

パスワードクラッカー

評判の悪い Cain や L0phtcrack (l0phtcrack.com) ツールなど、いくつかの Windows ツールがありますが、ほとんどの **パスワードクラッカー** は主に Linux 上で動作します。例えば、**Hashcat** (hashcat.net/hashcat) のようなツールは、次の一般的なシンタックスを使って動作します。

```
hashcat -m HashType -a AttackMode -o OutputFile
InputHashFile
```

入力ファイルには、特定のフォーマットを使った同じタイプのハッシュが含まれている必要があります (hashcat.net/wiki/doku.php?id=example_hashes)。Hashcat は、単一のワードリスト（辞書モード -a 0）または複数のワードリスト（コンビネーター モード -a 1）で使用できます。モード -a 3 はブルートフォース攻撃を実行しますが、文字位置ごとのマスクと組み合わせることができます。これによって検索すべき鍵スペースが減少するため攻撃のスピードが加速します。例えば、会社がパスワードに英数字のみを使用していることを習得したり直感で気づく可能性があります。数字と記号文字を省略すると、各ハッシュに対する攻撃のスピードが加速します。

```
[s]tatus [p]ause [b]ypass [c]heckpoint [q]uit => s

Session.....: hashcat
Status.....: Running
Hash.Type.....: NetNTLMv2
Hash.Target....: ADMINISTRATOR:515support:2f8cbd19fd1bfac9:881c5503...000000
Time.Started....: Mon Jan  6 11:25:16 2020 (1 min, 38 secs)
Time.Estimated...: Sat Jan 11 07:49:57 2020 (4 days, 20 hours)
Guess.Mask.....: ?1?1?1?1?1?1?1?1 [8]
Guess.Charset....: -1 pPaAsSwWo0rRdD0123456789$, -2 Undefined, -3 Undefined, -4
Undefined
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 364.1 kH/s (11.09ms) @ Accel:128 Loops:32 Thr:1 Vec:8
Recovered.....: 0/1 (0.00%) Digests, 0/1 (0.00%) Salts
Progress.....: 34233472/152587890625 (0.02%)
Rejected.....: 0/34233472 (0.00%)
Restore.Point....: 2176/9765625 (0.02%)
Restore.Sub.#1...: Salt:0 Amplifier:1824-1856 Iteration:0-32
Candidates.#1....: $87r8678 -> dSDoRS12
```

マスクされたブルートフォース攻撃の実行 — この例は VM で実行しているため、復旧速度が非常に遅くなります。(*Screenshot hashcat hashcat.net/hashcat.*)

認証管理

ユーザーはしばしば、会社のネットワークと利用者用ウェブサイトに同じパスワードを使用するなど、管理が非常に困難で貧弱な認証情報管理の慣行を採用してしまいます。これによって企業のネットワークセキュリティは、これらのウェブサイトからのデータ侵害に対して脆弱になります。パスワードの認証管理ソリューションは、認証情報ストレージのプロキシとしてデバイスやサービスを使うことによってこのリスクを緩和します。マネージャーは、ウェブベースのアカウントごとに一意で強いパスワードを生成します。ユーザーは、マスター パスワードを使って各サイトを認証する権限をマネージャーに付与します。

パスワードマネージャーはハードウェアトークンと一緒に、またはソフトウェアアプリとして実装することができます。

- パスワード鍵 — PC やスマートフォンに接続する USB トークン。物理的に接続する他に、近距離無線通信 (NFC) または Bluetooth を使用できるものもあります (theverge.com/2019/2/22/18235173/the-best-hardware-security-keys-yubico-titan-key-u2f)。

- パスワードポールト（保管庫）— ソフトウェアベースのパスワードマネージャで、どのデバイスからもアクセスできるように一般的にはクラウドサービスを使用しています。[\(pcmag.com/picks/the-best-password-managers\)](https://www.pcmag.com/picks/the-best-password-managers)。USB鍵もポールトをバックアップとして使用することができます。多くのオペレーティングシステムやブラウザが、ネイティブパスワードポールトを実装しています。例として、Windows Credential ManagerやAppleのiCloud Keychainが挙げられます([imore.com/icloud-keychain](https://www.imore.com/icloud-keychain))。



認証管理プロダクトは、連邦情報処理標準(FIPS 140-2)に基づいて認証されます。これによって、暗号の実装が一定の堅固なレベルを満たしていることが保証されます。

レビュー アク ティビティ： ナレッジベース認証

次の質問にお答えください。

1. PINが「知っていることによる認証」の中で特に弱いタイプなのはなぜですか？
2. どのようなシナリオでPAPは安全な認証方法になり得ると考えられますか？
3. 次の記述は正しいですか、誤りですか？サービスチケットを作成するために、Kerberos認証では認証用のターゲットアプリケーションサーバーにユーザーのパスワードを渡します。
4. ユーザーは、コンピューターのディレクトリ構造内の深い所に配置されているファイルに、一般に使用されるパスワードのリストを保持しています。これは安全なパスワード管理と言えますか？
5. ブルートフォース攻撃を撃退するうえで、平文パスワードのどのプロパティが最も有効ですか？

トピック7C

認証技術を実装する



対象試験範囲

- 2.4認証と認可の設計コンセプトを要約することができます
- 3.3与えられたシナリオに基づいて、安全なネットワーク設計を実装 (HSMのみ)
- 3.8与えられたシナリオに基づいて、認証と認可のソリューションを導入することができます

認証技術は、ユーザーが持っているものや所有権/所有物を要素として使用できます。多くの組織が、スマートカードやUSB鍵フォブに基づいて、多要素認証システムを開発しています。そのため、仕事の中でこれらの技術の設置や構成をサポートしなければならない可能性があります。

スマートカード認証

スマートカード認証とは、安全な処理チップを搭載したカードに暗号情報をプログラミングすることです。チップには、ユーザーのデジタル証明書、証明書と関連付けられる秘密鍵、カードを有効にする際に使用する個人識別番号(PIN)が保管されています。

Kerberos認証の場合、スマートカードログオンは次の通り機能します。

1. ユーザーがスマートカードをリーダーに提示すると、PINの入力を促されます。
2. 正しいPINを入力すると、スマートカードの暗号プロセッサが秘密鍵を使用して、認証サーバ(AS)に送信されるTGT (Ticket Granting Ticket) の要求を作成することができます。
3. ASは、一致する公開鍵を持ち、ローカルCAまたは信頼できるルートCAであるサードパーティ CAから発行されたユーザーの証明書を信頼するため、リクエストを復号することができます。
4. ASは、TGTとTicket Granting Service (TGS)セッション鍵で応答します。



「スマートカード」とは、広範囲にわたる様々な技術を意味しています。Kerberosベースの認証を行なうには、暗号プロセッサを搭載したカードが必要です (smartcardbasics.com/smart-card-types.html)。

鍵管理デバイス

スマートカード認証に公開鍵インフラストラクチャ (PKI)を使用する場合、各ユーザーに発行された秘密鍵が安全であることが極めて重要です。一つの問題は、ユーザーだけが終始秘密鍵の所有者でなければならない点です。ネットワーク管理者がこれらの鍵を閲覧できると、あらゆる対象になりますことができます。管理者が秘密鍵を生成したり、ユーザーに転送する必要性を回避するために様々な技術を使うことができます。

- スマートカード — カードに組み込まれた暗号プロセッサを使って鍵のマテリアル (材料) を生成するのに十分な機能を有するカードもあります。
- USB鍵 — 暗号プロセッサは、USBフォームファクターに実装することもできます。
- トラステッドプラットフォームモジュール(TPM) — PC、ラップトップ、スマートフォン、またはネットワークアプライアンスに実装された安全な暗号プロセッサエンクレーブ (厳重に保護された領域・飛び地)。TPMは通常CPU内のモジュールです。TPMデータの修正は、信

頗度の高い手順によってのみ許可されます。TPMは、仮想スマートカードを提示する際に使用できます(docs.microsoft.com/en-us/windows/security/identity-protection/virtual-smart-cards/virtual-smart-card-overview)。

スマートカード、USB鍵、仮想スマートカードは、個々のデバイスとして設定します。多くの場合、鍵はサーバーやネットワークアプライアンスなどの非ユーザーデバイスにも設定する必要があります。**ハードウェアセキュリティモジュール(HSM)**は、ネットワークのデバイスに対してPKI管理を集中的に実行するために設計されたネットワークアプライアンスです。これは、損失または損害が発生した際に、鍵のアーカイブまたはエスクローとして機能します。証明書サービスに汎用サーバーを使用する場合と比べて、HSMは役割に最適化されているため、攻撃対象領域が小さくなります。HSMは、内部脅威のリスクを軽減するために、改ざんができないように設計されています。企業レベルの強度を確保する暗号論的に安全な疑似乱数生成器(CSPRNG)を提供することもできます。HSMは、ラックマウントアプライアンス、プラグインPCIeアダプターカード、USB接続型外部周辺機器など、いくつかのフォームファクターで実装することができます。



FIPS 140-2スキームでは、暗号論的に強力なプロダクトの認証を行います。
(ncipher.com/faq/key-secrets-management/what-fips-140-2.)



スマートカード、スマートカードリーダー、ハードウェアセキュリティモジュール (画像提供：© 123RF.com)

拡張認証プロトコル/IEEE 802.1X

先に説明したスマートカード認証プロセスは、コンピューターをローカルネットワークに設置し、ユーザーがWindowsにログインする際のKerberos認証に使用されます。認証は、次のような他のコンテキストにおいても必要な場合があります。

- ユーザーがワイヤレスネットワークにアクセスしていて、ネットワークデータベースに対して認証を行う必要がある場合。
- デバイスがスイッチを介してネットワークに接続していて、デバイスの通信を許可する前にユーザーを認証することがネットワークポリシーで定められている場合。
- ユーザーが仮想プライベートネットワーク(VPN)によりパブリックネットワークを介してネットワークに接続する場合。

これらのシナリオにおいて、**拡張認証プロトコル(EAP)**は、複数のタイプの認証プロトコルや技術を展開するためのフレームワークを提供します。EAPでは様々な認証方法が使用できますが、そのうちの多くがサーバーやクライアントマシンに対してデジタル証明書を使用します。これによってマシンは信頼関係を構築し、ユーザーパスワードを使用せずにユーザーの認証情報を転送したり、スマートカード認証を実行するための安全なトンネルを作ることができます。

EAPが認証メカニズムを提供する場合、**IEEE 802.1X**ポートベースのネットワークアクセスコントロール(NAC)プロトコルは、デバイスがイーサネットスイッチポート、ワイヤレスアクセスポイント（エンタープライズ認証で設定された）、またはVPNゲートウェイに接続する際に、EAP方式を利用する手段を提供します。802.1Xは、認証、認可、アカウンティング(AAA)アーキテクチャを使用します。

- **サプリカント** — ユーザーのPCまたはラップトップなど、アクセスを要求するデバイス。
- ネットワークアクセスサーバー (NAS) — スイッチ、アクセスポイント、VPNゲートウェイなどのエッジネットワークアプライアンス。これらは、RADIUSクライアントまたはオーセンティケータ(authenticator)とも呼ばれます。
- AAAサーバー — ローカルネットワーク内に配置された認証サーバー。

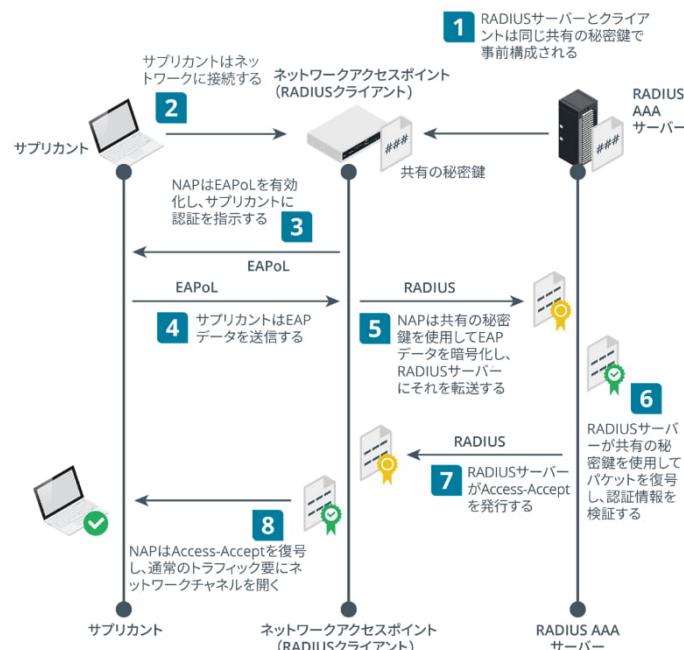
AAAを使用すると、NASデバイスは認証情報を保存する必要がなくなります。NASは、AAAサーバーとサプリカント間のデータ転送を行います。AAAサーバーには主に、次の2種類があります。RADIUSおよびTACACS+。

リモート認証ダイヤルインユーザーサービス

リモート認証ダイヤルインユーザーサービス(RADIUS)標準は、インターネットの標準として公開されています。RADIUSサーバーやクライアント製品には様々な種類があります。

NASデバイス (RADIUSクライアント) は、RADIUSサーバーのIPアドレスや共有の秘密鍵と共に設定されます。これによって、クライアントはサーバーに対する認証を実行できます。クライアントとはアクセスデバイス（スイッチ、アクセスポイント、またはVPNゲートウェイ）であって、ユーザーのPCまたはラップトップではありません。一般的なRADIUS認証ワークフローは次の通り行われます。

1. ユーザーのデバイス (サプリカント) は、アクセスポイント、スイッチ、またはリモートアクセスサーバーのようなNASアプライアンスに接続します。



EAPを用いたRADIUS認証の概要 (画像提供: © 123RF.com)

2. NASは、ユーザーに認証情報の提出を求めます。RADIUSは、PAP、CHAP、EAPをサポートします。PAPやCHAPは安全でないため、現在多くの実装でEAPが使用されています。EAP認証情報が必要な場合、NASはサブリカントが**EAP over LAN (EAPoL)**データを送ることを許可しますが、他のタイプのネットワークトライフックは許可されません。
3. サブリカントは認証情報をEAPoLデータとして提出します。RADIUSクライアントはこの情報を使用して、共有の秘密鍵を使って暗号化されたAccess-Request RADIUSパケットを作成します。RADIUSクライアントは、ポート1812（デフォルト）上のUDPを使って、AAAサーバーにAccess-Requestを送ります。
4. AAAサーバーは、共有の秘密鍵を使ってAccess-Requestを復号します。Access-Requestが復号できない場合（例えば、共有の秘密鍵が正しく設定されていないなどの理由で）、サーバーは応答しません。
5. EAPでは、認証方法の設定と認証情報の検証のために、Access-ChallengeとAccess-Request/パケットを交換します。NASはパススルーとして動作し、サーバーからRADIUSメッセージを取得し、サブリカントに転送するためにEAPoLとしてカプセル化します。
6. この交換の最後に、サブリカントが認証された場合は、AAAサーバーはAccess-Acceptパケットを使って応答し、サブリカントが認証されない場合は、Access-Rejectパケットが返されます。

オプションとして、NASはアカウンティング（ログイン）にRADIUSを使用できます。アカウンティングではポート1813を使用します。アカウンティングサーバーは認証サーバーと異なっても構いません。

Terminal Access Controller Access-Control System

RADIUSは主にネットワークアクセス制御に使われます。AAAサービスは、ネットワークアプライアンスの管理アカウントへのログインを集中管理する目的で使用することもあります。これによってネットワーク管理者に、スイッチ、ルーター、アクセスポイント、ファイアウォールに関する特別な権限が割り当てられます。RADIUSはネットワークアプライアンス管理ロールに使用することができますが、Ciscoが開発した**Terminal Access Controller Access-Control System Plus (TACACS+)**は、特にこの目的のために設計されています(<https://www.cisco.com/c/en/us/support/docs/security-vpn/remote-authentication-dial-user-service-radius/13838-10.html>)。

- TACACS+は、TCP通信（ポート49）を使用しますが、信頼できるコネクション型通信によって、サーバーのダウンを簡単に検出できます
- 認証データだけでなく、TACACS+パケット内のデータがすべて暗号化されます（パケットをTACACS+データとして識別するヘッダ以外）。これで、重要なネットワークインフラストラクチャデータを転送する際の機密性と完全性が確実に保持されます。
- 認証機能、認可機能、アカウンティング機能はそれぞれ別個のものです。多くのデバイス管理タスクは、再認証（sudoまたはUACのパスワードを再入力しなければならないのと同様）、コマンドごとの認可、ユーザー、グループ、役割ごとの特権を必要とします。TACACS+は、RADIUSよりも優れた方法でこのワークフローをサポートします。

トークン鍵と静的コード

スマートカード認証は、ネットワーク上で使用されるユーザーアカウントやデバイスを介して厳密に制御を行う際に有益に機能します。他のタイプの所有者ベースの認証技術では、さまざまなハードウェアやソフトウェアの**トークン**が使用されます。これらを使用することで、スマートカード認証で必要なデジタル認証を使用する際の管理の問題の一部を回避できます。

ワンタイムパスワード(OTP)は、ユーザーが選択するのではなく自動的に生成され、一度だけ使用します。そのため、パスワードの推測攻撃やスニッフィング攻撃などに対する脆弱性を克服できます。OTPは、シークレット値に対するある種のハッシュ関数と、タイムスタンプまたはカウンターなどの同期値（シード）を使って生成します。



キーフォルトーンのジェネレーター。(画像提供: © 123RF.com)

RSAのSecurIDトークンは、OTPトークン鍵の一般的な実装を表します。このデバイスは、現在の時刻とデバイス内にコード化して持っている秘密鍵を基にパスコードを生成します。ユーザーだけが知っているPINまたはパスワードと共にこのパスコードを入力します。ネットワークアクセスデバイスは、認証情報を横取りして検証用のAuthentication Managerサーバーに誘導するためのエージェントを構成する必要があります。このサーバーは、ADのようなディレクトリ製品と一体化できます。

また、デバイスにプログラムされた静的トークンを送信するだけの、より単純なトークン鍵やスマートカードもあります。例えば、多くのビルディングエンタリーシステムは、静的コードを基に動作します。これらのメカニズムは、クローニングやリプレイ攻撃に対して非常に脆弱です。

ハードウェアトークン鍵の実装方法は他にも複数存在します。例えば、Fast Identity Online (FIDO) Universal Second Factor (U2F) USBトークンは、認証サービスを使って公開鍵を登録します。次に認証メカニズムには、トークンに保存されている秘密鍵が必要で、PINまたは指紋アクティベーションを使って認可されます。fidoalliance.org/showcase/fido-u2f-security-key。これは、Windows Hello認証プロバイダーで使用することもできます(microsoft.com/security/blog/2019/06/10/advancing-windows-10-passwordless-platform)。

オープン認証

Initiative for Open Authentication (OATH)は、オープンで強い認証フレームワークを開発する目的で設立された業界団体です。オープンとは、様々なネットワークにわたってユーザーとデバイスの認証を実行する際に企業がリンクすることができるシステムであることを意味します。強いとは、システムがパスワードだけでなく、2要素または3要素認証、あるいは2段階認証にも基づいていることを意味します。OATHは、ワンタイムパスワード(OTP)を実装するための2つのアルゴリズムを開発しました。

HMAC-Based One-Time Password Algorithm (HOTP)

HMAC-based One-time Password Algorithm (HOTP)は、トークンベースの認証のためのアルゴリズムです(tools.ietf.org/html/rfc4226)。認証サーバートークンとクライアントトークンが、同じ共有の秘密鍵で設定されます。これは、暗号論的に強い乱数生成器によって生成された8バイトの値であるべきです。トークンは、フォブタイプのデバイスである場合や、スマートフォン認証/オーセンティケータアプリとして実装される場合があります。共有の秘密鍵は、電話のカメラで取得できるQRコード画像としてスマートフォンアプリに転送できるため、ユーザーは文字をタイプする必要はありません。もちろん、他のデバイスが共有の秘密鍵を取得できないことが重要です。ユーザーが認証するときに、共有の秘密鍵とカウンターが連結し、ワンタイムパス

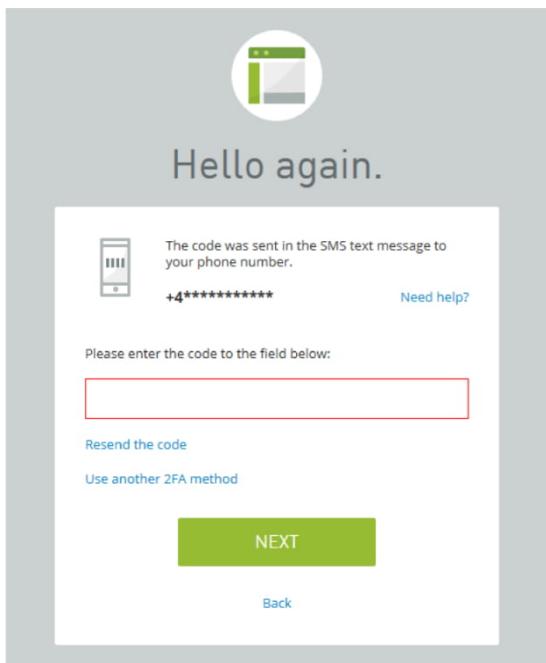
ワードが作成されます。デバイスとサーバーの両方でハッシュを計算し、6桁から8桁のHOTP値が抽出されます。これはサーバーで認証する場合、ユーザーが入力しなくてはならない値です。カウンターは1つずつインクリメントされます。



デバイスとサーバーカウンターが同期していない環境に対応するように、カウンターウィンドウを付けてサーバーを構成します。これは、例えばユーザーがOTPを生成したのに使用しない場合に生じます。

Time-Based One-Time Password Algorithm (TOTP)

Time-based One-time Password Algorithm (TOTP)は、HOTPの改良版です(tools.ietf.org/html/rfc6238)。HOTPに関する問題の一つは、トークンが有効期限が切れることなく持続可能なため、脅威アクターがトークンを取得してデータを復号するリスクが将来的に生じる可能性があることです。TOTPでは、HMACは共有秘密鍵と、デバイスとサーバーのローカルタイムスタンプから抽出した値で構築します。TOTPでは、短いウィンドウ（例えば60秒）の後、各トークンの有効期限が自動的に切れます。この機能を動作させるために、クライアントデバイスとサーバーは厳密に時間同期されます。HOTPとTOTPの実装としてはGoogle Authenticatorが良く知られています。



ウェブアプリケーションアクセスを保護する2段階認証メカニズム。このサイトは、5分の有効時間を持つTOTPパスワードを、登録した携帯電話にSMSで送ります。



OATH (Open Authentication)とOAuth (Open Authorization)を混同しないようにします。

2段階認証

2段階認証またはアウトオブバンドメカニズムは、サーバー上でソフトウェアトークンを生成し、ユーザーが安全に管理していると思われるリソースに送ります。このトークンをデバイスに転送する方法は、次のとおりいくつか存在します。

- ショートメッセージサービス(SMS)コードは登録した電話番号にテキストとして送られます。
- 電話コードは登録された電話番号に自動音声通話として送信されます。
- プッシュ通知コードはPCまたはスマートフォンの登録したオーセンティケーターアプリに送信されます。
- 電子メールコードは登録された電子メールアカウントに送信されます。

これらのメカニズムは2要素認証(2FA)と呼ばれることもあります。しかし、時間フレーム内に誰かがコードを傍受すると、所有したことを見たこともないデバイスでも、ユーザーが知っている情報として入力される可能性があります(auth0.com/blog/why-sms-multi-factor-still-matters)。

レビュー アク ティビティ：

認証技術：

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？スマートカードログオンを実施すると、ユーザーの秘密鍵がスマートカードに保管されます。
2. あなたは、構内ネットワークやクラウドサービスにスマートカード認証を実装するよう企業にアドバイスしています。サーバーベースの鍵と証明書管理サービスによってHSMを使用する主な利点はなんですか？
3. スマートカードは、どのネットワークアクセス制御フレームワークでサポートされていますか？
4. RADIUSクライアントとはなんですか？
5. EAPoLとはなんですか？
6. OTPは、パスワード推測攻撃またはスニッフィング攻撃からどのように保護しますか？

トピック7D

生体認証概念を要約する



対象試験範囲

2.4認証と認可の設計コンセプトを要約することができる

生体認証メカニズムでは、ユーザーは生理的特徴（指紋または虹彩パターンなど）または行動パターンを使ってアカウントにアクセスできます。生体認証メカニズムの利点と欠点を要約できること、これらの技術の展開や使用をサポートできるようになります。

生体認証

生体認証を設定する第一のステップは登録です。選択した生体認証情報を生体認証リーダーでスキャンし、バイナリ情報に変換します。スキャンプロセスには一般的に2つのステップがあります。

- センサー モジュールがターゲットから生体認証のサンプルを取得します。
- 特徴抽出 モジュールがターゲットを一意に識別するサンプルの特徴を記録します。

生体認証のテンプレートは、認証サーバーのデータベースに保存されます。ユーザーがリソースにアクセスすると、ユーザーが再スキャンされ、テンプレートと照合されます。これらが定義された許容度内で一致すると、アクセスが許可されます。

生物測定学的に人を識別する場合は、いくつかのパターンタイプが使用できます。これらは、身体的（指紋、目、顔認識）または行動（声、署名、タイピングパターンの一一致）に分類できます。生体認証パターンの取得や照合の有効率や、認証メカニズムとしての適合性を評価するために用いられる、主な指標や考慮事項は以下のとおりです。

- 本人拒否率(False Rejection Rate, FRR)** — 本物のユーザーが認識されない比率を表します。これは、タイプIエラーまたはFNMR (false non-match rate)とも呼ばれます。FRRは、パーセンテージで算出されます。
- 他人受入率(False Acceptance Rate, FAR)** — 侵入者が受け入れられる比率を表します（タイプIIエラーまたはFMR, false match rate）。FARは、パーセンテージで算出されます。

本人拒否はユーザーに不便をかけるが、他人受入はセキュリティ侵害につながるため、通常、最も重要な指標と考えられています。

- クロスオーバーエラー率(Crossover Error Rate, CER)** — FRRとFARが一致する点です。CERが低いほど、技術の有効性と信頼性が高まります。

システムを調整することで、経時的にエラーが減少します。これは一般的に、CERが十分な値になるまでシステムの感度を調整することによって実現します。

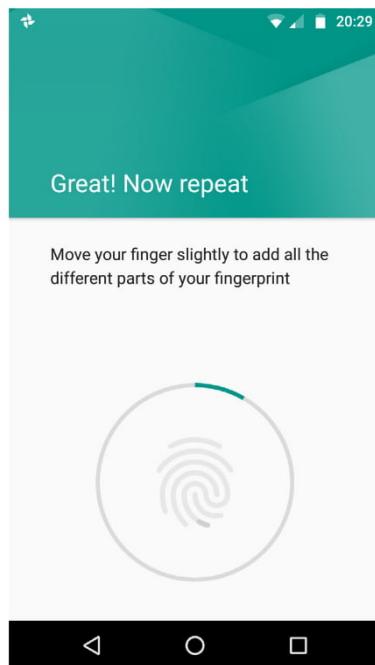
- スループット（スピード）** — ユーザーごとのテンプレート作成と認証に要する時間。これは主に空港や駅など、トラフィックの多いアクセスポイントについて検討します。

- 登録未対応率(Failure to Enroll Rate, FER) — テンプレートが作成できない、登録中にユーザーとマッチしないなどのインシデントの発生率。
- 費用/実装 — 一部のスキャナータイプには費用がかかり、他のタイプのものはモバイルデバイスへの組み込みが容易ではありません。
- ユーザーは煩わしいと感じたり、プライバシーの侵害だと感じる場合があります。
- この技術は、障害者にとって差別的であったり、アクセスしにくいものであったりします。

指紋認証

生理学上の生体認証の特性は、「Something You Are」（身体的特徴）要素を表します。これには、指紋のパターン、虹彩あるいは網膜認証、顔認証が含まれます。

指紋認証は、最も広く実施されている生体認証法です。指紋のスキャンや記録に必要な技術は比較的安価で、そのプロセスもかなり単純です。指紋センサーは通常、小容量セルとして実装され、パターンを構成している隆線から一意のパターンを検出することができます。この技術はあまり煩わしさは感じられず、使用も比較的簡単ですが、湿気や汚れが読み取りを妨げる可能性があります。



Androidスマートフォンに指紋認証を設定。
(AndroidはGoogle LLCの商標です)

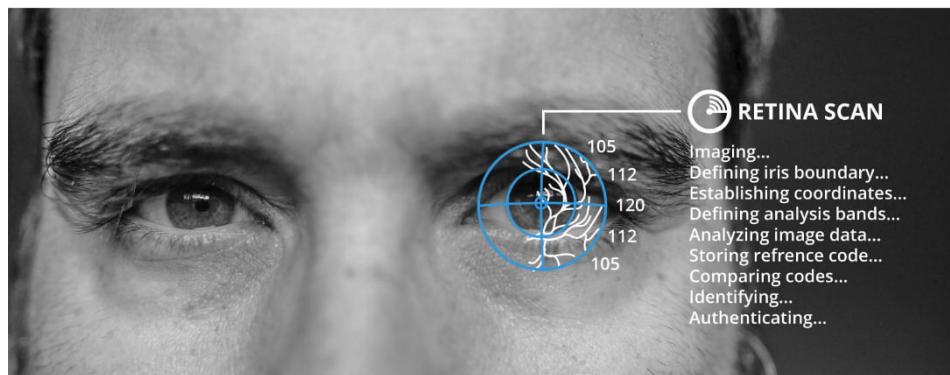
指紋スキャナーの主な問題点は、ユーザーの指紋のコピーを入手したり、スキャナーをだますための指紋の型を作成することができることです(tomsguide.com/us/iphone-touch-id-hack-news-20066.html)。しかし、静脈マッチングスキャナーや血管による生体認証が、こういった懸念を払しょくしてくれます。この場合、人間の指または手のひらの血管の一意のパターンからテンプレートを作成するため、赤外線の光源やカメラなどのさらに複雑なスキャナーが必要になります。

顔認証

顔認証では、目と目の間の距離、または鼻の幅や長さなど、顔のサイズや形に関する複数のインジケータを記録します。最適な照明条件下で最初のパターンを記録する必要があり、技術によっては、この作業に時間がかかります。さらに、この技術は法執行機関との関連が深く、個人のプライバシーの問題でユーザーを不快にさせる可能性が最も高い技術です。顔認証は、他人受け入れ率や本人拒否率が比較的高いため、スプーフィング攻撃を受ける可能性があります。認証目的よりも監視目的での技術開発が進んでいますが、スマートフォンに使用する方法が一般的になりつつあります。

顔認証の弱点は、さらに詳細に目の特徴をスキャンすることによって克服できます。

- 網膜スキャン—血管パターンを識別する際は、赤外線を目に照射します。血管の配列は非常に複雑で、通常、病気や怪我などの事情がない限り、生涯変わることはありません。したがって、網膜スキャンは最も正確な生体認証の形式です。網膜パターンは非常に安全ですが、必要な設備が高額で、プロセスも比較的煩わしく複雑です。白内障などの病気が原因で、検知漏れが起こる可能性があります。



網膜スキャンでは、赤外線を使って目の中の血管パターンを識別します。

(写真提供 : Ghost Presenter on Unsplash)

- 虹彩スキャン—近赤外イメージングを使って、目の表面のパターンを一致させるため、網膜スキャンより煩わしさが少なく（例えば、対象者は眼鏡をかけたままでよいなど）、迅速です。虹彩スキャンの正確性は網膜スキャンと同程度ですが、病気の影響を受ける可能性ははるかに低いです。虹彩認証は、空港のセキュリティなど、大量に使用される場合に最も適している技術です。しかし、他人の高解像度写真を使って虹彩スキャナーが騙される可能性はあります。

行動認証技術

「Something you do」は行動による生体パターン認証です。身体の属性をスキャンするのではなく、タイピング、署名の仕方、歩き方/動き方などの行動を分析してテンプレートを作成します。動きや圧力、歩き方のバリエーションで個人を一意に特定すると考えます。しかし実際のところ、これらの方の誤り率は高く、対象者に対して実行するには多くの問題があります。

- 音声認証—必要なハードウェアやソフトウェアが多くの標準PCやモバイルに搭載されているため、比較的安価で実装できます。しかし、正確なテンプレートの取得が難しく、時間もかかります。背景の雑音や他の環境要因がログオンを妨げる可能性もあります。声もまた、なりすましが可能です。
- **歩行分析**—人間の動き（移動）からテンプレートを作成します。この技術には、カメラを使用するものと、加速度センサーやジャイロスコープなどのスマートフォンの機能を使用するものがあります。
- 署名認証—署名は比較的簡単に複製できますが、正確な署名プロセスを模倣するのは簡単ではありません。署名マッチングは、署名を適用するユーザーを記録します（ストローク、スピード、筆圧）。
- タイピング—ユーザーがパスフレーズを入力するスピードとパターンを一致させます。

一部の生体認証技術や行動認証技術はログオン認証以外の目的で使われる可能性があります。

- 生体認証識別とは、ある意味で認証とは逆に、データベースと人を一致させることです。例えば、データセンターの床を歩いている個人が歩行分析と一致しない場合、システムはセキュリティ警告を発します(g4s.com/en-us/media/news/2017/12/06/keeping-data-centers-secure)。
- 繙続的認証では、ログオンしたユーザーがまだデバイスを操作していることを確認します。例えば、ユーザーが指紋を使ってスマートフォンに対する認証に成功した場合、デバイスはキーの動きやデバイスを持って操作する際の圧力の統計値の監視を継続します。これがベースラインから外れると、検知システムが電話をロックします。この類の技術は現在市場に出ていませんが（執筆時点）、多くの研究プロジェクトで研究されています。

レビュー アク ティビティ： 生体認証の概念

次の質問にお答えください。

1. 生体認証技術を評価する際の主な検討事項は、費用以外にどんなものがありますか？
2. 指紋リーダーは通常ハードウェアとしてどのように実装されていますか？
3. 網膜スキャンと虹彩スキャン、どちらのタイプの目の認証の実行が簡単ですか？
4. ログオン認証以外で生体認証技術を使用する方法を2つ挙げてください。

レッスン7

概要

オンプレミスネットワーク、ウェブ/クラウドアプリ、物理的セキュリティのための認証プロダクトを、機密性、完全性、可用性の要件の観点から設計を評価して使用できる必要があります。指定のプロダクトの設定ガイドに基づいて、Kerberos、スマートカード認証、EAP/RADIUSなどのプロトコルや技術を実装できる必要があります。また、パスワード攻撃の兆候やリスクを識別する必要があります。

認証制御を実装するためのガイドライン

認証制御を実装する際は次のガイドラインに従います：

- 認証ソリューションのコンテキストに基づいて、機密性、完全性、可用性の設計要件を評価します（プライベートネットワーク、パブリックウェブ、VPNゲートウェイ、物理的なサイトの建物など）。
- 多要素認証(MFA)の必要性、および知識要素と組み合わせる場合、どのハードウェアトークンや生体認証技術が要件に合うかを決定します。
 - 所有要素には、スマートカード、OTP鍵/フォブ、または信頼できるデバイスにインストールされたOTP認証アプリが含まれます。
 - 生体認証技術には、指紋、顔、虹彩、網膜、音声、静脈などがあり、FAR、FRR、CER、速度、アクセス性などの指標で有効性が判断されます。
 - 2段階認証は、SMS、電話の呼び出し、電子メール、プッシュ通知を介して信頼できるデバイスやアカウントに追加のトークンを提供することができます。
 - 保管庫とUSB鍵/ワイヤレスフォブによって、パスワード認証がより安全なものになります。
- 適切な認証プロトコルまたはフレームワークを選択します：
 - ローカルネットワークへのサインインには、スマートカード認証をサポートしているKerberos認証。
 - ネットワークアクセスデバイスでの認証には、スマートカード認証またはユーザーの認証情報の安全な転送をサポートする802.1X/EAP/RADIUS。
 - ネットワークアプライアンス管理にはTACACS+。
- 特にレガシープロコトル（PAPやCHAP）を使用し、ハッシュがキャプチャされる場合には、パスワード攻撃によるリスクを評価します。

レッスン8

アイデンティティとアカウント管理制御の実装

レッスン概要

管理するネットワークやデバイスに正規のユーザーとデバイスのみが接続できるようにすると共に、そのような対象には、リソースへのアクセスや変更をするために必要な許可や権限のみを与えるようにする必要があります。このようなタスクでは、オンプレミスネットワークとクラウドサービスの全体でアイデンティティを管理する必要があるため、複雑さが増しています。またアカウントのセキュリティは、人事とセキュリティに関するトレーニングの効果的な組織的ポリシーにかかっています。ベストプラクティスに沿ってこれらのポリシーの策定や更新に関わる機会や、セキュリティへの意識向上教育やトレーニングプログラムを提供する機会も頻繁におとずれます。

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。

- アイデンティティとアカウントタイプの導入
- アカウントポリシーの導入
- 認可ソリューションの導入
- 人事ポリシーの重要性の説明

トピック8A

アイデンティティとアカウントタイプを導入する



対象試験範囲

- 3.7 与えられたシナリオに基づいて、アイデンティティとアカウント管理制御を実装できる
5.3 組織のセキュリティに関するポリシーの重要性について説明することができる

最小限の特権は、ほとんどの組織のセキュリティポリシーの中核をなす原則です。アイデンティティと権限を管理することで、組織は一般ユーザーと管理者ユーザーの両方のアクションを把握できるようになります。アカウントにはデフォルト、共有、ゲスト、デバイスなどの種類があり、それらを1つのアイデンティティに紐づけることが困難なため、これらのシステムは複雑化しています。

アイデンティティ管理制御

プライベートネットワークでは、デジタルアイデンティティはアカウントによって表されます。ネットワーク管理者がアカウントをホスティングするサーバーの完全性を確保し、各ユーザーは認証情報を保護する責任を負います。これによって、本人のみがアカウントに認証され、使用できるようになります。またアカウントは、パブリックネットワーク上や、プライベートネットワークの外部保護レイヤーとして、一部の暗号化マテリアルによって特定される場合もあります。

証明書とスマートカード

検証済みの対象（ユーザーやサーバー）に対して認証機関(CA)が証明書を発行する場合、公開鍵インフラストラクチャ(PKI)を使用してデジタルアイデンティティを管理できます。そのCAを信頼する任意のサードパーティは、この対象のアイデンティティも信頼することができます。

証明書は対象の公開鍵を含み、CAの秘密鍵によって署名されます。これらの公開鍵を使用することで、サードパーティがその証明書と署名を検証できます。対象の公開鍵は、リンクされた秘密鍵との対の片方です。秘密鍵は、機密として保持する必要があります。コンピューターのファイルシステムか、トラステッドプラットフォームモジュール(TPM)に保存することができます。また、ユーザーの証明書や秘密鍵はスマートカードやUSB鍵に保存することや、別のPCやモバイルデバイスの認証に使用することもできます。

トークン

ユーザーにとっては、自分が使用するアプリケーションをいちいち認証するのは面倒です。シングルサインオンシステムでは、ユーザーはIDプロバイダー(IdP)で認証を受け、暗号トークンを受け取ります。ユーザーは自分が認証されていることの証拠として、そのトークンを対応するアプリケーションに提示し、そのアプリケーションから認可を受けることができます。トークンには、常に悪意のある脅威アクターに取得され、再使用されるリスクが伴います。トークンを活用するアプリケーションプロトコルは、この種類の攻撃に対抗できるように設計する必要があります。

IDプロバイダー

IDプロバイダーは、ユーザーアカウントを提供し、認証リクエストを処理するサービスです。プライベートネットワークでは、このようなアイデンティティディレクトリやアプリケーション認可サービスをローカルで運用しています。同じサイトで、アイデンティティ提供とアプリケーション