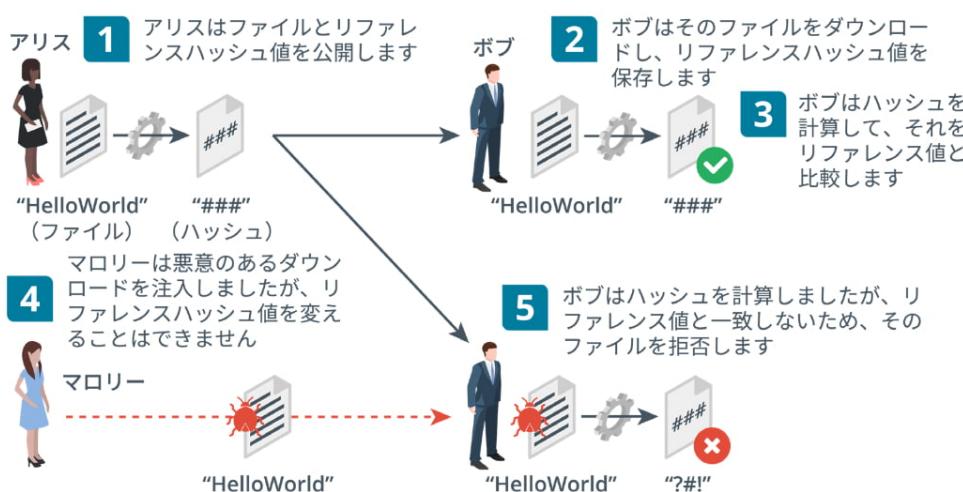
ハッシュ化とは、最も単純な暗号演算です。暗号ハッシュ化アルゴリズムは、任意の長さの入力平文から決まった長さの文字列を生成します。出力は**チェックサム**、メッセージダイジェスト、またはハッシュと呼ばれます。ダイジェストから平文データを復元できず（一方向）、異なる入力から同じ出力を生成（衝突）する可能性が低くなるように設計されています。

ハッシュ化アルゴリズムは、完全性の証明に使用されます。例えば、ボブとアリスは、次の方でパスワードに使用する値を比較できます：

1. ボブは既にアリスの平文パスワードから計算したダイジェストを持っています。ボブはハッシュから平文パスワードの値を復元することはできません。
2. アリスがボブを認証する必要がある場合、自分のパスワードを入力してハッシュに変換し、ボブにダイジェストを送ります。
3. ボブはアリスのダイジェストと自分の持っているハッシュ値を比較します。これらが一致すると、ボブは、アリスが同じパスワードを入力したことを確認できます。

パスワード値の比較と同様に、ファイルのハッシュは、転送後のファイルの完全性を確認するために使用できます。

1. アリスは、自分の製品のsetup.exeファイルに対してハッシュ関数を実行します。アリスは、ファイルのダウンロードリンクを使って自分のWebサイトにダイジェストを公開します。
2. ボブは、Webサイトからsetup.exeファイルをダウンロードし、公開されているダイジェストのコピー入手します。
3. ボブは、ダウンロードしたsetup.exeファイルに対して同じハッシュ関数で実行し、アリスが公開したダイジェストと比較します。Webサイトで公開された値と一致している場合、ファイルは同じであるとみなすことができます。
4. マロリーが悪意のあるファイルをダウンロードファイルとすり替えることができたと仮定します。その場合、マロリーは参照ハッシュを変更することはできません。
5. 今回はボブがハッシュを計算しても一致しないので、ファイルが改竄されていることを疑うことができます。



暗号ハッシュを使ったファイルダウンロードの確認（画像提供：© 123RF.com）

一般的に、2つのハッシュアルゴリズムの実装方法があります：

- **SHA (Secure Hash Algorithm)**—最も強いアルゴリズムであると考えられています。異なるサイズの出力を生成するバリエーションが複数存在し、ダイジェストが長ければ長いほどより安全だと考えられています。最も一般的なバリアントはSHA-256で、256ビットのダイジェストを生成します。
- **MD5 (Message Digest Algorithm #5：メッセージダイジェストアルゴリズム#5)**—128ビットのダイジェストを生成します。MD5は、SHA-256ほど安全に使えるとは考えられていませんが、セキュリティ製品間の互換性を考えると必要な場合があります。

```
C:\Users\James\Downloads>fciv -sha1 "c:\users\james\documents\photo.jpg"
// File Checksum Integrity Verifier version 2.05.
haa30028bd0cac06b9d200993ddaa7e613c0af4e6 c :\users\james\documents\photo.jpg
C:\Users\James\Downloads>
```

ファイルからのSHA値の計算。(スクリーンショットはMicrosoftからの許可を得て使用。)

暗号アルゴリズムと鍵

ハッシュ関数はデータの完全性を証明するためには使用できますが、データを保存または送信するためには使用できません。ダイジェストから平文を復元することはできません。暗号アルゴリズムは、データを復元または解読できるように符号化する暗号プロセスです。暗号アルゴリズムと共に鍵を使用すると、権限のある人物だけが解読できるようになります。

換字式暗号と転置式暗号

暗号のしくみを理解するには、単純な換字式暗号と転置式暗号を見るとよく分かります。換字式暗号では、平文内の単位（1文字または文字ブロック）ごとに暗号文と置換します。単純な換字式暗号では、アルファベットの文字列をシフトさせたり、順番を入れ替えたりします。例えば、ROT13の場合（シーザー暗号の一例）、それぞれの文字を13文字分シフトさせることを意味します（例えばAはNとなる）。暗号文「Uryyb Jbeyq」は、「Hello World」を意味します。

転置式暗号では、換字式暗号とは対照的に、平文と暗号文の単位は変りませんが、文字の順番が何らかのメカニズムによって変更されます。暗号文「HLOOLELWRD」がどのように生成されたかを考えてみます：

H L O O L

E L W R D

この暗号文は、縦方向に文字を並べ、それを横方向に連結して作成されています。これは、レールフェンス(rail fence)暗号と呼ばれます。最近の暗号でも、換字式や転置式暗号の基本技術がさらに複雑な方法で使われています。

鍵と暗号アルゴリズム

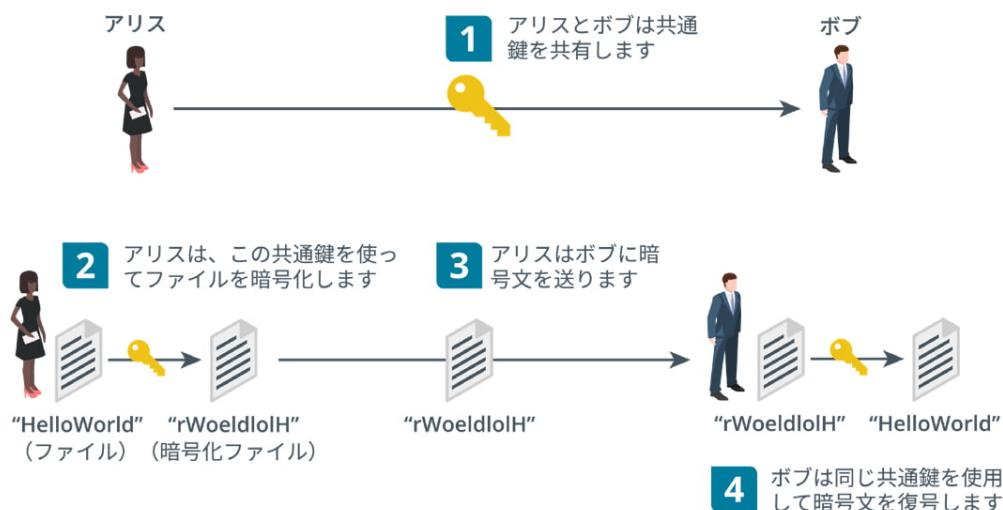
暗号アルゴリズムは、暗号プロセスのセキュリティを強化するために鍵を使用します。例えば、シーザー暗号ROT13の場合の鍵は「13」です。「17」を使用すると、同じ方法で異なる暗号文を作成できます。暗号アルゴリズムが知られても、特定の鍵が分からなければ、メッセージを復号することはできないため、鍵の役割は重要です。これは、最新の暗号技術において特に重要な点です。暗号アルゴリズム（秘密のアルゴリズム）の詳細を隠そうとすることは、「隠ぺいによるセキュリティ」に該当します。最近の暗号は、第三者のレビュー（暗号解析）を受け入れることで、より強化されています。

対称暗号方式（共通鍵暗号方式）

対称暗号方式（共通鍵暗号方式）とは、暗号化も復号も同じ秘密鍵（シークレットキー）で行う暗号アルゴリズムのことです。同じ鍵を使うことから共通鍵暗号方式とも呼ばれ、また秘密鍵を共通鍵とも呼びます。秘密鍵は、文字通り秘匿性が維持される必要があるのでこの名前で呼ばれます。この鍵を紛失したり盗難に遭った場合は、セキュリティが侵害されます。

対称暗号方式は、機密性を維持するために使用します。例えば、アリスとボブは次の方法で、機密ファイルを共有できます：

1. 2人はどの暗号方式と共通鍵の値を使用するかを直接会って確認します。両者は、他の人が発見できないようにこの共通鍵の値を記録します。
2. アリスは、この暗号方式と鍵を使ってファイルを暗号化します。
3. 次にネットワークを介して、暗号文のみをボブに送ります。
4. ボブは、暗号文を受け取り、共通鍵のコピーを同じ暗号方式に適用すると暗号文を復号することができます。



共通暗号化操作と脆弱性（画像提供：© 123RF.com）

対称暗号方式は、单一鍵、秘密鍵、共有秘密とも呼ばれます。なお、「秘密鍵」は非対称暗号方式（公開鍵暗号方式）でも使われる所以、混同しないように注意しましょう。

対称暗号方式は非常に迅速です。対称暗号方式は、大量のデータの暗号化に使用します。主な課題は鍵の分配と保管を安全に行うこと、アリスとボブが「直接会って」鍵を確認するのはそのためです。もしマロリーが鍵を傍受し、暗号文を取得した場合、セキュリティは破壊されます。

アリスとボブはお互いに同じ鍵を知っていたため、暗号化と復号を行うことができましたが、対称暗号方式は認証または完全性の確認に使用できないことに注意してください。

ストリーム暗号とブロック暗号

対称暗号方式には、ストリーム暗号とブロック暗号の2つのタイプがあります。

ストリーム暗号

ストリーム暗号では、平文内のデータの各バイトまたはビットが一度に暗号化されます。ストリーム暗号は、メッセージの長さが分からぬ通信を暗号化するのに適しています。平文は、鍵と初期化ベクトル(IV)から計算される、別のランダムに生成されたメッセージと組み合わされます。IVは、同じ平文から異なる一意の暗号文が鍵によって生成されることを保証します。キーストリームは一意でなければならないので、IVは同じ鍵で再利用することはできません。受信者は、送信者と同じキーストリームを生成でき、ストリームが同期されている必要があります。ストリーム暗号は、同期や再送信を可能にするためにマーカーを使用する場合があります。ストリーム暗号の中には自己同期するタイプのものもあります。

ブロック暗号

ブロック暗号では、平文を同じサイズのブロックに分割します（通常128ビット）。平文に十分なデータがない場合、正確なサイズになるようアルゴリズムで規定された文字列を使ってパディングします。例えば、1200ビットの平文の場合、 10×128 ビットブロックに適合するよう、80ビットを追加します。次に、各ブロックでは、使用する鍵の値に基づいて、複雑な転字演算や換字演算が実行されます。

AES (Advanced Encryption Standard)は、多くの製品に使用されるデフォルトの対称暗号方式です。基本のAESの鍵の長さは128ビットですが、最も一般的に使われるバリエントはAES256で、256ビット鍵を使用します。

鍵の長さ

特定の暗号アルゴリズムと共に使用可能な鍵の値の範囲はキースペースと呼ばれます。キースペースは、鍵のビット長をnとすると、 2^n 乗にほぼ等しくなります。より長い鍵（例えば、128ビットではなく256ビット）を使用すると、暗号化スキームがより強力になります。しかし、異なるアルゴリズムを比較する場合で、同じレベルの強度を実現する鍵の長さは等しくない点に注意してください。特定のアルゴリズムの最小鍵長に関する推奨事項は、暗号解読技術に対してアルゴリズムが脆弱ではないか、および現在の処理リソースに対する「ブルートフォース」攻撃にどれくらい時間がかかるかを前提にしています。

非対称暗号方式（公開鍵暗号方式）

対称暗号方式では、暗号化および復号の両方の演算に1つの共通鍵が使用されます。非対称暗号方式（公開鍵暗号方式）に関しては、2つの関連する公開鍵と**秘密鍵**のペアで演算を実行します。

それぞれの鍵は、そのペアの鍵による演算を反転することができます。例えば、メッセージの暗号化に公開鍵（パブリックキー）を使用した場合、生成された暗号文を復号できるのはペアの秘密鍵だけです。**公開鍵（パブリックキー）**は、暗号化に使用したものであっても、暗号文を復号する時には使用できません。

この鍵は、一方から他方を抽出できないようになっています。これで、鍵の保持者は、安全なメッセージを受け取りたい相手に公開鍵を配布することができます。公開鍵を使ってメッセージを復号することはできず、関連付けられた秘密鍵（プライベートキー）によってのみ復号できます。

1. ボブは鍵のペアを生成し、秘密鍵を秘密にしています。
2. ボブは公開鍵を公開します。アリスはボブに機密メッセージを送りたいと思っているため、ボブの公開鍵をコピーします。
3. アリスはボブの公開鍵を使用してメッセージを暗号化します。