RAPID7

RESEARCH

UNDER - THE - HOODIE 2020

By Rapid7 Global Consulting

目次

エグゼクティブサマリー

ペネトレーションテスト（侵入テスト）とは、セキュリティ対策の防御面で問題があるかどうか発見して修正するために、機密性の高い領域に対するセキュリティ侵害をシミュレーションする行為です。評判の高い侵入テスト企業であるCoalfire Systems社の従業員が侵入テストの範囲と基本的な正当性について誤解を受けて2019年9月に逮捕されたことからも、未だにかなりオカルト的なテーマと言えます。12020年2月に最終的に告訴は取り下げられましたが、この事件で、侵入テストの世界は大きく揺るがされました。明らかに、Offensive Securityに携わるすべての人が、物理世界と仮想世界で日常的に実施されている侵入テストの価値について、世の中から理解を得るために努力する必要があるのです。

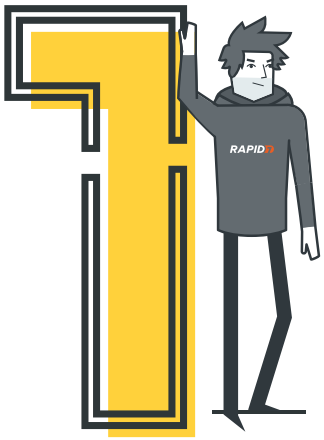
本レポートの目的は、主に内部および外部のネットワーク侵害をカバーする侵入テストの方法と理由を調査し、ソーシャルエンジニアリングとレッドチームシミュレーションに関する補足データを提供することにあります。12か月分の調査データの過程で、次のことがわかりました。

- 内部ネットワーク構成とパッチ管理は引き続き「容易な」ソフトターゲットをペネトレーションテスターに提供しています。ペネトレーションテスターは、市販の商品攻撃を使用して特権をエスカレートし、検知されずにネットワーク内を横方向に移動できます。
- 企業レベルでのパスワード管理と2要素認証（2FA）などの二次的な制御が大幅に不足しているため、シミュレートされたセキュリティ侵害の際に取得されたパスワードスプレーとハッシュされたパスワードの復号化の双方に対する「容易な」コンプロマイズにつながります。
- 世界のナレッジワーカーは、オンサイトの従来の内部ネットワークコントロールではなく、VPNとインターネットベースのアプリケーションにこれまで以上に依存しているため、侵入テスターは、これらのVPNターミネーターとカスタムWebアプリに重大な欠陥を見つけています。

これは、皮肉を込めて言いますが、ほとんどのポリアンナ的な（小学生レベルの）セキュリティ研究者にさえ衝撃的ではありませんが、世界中の企業が次の侵入テストに何を期待すべきかを理解し、事前に何を調査して修正すべきかをまとめたチェックリストとして利用するのに役立つ確かなデータです。

最後に、このレポートの調査データは2019年6月から2020年6月まで収集され、合計206のエンゲージメントから回答が収集されています。世界で最も多忙を極めるペンテスターは、1年あたり25回弱の侵入テストに関与しているという傾向があります（約8〜10日の平均エンゲージメント時間を想定）。このため、本レポートは、業界全体の傾向について関心のあるすべてのペンテスターに役立つはずです。

[1] https://www.theregister.com/2020/08/05/coalfire_pentest_iowa_black_hat/



スコープ、計画、および実行

今年のUnder the Hoodie調査の結果に入る前に、まずいくつかの用語を定義します。結局のところ、本ホワイトペーパーの目的は侵入テストをわかりやすく説明することですので、まずは専門用語の説明に行きましょう。

コードレビュー

コードレビューは、エンゲージメントの中で最も「実稼働中」という形式になりやすい傾向があります。また、侵入テスト担当者が完全またはほぼ完全にアクセスできる1つのアプリケーションまたは製品のみに焦点を当てる傾向があります。このエンゲージメントの目標は、テスト中のシステムまたはデバイスのプロアクティブな品質保証レビューに最も類似しています。侵入テスターの目標は、そのシステムまたはデバイスの予期しないセキュリティ関連のバグを見つけることにありです。

内部ネットワークの評価

内部ネットワークの評価は、クライアントの内部ネットワークの包括的な調査です。これには、LANで「ローカル」に表示されると予想されるシステムのタイプ（Windowsクライアントとサーバー、プリンター、内部アプリケーション、およびWi-Fiアクセスポイント）が含まれる傾向があります。多くの場合、テスターはオンサイトで招待されるか、VPN接続を介して内部アクセスポイントにアクセスできます。内部ネットワーク評価の最終的な目標は、やる気のある悪意のある内部関係者によって侵害されたり、他の方法で内部ネットワークの存在を獲得した攻撃者によって悪用されたりする可能性がある、内部ネットワークに固有の情報セキュリティリスクを文書化して実証することです。もちろん、これは脆弱性が外部から悪用される可能性があることを意味します。

外部ネットワークの評価

外部評価では、内部評価と同様に、クライアントのITインフラストラクチャの脆弱性と構成ミスを探します。しかし、その名前が示すように、実際には外部の攻撃者から見ることができる到達可能なものだけです。したがって、これらのエンゲージメントは、多くのWebアプリケーションハッキング、およびインターネット全体に公開されているサービス（VPNエンドポイント、電子メール、および偶然に外部に公開されているすべてのサービス）のマッピングと悪用を特徴としている傾向があります（私のDMZ内のTelnetコンソール？思っているよりも可能性は高いです！）。

ソーシャルエンジニアリングのエンゲージメント

ソーシャルエンジニアリングのエンゲージメントには、ほとんどの場合、電子メールコンポーネントが含まれています。これは、パスワード情報を抽出する方法（多くの場合、標的となる被害者が向けられている偽のログインページなど）または外部の攻撃者を許可する被害者のコンピューターに初期マルウェアをドロップするための好ましい方法です。多くの場合、コンサルタントは物理的なソーシャルエンジニアリングの関与を課されます。ここでのタスクは、クライアントの組織の物理インフラストラクチャの制限された領域または機密領域に侵入することです。ロックされたネットワーククローゼット、製造現場の制限された領域などです。

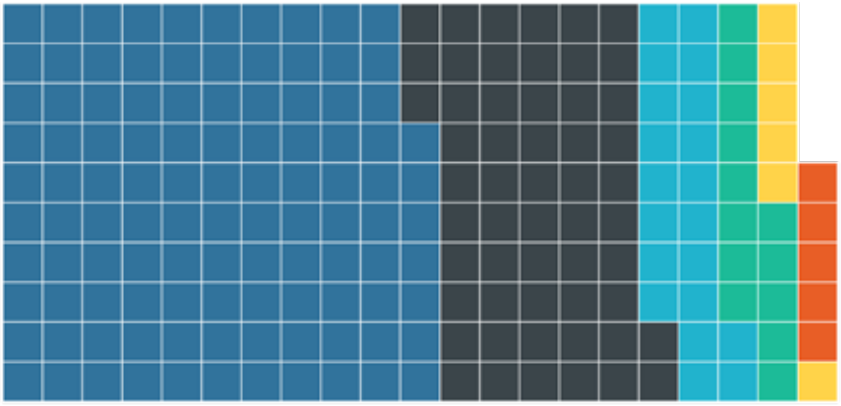
レッドチームのシミュレーション

レッドチームの演習はかなり複雑な作業であり、ソーシャルエンジニアリング演習（主要な内部関係者の信頼を得るため）、外部ネットワークアセスメント（インターネットに公開された露出とネットワークの脆弱性を計画して悪用するため）および内部ネットワークの評価（特権を外部の見知らぬ人から内部のスーパーユーザーに昇格させるため）の要素を組み合わせている傾向があります。これらはすべて、クライアントの通常の検知および防御テクノロジーがアクティブなときに発生します。ネットワークの評価を150ポイントの自動車検査と見なすことができ、システム全体を評価する場合、レッドチームエンゲージメントはクラッシュテストに似ています。すべてのシステムは、実際の攻撃に最も近い演習によりテストされます。

エンゲージメントタイプと比較

それぞれのエンゲージメントタイプの違いを定義しておくと、「内部」ジョブと「外部」ジョブのより広範なカテゴリ、およびクライアントが各種類をリクエストする頻度を確認できます。図1は、調査された一連のエンゲージメントのエンゲージメントタイプ別の内訳を示しています。

図1：実施された侵入テストの評価



出典：Rapid7

これらは、「内部ジョブ」と「外部ジョブ」の2つに大まかに分類されると考えられます。内部ジョブの場合、攻撃者は、内部ネットワークまたはソースコードへのアクセス、またはテスト対象のデバイスへの物理的アクセスを利用することで既に有利な状態となります。**コードレビュー**（2.43%）と**内部ネットワーク**（26.70%）を組み合わせると、すべてのエンゲージメントの29.13%がジョブ内にあることがわかります。さて、これは不公平に聞こえるかもしれませんが、組織は、内部の攻撃者が「不満を持つ従業員」である場合と、「外部」から「内部」に入り込んできた攻撃者の場合の双方に対して、あらゆる手段を通じて防御能力をテストしておくことが賢明です。

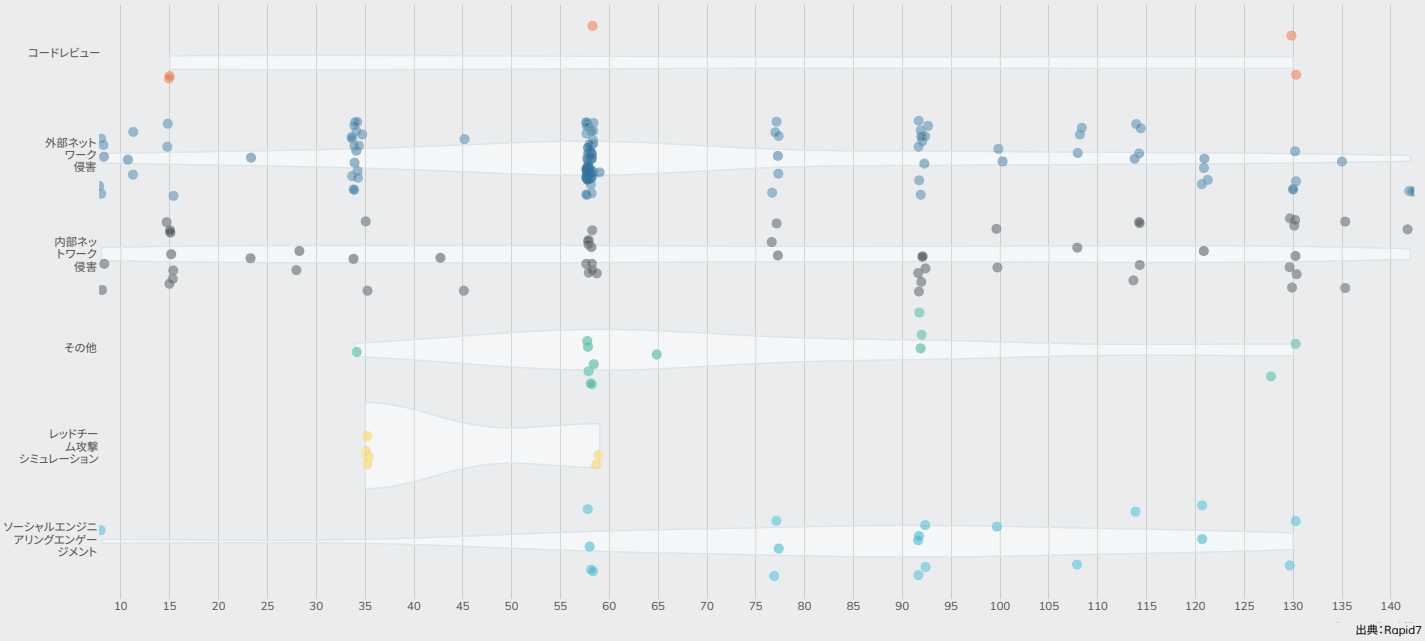
「外部ジョブ」は、従来の外部の攻撃者が操作する場所です。**外部ネットワーク**（51.94%）、**レッドチーム**（2.91%）、**ソーシャルエンジニアリング**（9.71%）はすべて「外部」からのものであり、合計で最大64.56%です。これは何を意味するのでしょうか？ペンテストクライアントは、内部の脅威よりも外部の脅威に関心がある傾向があります。実際、今年は、昨年よりも外部からの評価を好んでいます。昨年の「外部」バケットは、すべてのエンゲージメントの最大50%であり、「内部」は39.4%を占めています（両方の年で、「その他」はほぼ変わらなかった） が8%です。Rapid7のデータはパンデミックへの懸念で生じた世界的な封鎖の期間をカバーしているため、昨年よりも内部よりも外部からの攻撃を重んじる傾向が今年以上に高まることが確実に予想されます。

タイミングに関する考察

今までのUnder the Hoodieレポートでは、Rapid7またはRapid7が雇った下請業者によって実行された侵入テストのみを検討しています。侵入テストプロバイダー（n = 1）には多様性がないため、契約時間数はかなり標準化されていると予想されます。過去にも、これとまったく同じ傾向が見られました。エンゲージメントは、契約時間数の80時間のマークに正しく到達する傾向があり、作業に割り当てられた時間にわずかな変動しかありませんでした。

ただし、図2に示すように、今年のデータでは、契約時間に多様性が見られます。

図2：評価タイプごとのエンゲージメントの時間

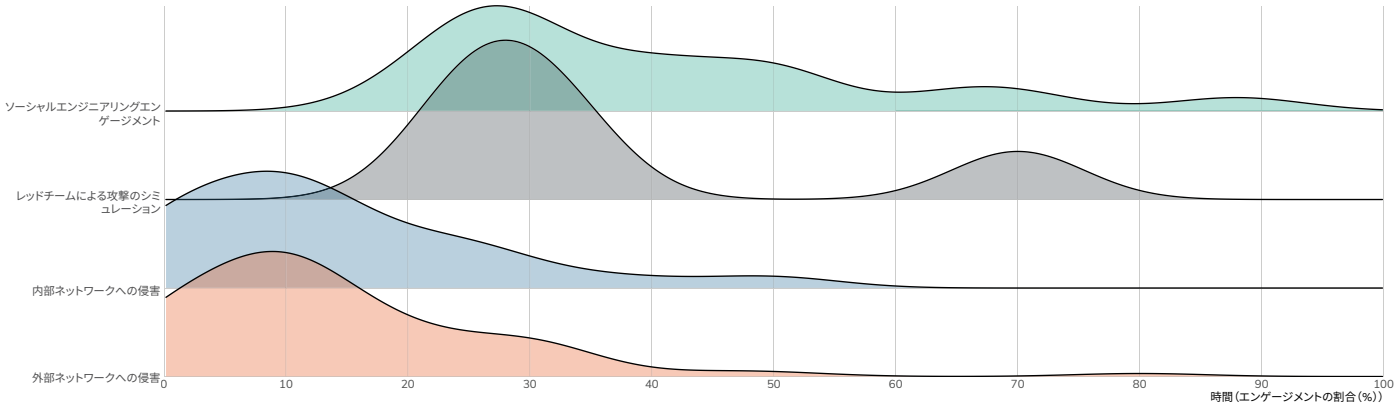


ここでは、明らかに望ましい契約時間に多くが集まっているものの、すべてのエンゲージメントタイプ全体で合計時間がかなり多様なカテゴリにあることがわかります。結局のところ、各クライアントのエンゲージメントはかなり独特であり、すべてのエンゲージメントを一緒くたにまとめることができる、万能のタイムボックスなどは実際にはありません。実際のところ、契約時間のこの多様性は、クライアントとペネトレーションテスターの間の最初のスコーピングと計画に関する打ち合わせがいかに重要か物語っています。特定のエンゲージメントは、約150時間までの約35時間の作業に相当します。それはすべて、ネットワークの複雑さとサイズ、そしてクライアントがどこまで深堀したいかによって異なります。

あまり変わらないのは、特定のクライアントがネットワーク上の偵察の計画と実行に費やした時間の割合です。ソーシャルエンジニアリングとレッドチームのより複雑なエンゲージメントタイプでは、総時間の25%がこれらの攻撃前の活動に費やされていますが、内部および外部のネットワーク評価では、おそらく10%の時間が偵察に費やされています。

図3：計画と偵察に費やされたエンゲージメントタイプ

所要時間の詳細が提供されたエンゲージメントのみのデータセット（N = 148）



ペンテストの実際：

法律事務所に電話！

By Jonathan Stines

法律事務所のための電話による虚偽の口実のエンゲージメントの一環として、機密情報の特定、資格情報の取得、およびマシンでのリバースシェルの取得を目的として電話をかけるために従業員の電話番号が提供されました。

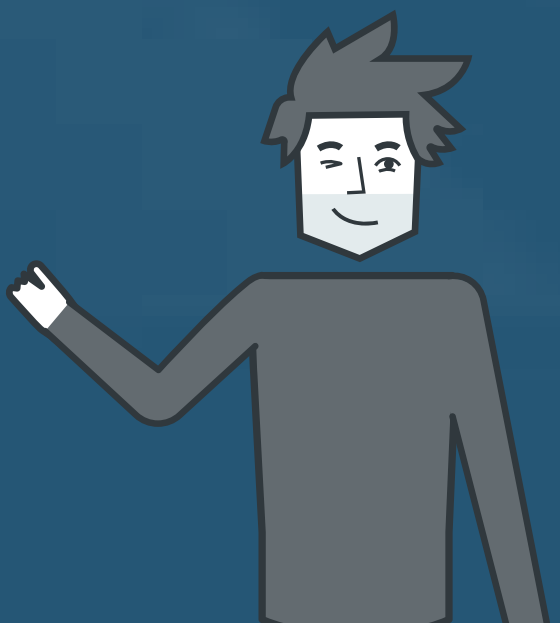
ITアナリストを装って担当者に最初に電話をかけ、その担当者との信頼関係を築き、苦情について話を聞くために、それまでユーザーが経験していた問題点について一般的な質問をしました。結局のところ、ITについて不満を言うのが大好きな人が多いのです！信頼関係を構築したら、次に、使用中のVPNテクノロジー、エンドポイント保護、使用中のバージョン、そしてその後に、別の虚偽の口実の一部として使用できる他の使用中のテクノロジーに関するより鋭い質問に移りました。

エンドポイント保護ソリューションのベンダーとバージョンを知ったので、その特定のセキュリティツールを回避するように設計されたペイロードを作成しました。そして、会社のリモートアクセスVPNクライアントが古いバージョンで更新が必要なユーザーリストに対して、ITアナリストになりすまして伝える虚偽の口実を思いつきました。

そしてその従業員に電話し、VPNクライアントのバージョンが古く、複数の既知の脆弱性があり、更新する必要があることを説明しました。次に、スタートを押してCMDを実行し、ステージャーワンライナーを実行するように指示しました。これが行われると、ワンライナーはホストされたペイロードアセンブリをダウンロードしてユーザーのメモリで実行し、コマンドアンドコントロール（C2）サーバーへのリバースシェルを確立して、ワークステーションへのフルアクセスを可能にします。

次に、ネットワークへの内部アクセスを使用して、ドメイン管理者などのセキュリティグループのメンバーの特定やドメインコントローラーの場所の特定など、高度な偵察を行いました。さらに、「Kerberoasting」と呼ばれる手法を実行しました。これは、ドメインのサービスアカウントのKerberos認証ハッシュを返しました。テスト中に、特権の高いDomain Adminsセキュリティグループのメンバーであったアカウントの Kerberosハッシュの1つをクラックしました。

この時点で、お客様にどこまで行けたのか説明して、さらなる指示を仰ぎました。電話による虚偽の口実に対するリスクを十分に示すことができたため、電話をかけるのを直ちに中止するよう指示を受けました。



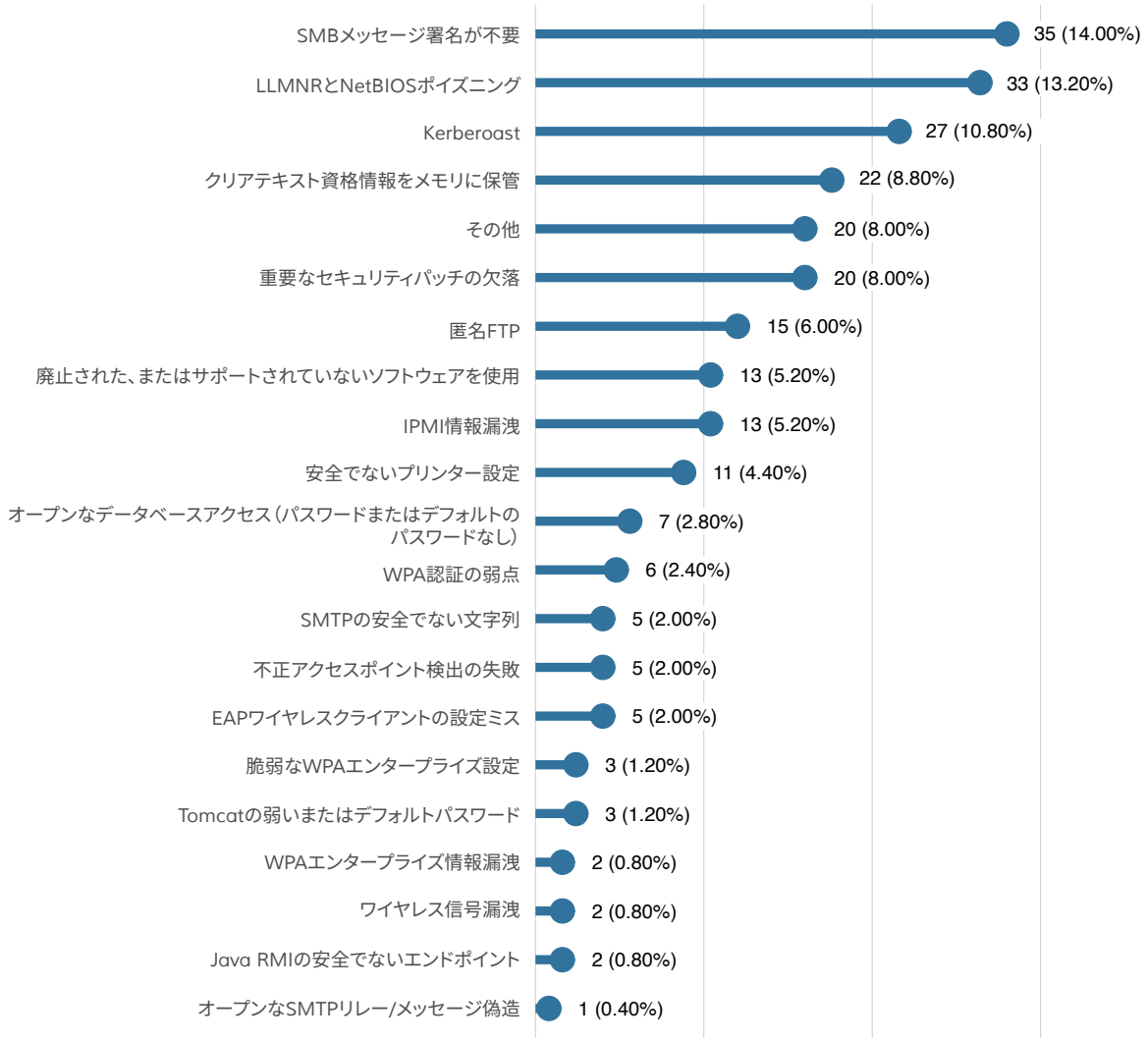
脆弱性の深堀調査

ペンテスターが見つけて悪用する脆弱性の種類は、実行される侵入テストの種類に大きく依存します。このセクションでは、これらの脆弱性プロファイルを、内部のジョブ（内部ネットワークの評価とコードレビュー）と外部のジョブ（外部のネットワーク評価、ソーシャルエンジニアリングエンゲージメント、レッドチームのシミュレーション）で比較します。

内部ジョブ

内部ネットワークの評価は、結果として、Windowsベースの攻撃にかなり重点を置いていますが、それには正当な理由があります。結局のところ、Windowsクライアントは、業界やターゲットのサイズに関係なく、エンタープライズ環境にかなり多く存在しているため、意欲的で実践的なペンテスターも、必ずツールと経験を通常のWindows環境に合わせるようにしています。図4は、内部環境で成功する最も一般的なタイプの攻撃の詳細を示しています。

図4：内部エンゲージメント： 発見した脆弱性の種類



出典：Rapid7

どこにでもあるWindows

これらの内部エンゲージメントで発生した最も一般的な3つの脆弱性は、**SMBメッセージ署名が不要**（14%）、**LLMNRとNetBIOSポイズニング**（13.2%）、および**Kerberoast**　（10.8%）です。これらをまとめると強力な脆弱性となっており、攻撃者は、迅速にドメイン管理者（DA）特権に昇格できるようになります。

1つ目のSMBメッセージ署名に関しては、MicrosoftとMicrosoft以外のSMBクライアントの混合環境で一般的です。SMBメッセージ署名は、SMBクライアントとサーバーが相互に適切に認証できるようにする基本的な暗号制御です。ただし、ほとんどの場合、デフォルトでは有効になっていません。Windows管理者にとって残念なことに、SMBのMicrosoft以外の実装では、SMB署名が必要でない場合が多く、サポートされていない場合もあります。さらに、SMBメッセージ署名は、Windowsではドメインコントローラーに対してのみデフォルトで有効になっており、他のWindowsクライアントやサーバーではデフォルトで有効になっていません。 SMBメッセージ署名を有効にする手順はMicrosoftによって十分に文書化されていますが[2]、多くのWindows管理者は、特にネットワークの資産の数が非常に動的であるか、IT運用スタッフによく知られていない場合は特に、この基本的な暗号制御の適用についてあまり積極的ではありません。

最後に、SMB接続の問題のトラブルシューティングは、SMB署名を一時的に無効にすることから始まる傾向があり、一時的なトラブルシューティングの対策は、企業の永続的な設定ミスの悪名高い原因です。つまり、ネットワーク上のすべてのホストでSMB署名が厳密に要求されていない場合、ペンテスターは、宛先サーバーのIDの暗号化の保証がないことを利用して、SMBベースの資格情報を盗むことができます。これらのクライアントに、クライアントがすでにネットワーク関係にある可能性のある偽のサーバーと通信させることができます。

次に、2番目に人気のある脆弱性であるLLMNRおよびNetBIOSスプーフィングが発生します。クライアントが内部ネットワーク上の特定のサーバーのキャッシュされたネットワークアドレス情報をまだ持っていない場合、クライアントは「リンクローカルマルチキャスト名前解決」（またはLLMNR）を実行しようとする可能性があり、ローカルネットワーク上のどのマシンが最初に応答してもクライアントが信じるのは宛先サーバーです[3]。もちろん、実際のクライアントと実際のサーバーの両方がSMB署名を必要とする場合、なりすましたSMBサーバーは正しく認証できないため、この偽装はあまり長く続きません。ただし、上記で説明したように、ペンテスターは、内部の契約でSMB署名が不要な状況に遭遇することがよくあります。

したがって、上記の2つの問題を組み合わせると、攻撃者はWindowsのパスワードハッシュを取得し始めることができます。SMB署名を必要としないクライアントは、アドレスが割り当てられていないサーバーを見つける必要があり、そのサーバーには（なんらかの理由で）定義済みのDNSエントリがありません。それは、ネットワークに「新しいキャッシュ、だれですか？」と尋ねます。攻撃者のマシンは「それは私だ！誰だ？」クライアントにWindows認証を要求します。クライアントは、攻撃者のサーバーを信じて、SMBチャレンジ/レスポンスの一部としてハッシュを提供します。ハッシュは後で攻撃者によってクラックされ、クライアントに直接ログインするために再利用されます。

攻撃者がWindowsドメインユーザーのアカウントにアクセスすると、通常、そのドメイン内の任意のワークステーション（具体的には、Kerberos対応のサービスアカウントを実行する必要があるサービスを実行している可能性のあるワークステーション）にログインできます。「Kerberoasting」攻撃を入力します。ローカルマシンの攻撃者は、実行中のプロセスをのぞき見し、ドメインサービスアカウントのパスワードの暗号化された（ただしクラック可能な）バージョンにアクセスする可能性があります[4]ドメインバックアップやウイルス対策を実行するサービスアカウントは、かなり高い権限で実行される傾向があり、パスワードが人間によって設定されている場合、短時間でかなりクラックされる傾向があります。

最後に、4番目の脆弱性であるCleartext Credentials Stored in Memoryに示されているように、一部のサービスは認証にドメインベースのKerberosチケットを利用しません。繰り返しになりますが、攻撃者がドメインに接続されたワークステーションにログインできる場合、それらの資格情報はMimikatzのセッションから離れている状態にすぎません[5]。

---

[2] https://docs.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/microsoft-network-client-digitally-sign-communications-always

[3] NetBIOSは基本的に同じ方法で（ローカルネットワークブロードキャストリクエストを使用して）、機能的に同一の状態で動作するため、両方のテクニックが一緒に詰め込まれる傾向にあります。

[4] これらすべてが機能する仕組みは非常に複雑ですが、Rob "Mubix" Fullerが3部にわたるブログシリーズでわかりやすく説明しています。パート1はこちら：https://room362.com/post/2016/kerberoast-pt1/。

[5] https://github.com/gentilkiwi/mimikatz、Benjamin Delpyによって最初にリリースされた後、ほとんどすべての侵入テストフレームワークで再実装されています。

Windowsの設定ミスだけではない

もちろん、ほとんどの内部ネットワークでは、Microsoftクライアントとサーバーの標準のデフォルト構成だけでなく、いくつかのタイプのオペレーティングシステムとネットワークソフトウェアが実行されています。これらの状況では、ペンテスターは、**重要なセキュリティパッチが欠落している**（この調査では8%の時間利用されている）、または**廃止された、またはサポートされていないソフトウェア**（5.2%のエンゲージメントに含まれる）を実行しているソフトウェアを探し、数千もの既知の事前のソフトウェアを利用できます。つまり利用可能な既存のプルーフオブコンセプトもしくは実用的な悪用方法なのです。

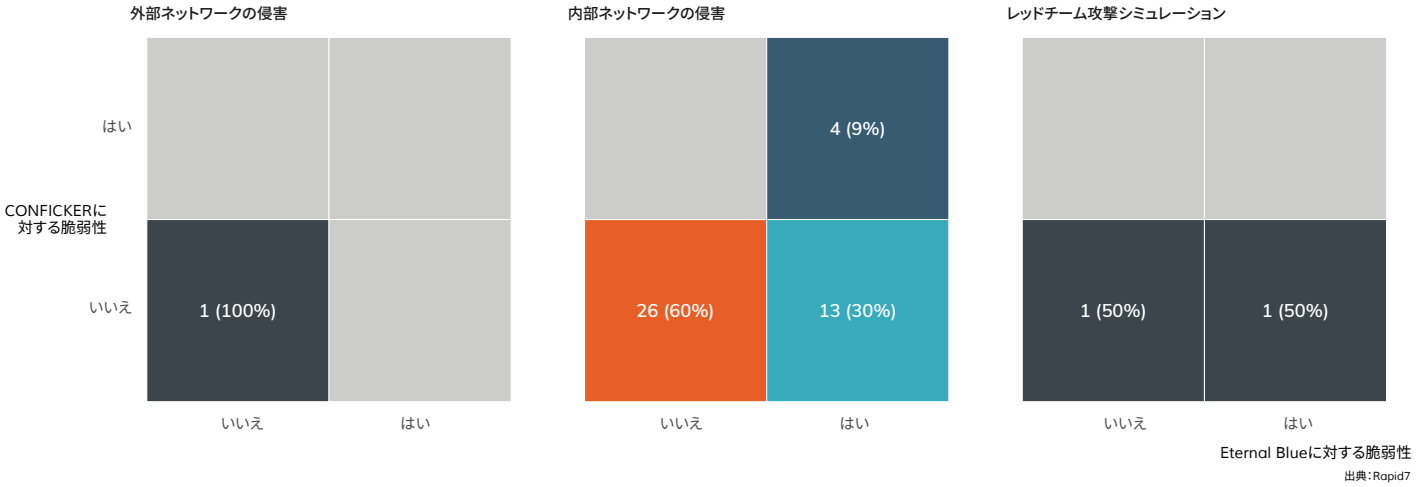
MS08-067およびMS17-10

2つの脆弱性は、内部スコープのネットワーク評価のかなり標準的な頼りどころとして際立っています。2008年にConficker エクスプロイトで兵器化されたMS08-067と、2017年のEternalBlueエクスプロイトキットの中心的な脆弱性であったMS17-10です。これらの2つの問題は、過去10年間における有名な脆弱性の1つであるため、ITチームとITセキュリティチームはずっと前に内部ネットワークからこれらの脆弱性を排除していたと思います。

残念ながら、図5が示すように、実際にはそうではありません。

図5：ETERNALBLUEやCONFICKERに対して脆弱なホスト

回答のあった数のみを反映しています。パーセンテージは評価タイプ内で計算されています。



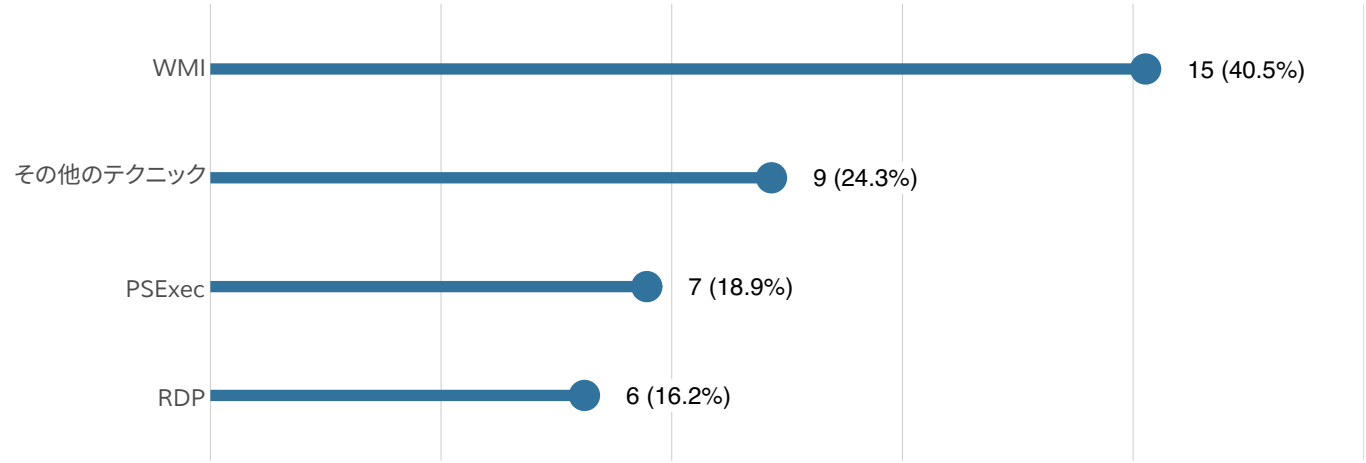
出典：Rapid7

未知の未知

パッチと資産の管理はIT運用の重要な機能であり、パッチが適用されていない旧式のシステムを利用することは、多くの場合、侵入テスト担当者の最初の主要な作業と見なさレテおり、IT運用担当者はパッチ管理プログラムの盲点を知ることができます。これは確かに侵入テストの重要な側面ですが、最も人気のある5番目の脆弱性は、他にはないものです（8%の時点でパッチがない場合でも）。このカテゴリは、5位であり、侵入テストの真の価値を示しています。これまでのところ、セキュリティ制御を完全に自動化する方法（またはそれらの制御をテストする方法）を把握することはできていません。そのため、内部ネットワークの新しい、斬新な、予期しない脆弱性の説明については、精通したペンテスターの創造性と状況認識に依存しています。

図6：内部エンゲージメント：活用した横移動のテクニック

データセットは横移動が可能だった場合のみの数字です。



出典：Rapid7

横移動のテクニック

ペンテスターは、ネットワーク内で少なくとも1台のマシンへのアクセスを取得すると、通常、次の作業として、そのアクセスを活用してネットワーク内を移動します。図6に示すように、このような横方向の動きを実行するにはいくつかの方法があります。

これらの手法の2つであるWMIおよびPSExecは、組み込みのWindowsシステムインターフェイスを使用して、ネットワーク全体の制御を拡張します。これらは通常のネットワークメンテナンス、および2017年と2018年の悪名高いWannaCryとNotPetyaの自己増殖ワームによって活用される手法の双方で一般的に使用されます。これらのワームに関するほとんどの議論は、実装されたEternalBlueエクスプロイトに焦点を当てています[6]。しかし、これらのワームが非常に効果的である本当の理由については触れないでください。これらのワームは、実際の攻撃者やペンテスターと同様に、横方向の動きに同じ高度な手法を使用します。攻撃者は、侵害されたシステムでパスワードを回復して再利用することにより、最終的なターゲットを探すためにマシン間を飛び回ることができます。

コードレビューコーナー

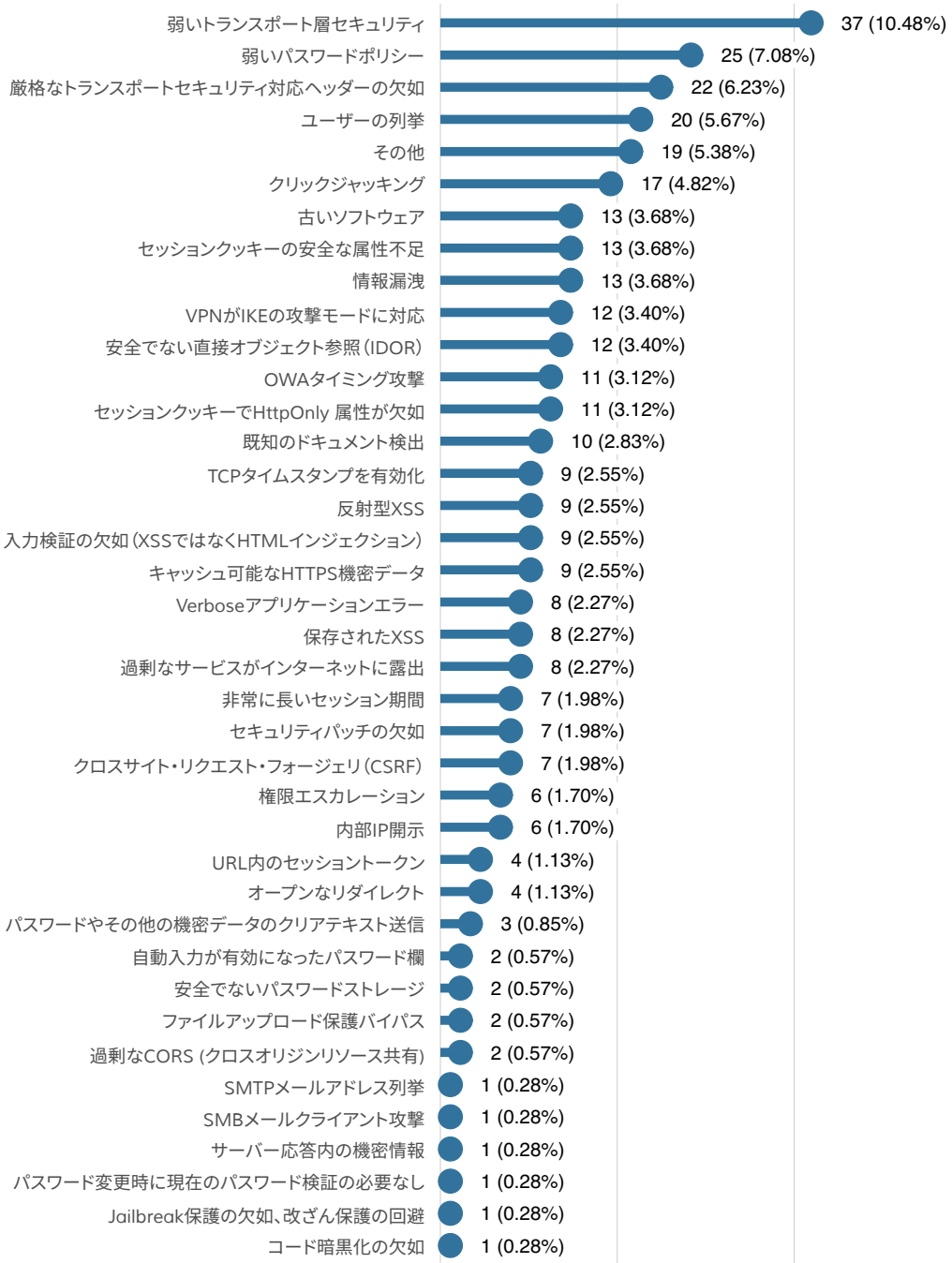
コードレビューの評価はコーパス全体の約3%にすぎないため、コードレビューで明らかにされた脆弱性の種類について統計的に関連する情報はあまりありません。とはいえ、私たちが見つけた脆弱性は、SQLインジェクション、反映されたXSS、安全でないデータ転送とストレージ、CSRF、その他すべての典型的なWebアプリケーションの脆弱性をカバーしていました。つまり、外部Webアプリケーション中心のペンテストとほとんど同じように見えました（詳細については、次のセクションを参照してください）。ただし、例が少ないため、これが普通なものなのかどうかを判断するのは困難です。

[6]本ドキュメントに含まれています。数段落前です！

外部ジョブ

ほとんどの侵入テストは、少なくともこのドキュメントのパート1で説明されているように、約65%程度の外部ベースで開始されます。したがって、これらの調査では、図7に示すように、プラットフォームに依存しないWebベースのテクノロジーに重点を置いています。

図7：外部の関与：どのような脆弱性が見つかりましたか？



出典：Rapid7

これらの脆弱性は、通常「重大度の高い」脆弱性と見なされるものでもありません（Common Vulnerability Scoring Systemでは8.5以上にランク付けされています）。このような脆弱性には、ほとんど常に何らかのリモートコード実行コンポーネントがありますが、外部侵入テスターが遭遇する主な脆弱性は次のとおりです。

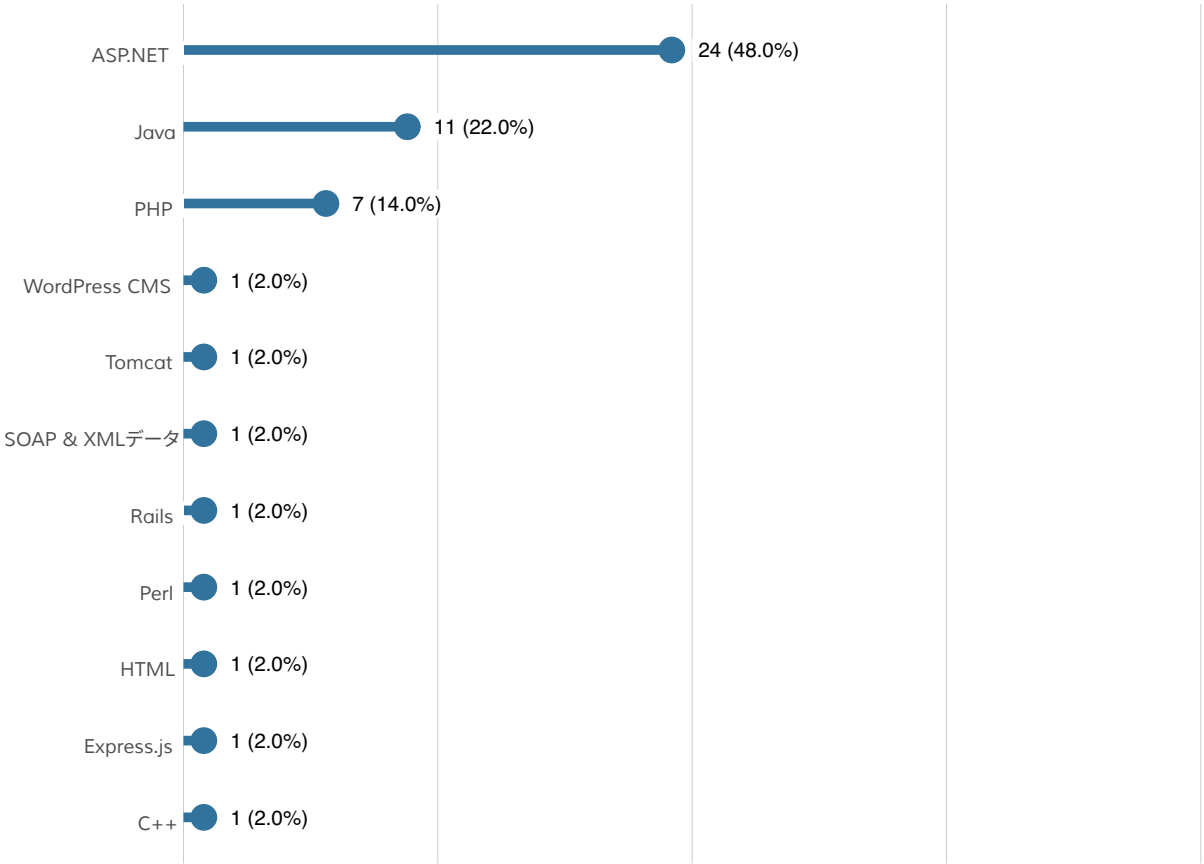
- 弱いトランスポート層セキュリティ（10.48%）
- 脆弱なパスワードポリシー（7.08%）
- Strict-Transport-Security（STS）応答ヘッダーがない（6.23%）
- ユーザー列挙（5.67%）

これらの脆弱性はすべて、情報漏えいをより目指しており、攻撃者の側で他の何らかのアクションが必要です。つまり、少なくともアプリケーションに対する認証後の特権を取得するために、暗号制御が弱いことから通常の通信を傍受するか、クレデンシャルのスタッフィングか、もしくは、パスワードスプレーベースの攻撃をします。**その他**、（5.38パーセント）**古いソフトウェア**（3.68パーセント）、そしてさらに深刻な場合に、straight-shot-to-command-executionの脆弱性と悪用が見られる**安全でない直接オブジェクト参照**（IDOR）　（3.4%）に取り掛かるまでそうはなりません。

Webアプリケーション技術

外部のペンテスターがWebアプリケーションの問題を発見したら、そのWebアプリケーションが何で書かれているか、通用する実用的な知識があれば、多くの場合役に立ちます。

図8：どのテクノロジーに基づいてWEBアプリケーションが構築されていますか？



出典：Rapid7

外部アプリケーションと内部アプリケーションの両方で、**ASP.NET** （48%）がリストの上位にあり、**Java** （22%）と**PHP** （14%）がそれに続いています。StackOverflowの最近の「Most Loved Languages」調査ではこれらのWebアプリテクノロジーがうまく機能しないため、これはWebアプリの侵入テスト分野のキャリアを検討しているWebアプリデベロッパーにとっては悲しいニュースになるかもしれません。[7]

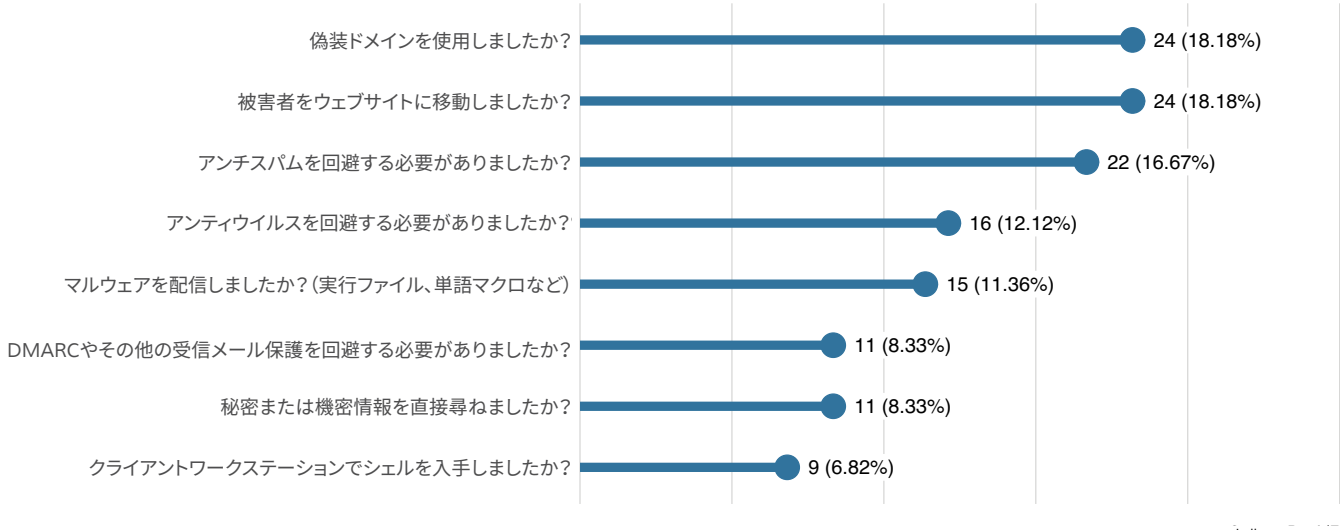
実際、Rust、Kotlin、TypeScriptなどの最もホットなWebアプリフレームワークがエンタープライズの侵入テストで登場することは（まだ）めったにありません。これは、少なくともWebアプリケーション開発に関しては、企業のITが最新のテクノロジーに追いつくために必要であり、これらのテクノロジーを導入することで利益を得ることができることを示しています。より最新のものでありタイプセーフであり、多くの一般的なWebアプリの脆弱性を防ぐための組み込みのバックストップがあります。

電子的なソーシャルエンジニアリング

多くの高度な外部エンゲージメントには、電子的なソーシャルエンジニアリング（ESE）が関係しています。これらは「レイヤー8」の脆弱性（人間の相互作用や、信頼、恐れ、貪欲などの傾向を含むもの）を悪用する傾向がありますが、図9に示すように、このような攻撃には依然として重要な技術的な要素があります。

図9：メールベースの電子ソーシャルエンジニアリング：何が起こったのか？

合計試行回数に基づいて計算されたパーセンテージ。



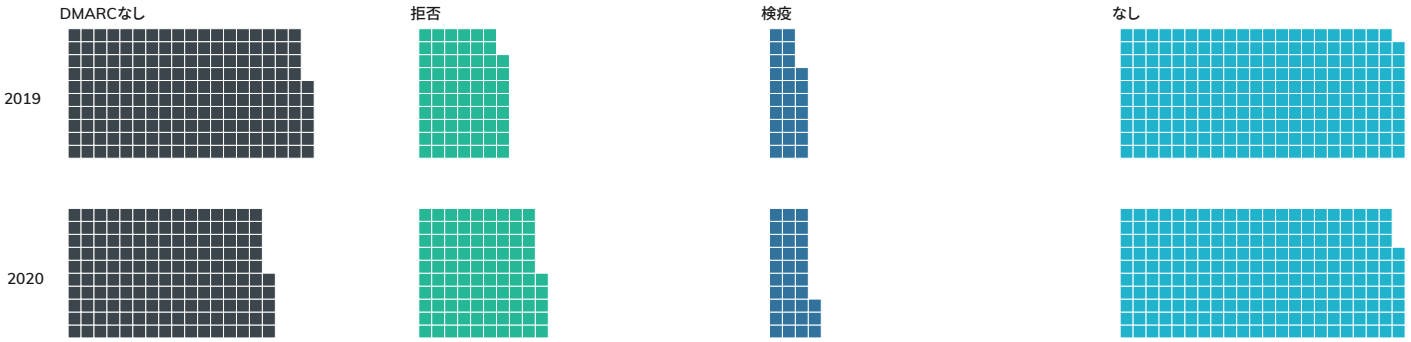
出典：Rapid7

偽装されたドメインは、 ESE調査回答の18%にのぼり、回答者はまれにしかDMARC回避（8.33パーセント）を行う必要がありませんでした。これは、従業員をこれらの攻撃に陥ることから保護する上でスプーフィング対策がいかに重要であるかを示しています。DMARC、つまりドメインベースのメッセージ認証、レポート、および適合性により、フィッシングメールが標的のドメイン内から送信されたと主張することが難しくなります。つまり、攻撃者は、他の見つけやすいテクニックを使用して、より説得力のあるフィッシングルアーを作成する必要があります。確かに、Rapid7によるFortune 500の最近の再評価とDMARCステータスの数値は、積極的な動きがいくつかある[8]ことを示しています。

[7] https://insights.stackoverflow.com/survey/2020#technology-most-loved-dreaded-and-wanted-languages-loved （ASP.NETはStackOverflowで特に調査されていませんが、ほとんどのASP.NETアプリケーションはC#で書かれています。）

[8] https://blog.rapid7.com/2019/10/18/what-a-difference-a-year-makes-revisiting-our-inaugural-fortune-500-icer-one-year-later/

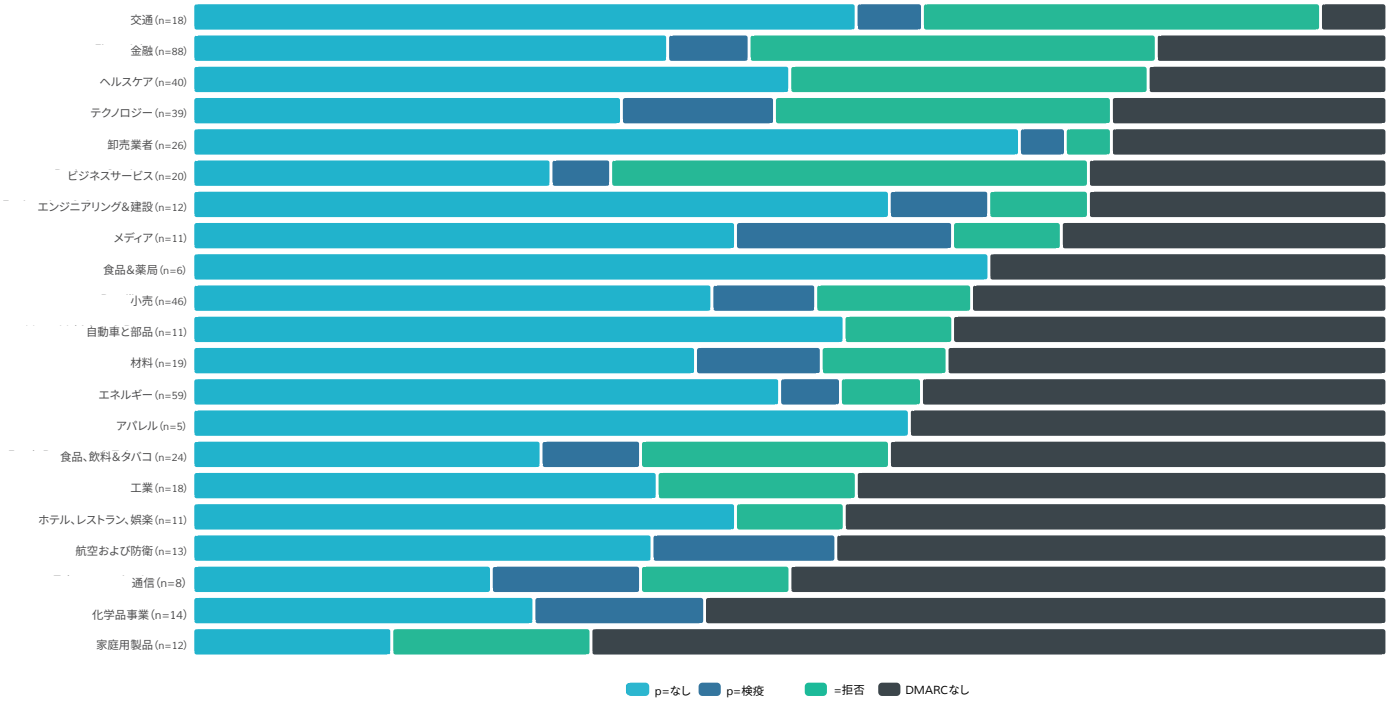
図10：FORTUNE 500のDMARCステータス：2019と2020



出典：Rapid7

2019年から2020年までに、Fortune 500のサイトでは、「DMARCなし」から少なくとも「なし」、「検疫」、「拒否」のいずれかに設定されたDMARCポリシーに20%増加しました。同時に「拒否」DMARCポリシーは40%増加しました。DMARCの導入が順調に進んでいるのは素晴らしいことですが、これらのメリットはすべての業界に均等に行きわたっておらず、一部の業界ではまだ「なし」のポリシーから始めているようです。

図11：FORTUNE 500のDMARCステータス



更新日：2020年8月

また、ESE攻撃の最も一般的な「ペイロード」は、攻撃者が制御するWebサイトにユーザーを誘導することです（時間の18.18%）。これは、マルウェアを配信する（11.36%）よりもはるかに一般的です。ブラウザーの製造元がブラウザー制御の暗号化DNS（企業のITセキュリティ機関による検査を免れる）[9]を要求していることもあり、ウェブサイトを仲介者として使用してパスワードを収集し、ブラウザーセッションを攻撃することと、（企業が管理するVPNの外で電子メールを読んだりWebを閲覧したりする可能性がはるかに高い）パンデミックによって生じた世界中のナレッジワーカーの点在化により、攻撃がより成功することが予測されます。

[9] https://spectrum.ieee.org/tech-talk/telecom/security/the-fight-over-encrypted-dns-boils-over

レッドチームのシミュレーション

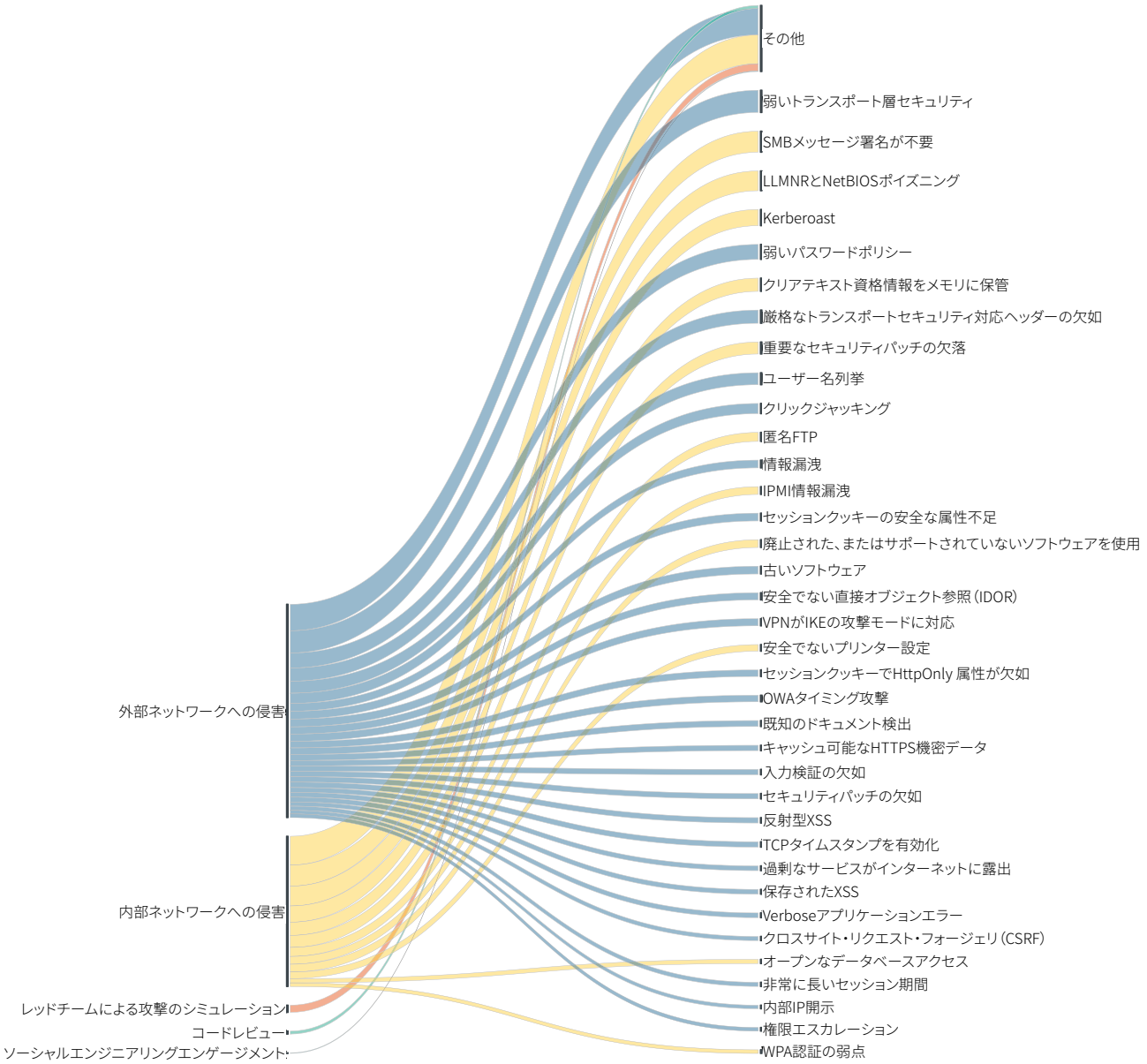
内部ジョブのコードレビューのカテゴリと同様に、レッドチームのシミュレーションは収集されたデータの数がかなり少ないため、エンゲージメントレポートから統計的に有意な結果を見つけることが困難です。とはいえ、これらのシミュレーションで明らかになったNo.1の脆弱性の種類は「その他」であり、これらのエンゲージメントは、効果的に完了させるために、かなりの量の状況認識とその場で創造力を発揮させる傾向があるという事実を物語っています。これは、レッドチームのエンゲージメントにサインアップする顧客の種類が、セキュリティの成熟度のスペクトルのかなり高い位置にあり、ビジネスがかなり厳しく施錠されているためと考えられます。

人気のある脆弱性：何を守るべきか

評価の種類（社内または社外）に関係なく、ペネトレーションテスターによって報告された脆弱性のリスト全体の中で、どの脆弱性が上位にあるかを確認することは非常に重要です。

図12：エンゲージメントタイプ別の脆弱性

「その他」には、発生が5回未満の脆弱性の種類が含まれます。縦の帯は出現回数に応じています。

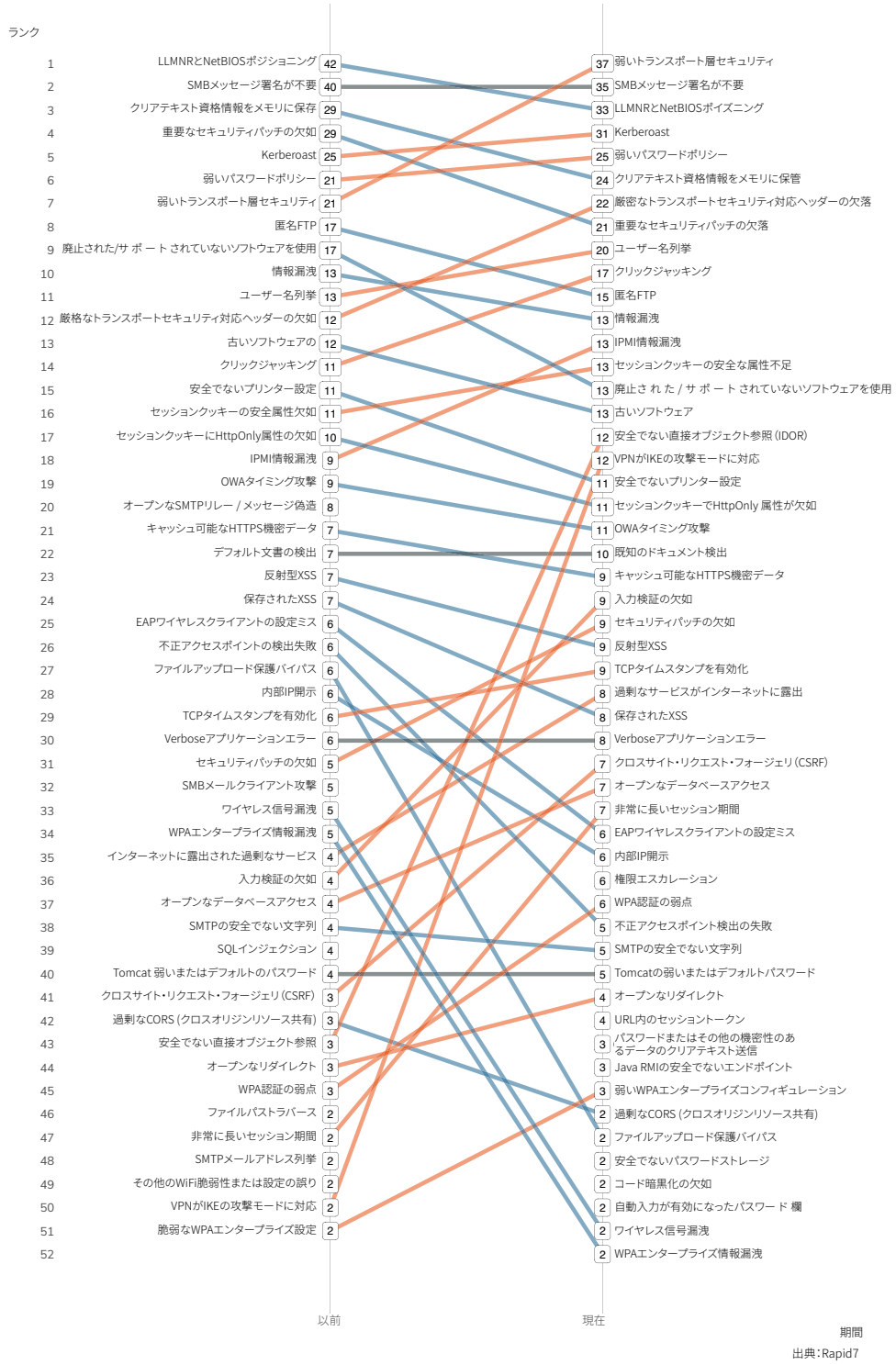


ここでは、その他が最も一般的な脆弱性の種類であることがわかります。これは、侵入テスターがネットワークの侵害の動きを網羅できないという主張をさらに強調しています。各ネットワークは独自の方法で一意であり、成功した侵入テストは、何かがうまくいかなくなったときに気づく熟練した独自の侵入テスターの専門知識に依存しています。

これらの未知の未知のものとは別に、IT運用チームは、侵入テスト担当者が現場に到着する前に、悪用された最も一般的な脆弱性について、自社のネットワークを調査するのが得策です。これは、定期的な侵入テストの規制要件がある業界で特に当てはまります。これらの基本事項に注意を払い、予測可能なエンゲージメントを準備することで、侵入テスト担当者は、ネットワークのあまり調査されていない領域に集中できるようになるだけでなく、犯罪者が実行する最も一般的な種類の攻撃に対して企業環境を強化できます。

図13：以前の脆弱性と現在の脆弱性

期間内の相対頻度に基づく順序。数値ラベルは、脆弱性インスタンスの数を反映しています。



出典：Rapid7

前年比の変化

本レポートは、Rapid7が脆弱性に関する同一のアンケートを使用して作成した2回目のUnder the Hoodie Reportです。結果として、ペネトレーションテスターに対して、個々の脆弱性が年ごとにどのように進展しているかを確認する機会が得られました。

2つの最も急激な増加には、**VPNがIKEアグレッシブモードをサポートしていることと、Insecure Direct Object Reference（IDOR）が関係しています。**ベースの調査は、ほぼ確実に、パンデミック中のVPN依存の新たな現実を反映して、テレワーカーの突然の膨大な人口増加で、オンラインでの生産性が向上する一方で、IDORの露出は、同様にWebアプリケーションへの関心が高まっていることを反映しています。それらがどのように悪用されて外部の位置から任意のコードを実行できるかを図にしました。

ペンテストの実際：

XMLで うまくいく

By: Tommy Dew

ごく最近、私は通常のWebアプリケーション侵入テストのようなものに取り組んでいました。通常、クライアントは、管理者アカウントやいくつかの異なるユーザーロールなど、いくつかのユーザーアカウントを提供します。また、関連するAPIドキュメントと例、または設計情報も提供します。ただし、アプリケーションのドキュメントで目立ったのは、仕様上、アプリケーションがXMLファイルのアップロードを特別に許可していることでした。通常のWebアプリケーションのテスト方法論を試してみて、アプリケーションの感触を得たら、XML外部エンティティの処理を可能にする特別に細工されたXMLファイルの作成を開始しました。

XML外部エンティティ処理攻撃の仕組みに慣れていない場合は、次のようになります。ペネトレーションテスターまたは悪意のあるアクターは、弱く構成されたXMLパーサーによる悪意のあるXMLデータの処理を可能にするアプリケーション機能をターゲットにします。悪意のあるXMLがXMLパーサーによって正常に処理された場合、テスターはペイロードに外部エンティティを導入し、Webアプリケーションサーバーのシステムファイル（/ etc / passwdなど）を取得したり、サーバー側要求偽造（SSRF）を実行したりする可能性があります。 ）攻撃。Webアプリケーションサーバーに、サーバー側にネットワーク下りフィルターがない場合、任意のURLにHTTPリクエストを送信するように指示します。

WebアプリケーションがXMLを受け入れることを知っていたので、Rapid7が制御するWebサーバーに要求を出す悪意のあるXMLを盲目的に受け入れるかどうかを確認したかったのです。最初に、Webアプリケーションサーバーに GETリクエストを送信するように指示するだけのXMLファイルを作成しました。ジャンプボックスで、単純な一時Webサーバーを起動し、悪意のあるXMLをWebアプリケーションに送信しました。すぐに、IPアドレスから「GET」リクエストを受け取りました。簡単に確認したところ、Webアプリケーションサーバーの IPアドレスであることを確認しました。

Webアプリケーションが悪意のあるXMLファイルを受け入れたことを知った後、私は追加のリスクを示し、前述のSSRF攻撃などの追加のアクションを実行できるかどうかを確認したいと思いました。一部のチームメンバーとの簡単な議論と推奨事項の後で、再び新しいXMLファイルを作成しましたが、今回はXMLファイルを定義して、存在しないファイルをジャンプボックスで取得するようにWebアプリケーションサーバーに指示しました。 Webアプリケーションサーバーは Windowsだったので、攻撃が成功した場合は、Windows NTLMハッシュも送信されることがわかっていました。次に、Windowsハッシュのキャプチャを可能にするプログラムであるResponderをサーバーで起動し、ペイロードをアプリケーションにアップロードしました。これにより、Webサーバーのハッシュされた Windows資格情報が正常に送信されました。

しかしながら、資格情報はWindowsマシンアカウント用でした。これは、Webアプリケーションの侵入テストの妥当な時間中に解読することは非常に難しいことで有名です。しかし、私のクライアントは積極的に対応し、数日で脆弱性を解決するために私たちと協力しました。

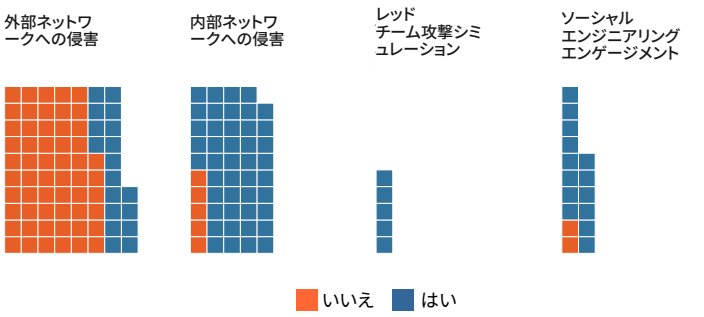
すべてのアプリケーション評価でこの種の脆弱性に遭遇することはありませんが、特にアプリケーションがエンドポイントを介したコンテンツのアップロードまたはXMLの送信を許可している場合、これは常にテストされるものであることに疑いはありません。XMLとSSRFを組み合わせることによるリスクのデモンストレーションは、重大度のレベルをクライアントに変換し、リソースをアプリケーション専用にして問題をすぐに解決できると信じています。これは侵入テストの最もやりがいのある側面の1つであり、可能な場合は、作業がプラスの影響を及ぼし、セキュリティ体制を改善しているのを見ることができます。

クレデンシャルの収集

侵入テストの中心は脆弱性や設定ミスの悪用ですが、企業で使用されている資格情報をいかに適切に保護するかが重要です。結局のところ、慎重に作成されたバッファオーバーフローでシェルをポップするのはとても楽しいのですが、許可されたユーザーになりすまして、それ以外の場合は許可されていないアクションを実行する方が簡単な場合がよくあります。さらに、多くの直接的な技術的攻撃の目的は、キーパスワードを回復し、ドメイン管理者の役割を引き受けることです。図14は、特定のシナリオで資格情報が侵害される頻度を示しています。

図14：資格情報を取得しましたか？

エンゲージメントの範囲内で資格の取得があった場合のサブセットデータ。



出典：Rapid7

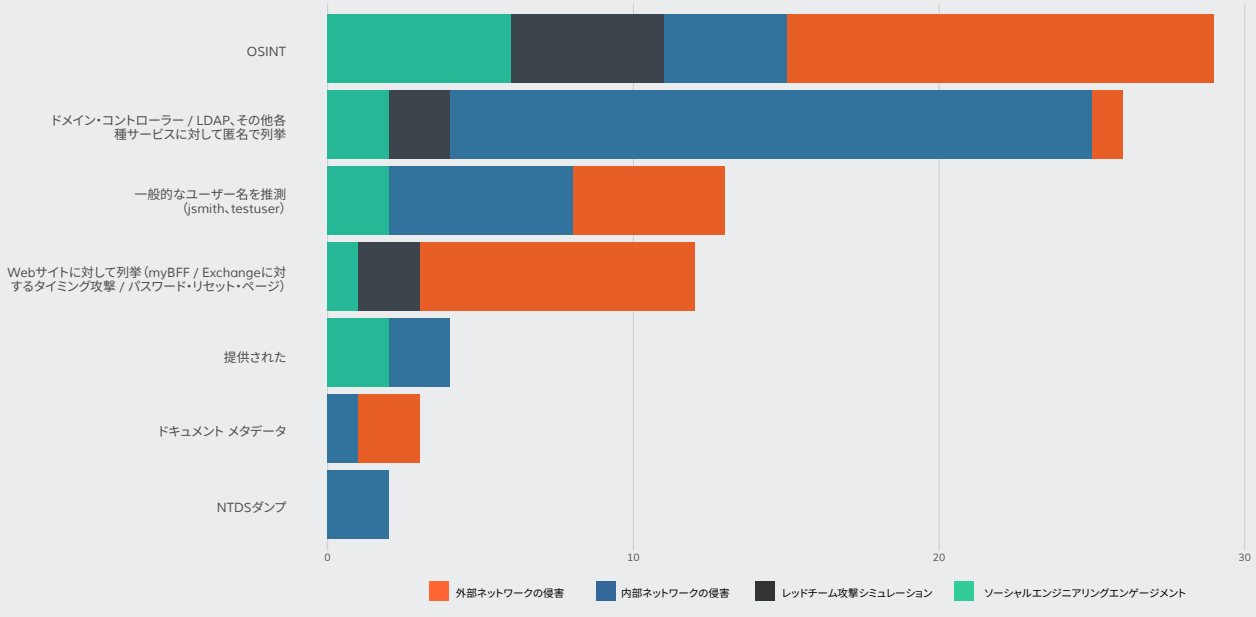
ここでは、内部の仕事と外部の仕事の明確な違いを見ることができます。外部侵入テストは、多くの場合、資格情報の盗難に関係していませんが、内部ネットワークの侵害は通常、すべて資格情報に関係しています。また、レッドチームの攻撃シミュレーションやソーシャルエンジニアリングのエンゲージメントを含むより高度なエンゲージメントは、ほとんどいつもパスワードの盗用に成功しています。

ユーザー名について

通常、資格情報を収集する最初のステップは、有効なユーザー名のリストを取得することです。図15では、攻撃者がこの資格情報の半分を回復するために使用するさまざまな方法を確認できます。

図15：資格情報の取得：ユーザー名はどのように収集されましたか？

テクニックは、複数回に渡って反映されました。



出典：Rapid7

現在、ユーザー名は名目上は秘密ではなく、技術的には「公開」されている場合があります。結局のところ、あなたのユーザー名をだれも知らなければ、彼らはあなたにメールしたり、ドキュメントのアクセス許可を割り当てたりすることはできません。つまり、**OSINT**（またはオープンソースインテリジェンス）は、従業員が望む望まないに関わらず、名前、ユーザー名、勤務先を共有することができる、LinkedInやFacebookなどのソーシャルネットワーキングサイトを含む、特定の企業で使用されている可能性のあるユーザー名を収集するために使用されることが多いことがわかります。内部の関与について、攻撃者は通常、ドメインコントローラーやLDAPサービスなど、内部で公開された**匿名リスト**のソースを使用していることがわかります。

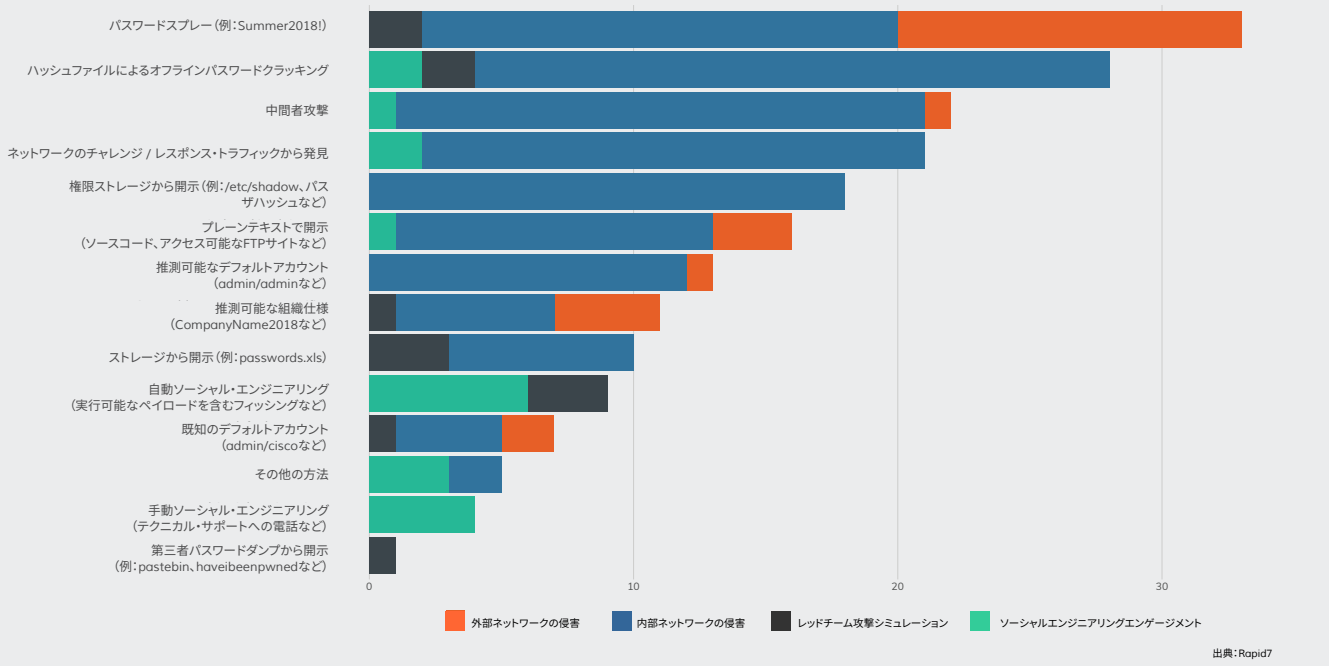
ユーザー名を内部の機密情報として扱うべきだと断言できる段階に達しているとはまだ思っていませんが、企業は、攻撃者が有効なユーザー名のリストを簡単に作成できるようにすべきではないと言えます。実際、外部からユーザー名を収集する2番目に最も一般的なのは、**Webサイトの列挙**を使用する方法です。これは、Webアプリケーションが有効なユーザー名とそうでないものに関する情報を提供する場合です。パブリックOSINTの制御は最も秘密の組織以外では困難ですが、過度に許容的なドメインコントローラーやWebアプリケーションを通じて匿名の列挙ソースを検知して無効にすることで、攻撃者の作業をさらに困難にすることができます。

パスワードの盗用

もちろん、パスワードは秘密になっているはずです。組織内の全員が自分のためだけに持っておくことになっている秘密であり、組織を機能させるたに共有されている内部秘密にアクセスできるようにするためのものです。ただし、これらのパスワードを取得する方法は無数にあります。とりわけ、パスワードが人間によって生成され、その発想の独自性があまりに貧弱な場合に容易になります。

そして、内部または外部のエンゲージメントでパスワードを「推測」する最も一般的な方法は、きちんと正確に推測することです。一番の方法は、**パスワードスプレー**です。攻撃者は有効なユーザー名のリストを既に知っています（または良い考えを持っています）が、非常に特殊な、オリジナルではないいくつかのパスワードしか試行しません。このテクニックは、昔のハックスターの「コールドリーディング」のテクニックによく似ており、「超能力者」がテーマについて非常に関連性の高い話をしているかのようです。しかし実際のところ、「超能力者」は、ごく一般的な人間の心理状態に関する知識に依存しています。ほとんどすべての人が、名前のイニシャルの「M」、亡くなった親類の名前のイニシャルの「J」や、完了していない仕事への取り組みに対する熱意を失っています。

図16：資格情報の取得：パスワードまたはパスワードハッシュはどのように取得しましたか？



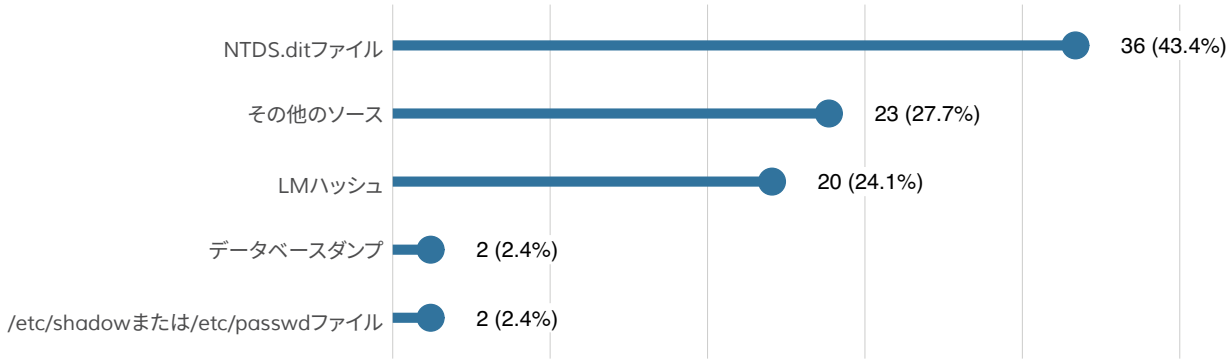
出典：Rapid7

そのため、企業のポリシーに適合するパスワードスキームを常に編み出したり、再度、編み出したりしています。少なくとも8文字で、いくつかの数字と少しの句読点があり、タイプしやすく、覚えやすく、簡単に変更できる必要があります。 90日のスケジュールでで更新しやすいものです。「Summer2020！」よりもこのルールに相応しいものなどありません。[10]

ハッシュのハッキング

適切なパスワードを取得するための2番目に一般的な方法は、最初に一連のハッシュまたは暗号化されたパスワードを収集し、次にそれらにプロセッサー能力とともに巨大な辞書を投入することです。実際、図17に示すように、ハッシュのソースは多少異なります。

図17：収集したハッシュの種類

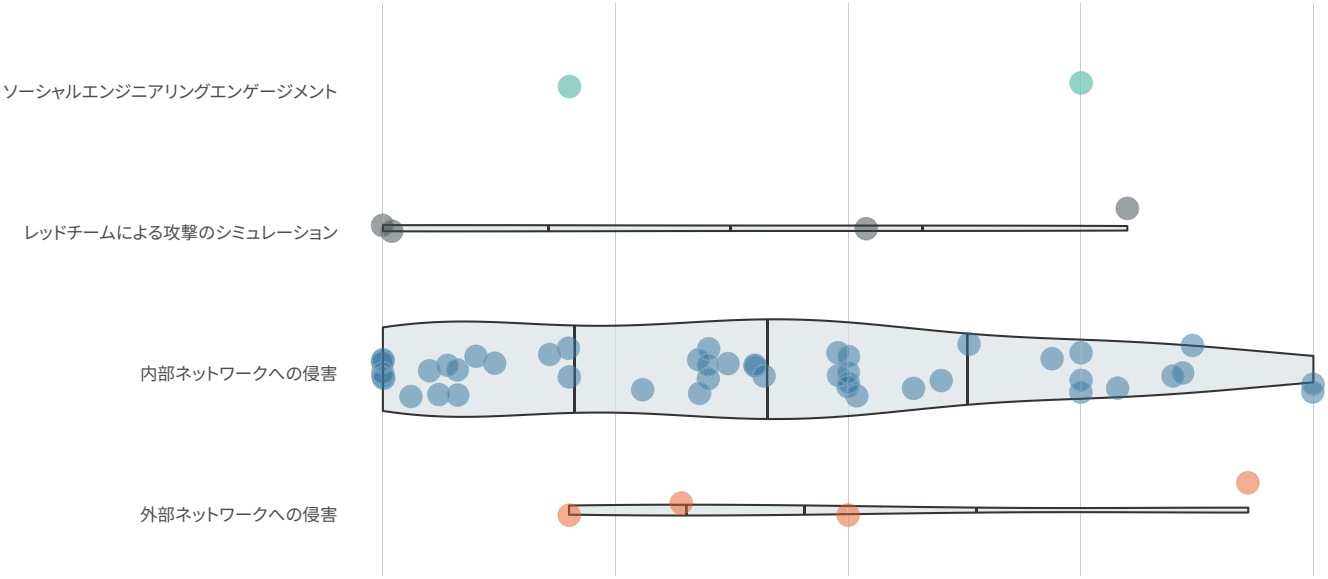


出典：Rapid7

パスワードハッシュは非常に有用であるため、収集は通常、内部ネットワーク評価の1次的な目標ですが、パスワードは正確にハッシュであるため、攻撃者が単純に読み取って使用することはできません。ただし、ハッシュファイルが悪用された場合、簡単に解読できるパスワードでいっぱいになる傾向があるのは残念なことです。図18は、ハッシュファイルが取得されると、同意されたエンゲージメント時間内に、一部のパスワードがクラックされることを示しています。

図18：解読したパスワードの割合について

外れ値はデータ分布に対応。25%、50%（中央値）、および75%の変位値で改行しています。

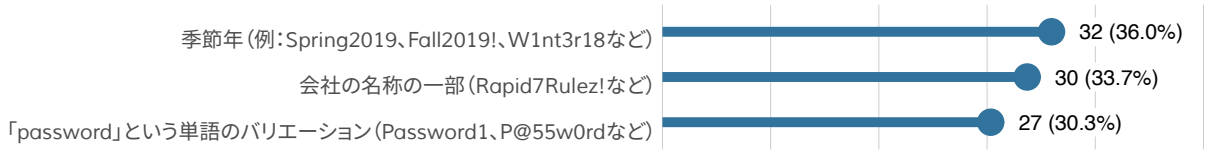


[10]これがパスワードなら、今すぐ変更してください。今すぐにです。「Autumn2020!」にはしないでください。

与えられたハッシュファイルでは本当に強力なパスワードがめったに見られないだけでなく、半分の確率で、非常に推測可能で予測可能なパスワードを選択することが確認できます。

図19：推測可能なクラックされたパスワード

パスワードクラッキングが対象範囲内にあるエンゲージメントに限定。



出典：Rapid7

Nullセッションに対する注意

推測またはクラックされたパスワードを確認するための一般的な方法の1つは、Windows nullセッションを利用することです。以前は、WindowsベースのLANにおいてnullセッションが非常に一般的でしたが、近年、Windows管理者は、攻撃者にこの便利なサービス以外にサービスを提供していないことに気づきました。今日、調査されたペネトレーションテスターの40%だけが、ドメインコントローラーでnullセッションが有効になっていることを発見しました。これは、2019年のレポートの半分から減少しています。

現在、匿名のnullセッションは無効になっているか、その状態が新しいWindowsマシンで定義されていません。[11]デフォルトでは全体的に無効になっていないので、Windowsドメイン管理者は、それらを無効にするために自分で設定していく必要があります。

図20：ドメインコントローラーでNULLセッションが有効になっていますか？

出典：Rapid7

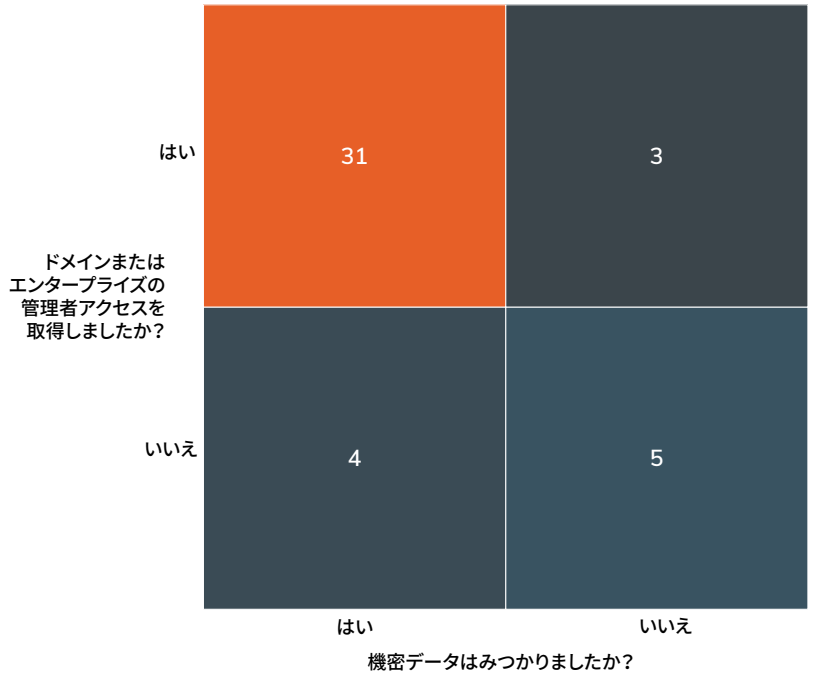
[11] https://docs.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/network-access-restrict-anonymous-access-to-named-pipes-and-shares

特権アカウントと機密データ

通常のユーザーアカウントは多くの場合、特定の組織への道のりですが、最も機密性の高いデータは、推測可能なパスワードを持つその1人のユーザーの手の届かない範囲にあることがよくあります。機密情報を取得する最も簡単な方法は、ドメインまたはエンタープライズ管理者のレベルにエスカレートすることです。31の内部ネットワークの侵害では、図21に示すように、一方が他方につながります。

図21：内部ネットワークの侵害：機密データへの抵抗が最も少ない経路

特権アカウントは、認証ボールトのロックを解除するためのキーを提供します。セルの値はエンゲージメントの回数を示しています。回答があった場合のみ。



出典：Rapid7

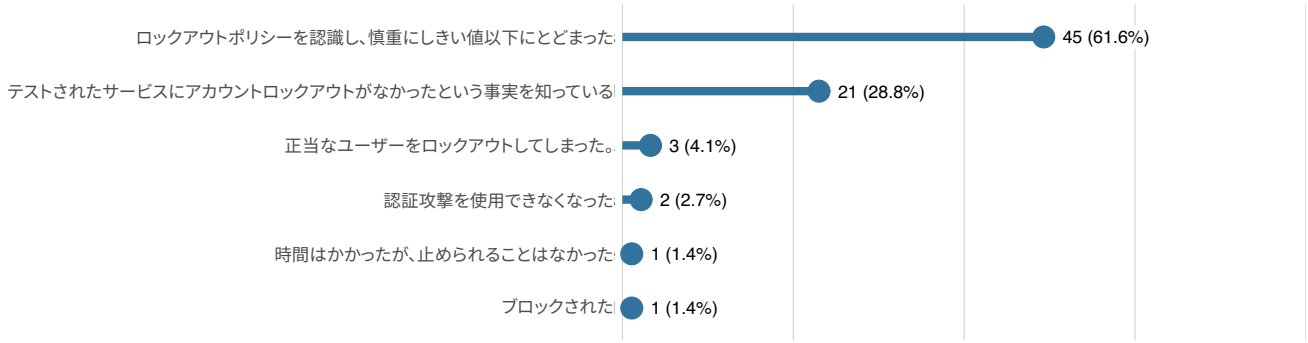
ロックアウトポリシーと2FA

ロックアウトポリシーと2要素認証（2FA）は、パスワードを保護し、攻撃者が発見したユーティリティを制限するための重要なセキュリティコントロールです。残念ながら、それらの展開はまばらであるか、さもなければ今日のエンタープライズ環境において効果がありません。図22は特に厄介な問題です。

ここでは、ごく少数の外部エンゲージメントのみでアカウントロックアウトポリシーによって大きく妨害されていることがわかります。これは、ロックアウトポリシーが対象のサービスに効果がないと言っている訳ではありません。むしろ、ペネトレーションテスターが少数のパスワードをすべてのユーザーにスプレーしているため（つまり、テストされた各ユーザーのしきい値を下回っていないため）、ロックアウトポリシーが無効であるか、メイン認証システムで採用されているポリシーがロックアウトを強制しないサービス（通常は電子メール）です。

図22：外部エンゲージメント：ロックアウトはどの程度効果的でしたか？

数字は、利用可能な回答のみを反映しています。



出典：Rapid7

とはいえ、図23に示すように、調査したペンテスターは、昨年よりも昨年より2FAに頻繁に遭遇しました。

図23：テストしたサービスで2要素認証が有効になりましたか？

出典：Rapid7

2019年の調査では、2FAは21%しか見られませんでしたが、今年は35%と大幅に増えています。繰り返しますが、効果を上げるためには、2FAがすべての出口ポイントをカバーする必要があります。つまり、すべてのセカンダリ認証システムが2FAを必要とするか、異なる一意のパスワードを使用する必要があります。

資格情報の保護

パスワードスプレーと、ハッシュファイルのコンテンツに対して一般的に使用されているクラッキングの双方に対する最善の防御策は、少し驚くべきことに同一なのです。人間に頼ってパスワードを選択させないでください。組織のセキュリティを最適に強化するためにIT組織が実行できる1つのプログラムは、**マシン制御のパスワード管理**を標準化することです。選択できるソリューションは多数ありますが、最終的には、強力で長く一意のパスワードを生成して保存し、すべてを取得するために何らかのメカニズムが必要です。そして、社内でも、職場でも、組織内の*全員*がこれを採用する必要があります。侵入テストでシミュレートされた、または実際の犯罪者が実際に行ったほぼすべての攻撃的なオペレーションには、プロセスのある段階で、推測、クラッキング、またはその他の方法で重要なパスワードを取得することなどがあります。

ペンテストの実際：

Vexing VPNの裏をかく

By: Robert Stewart

大部分のエンゲージメントと同様、ターゲットとしている従業員に関する情報を探すためにインターネットを探索することから始めました。私が理解するのに苦労したことの1つは、ユーザー名の形式が何であるかでした。ホストしているドキュメントからメタデータを削り取り、hunter.ioなどのリソースを使用し、またHarvesterを使用して自分ができることを掘り下げました。しかし、私はまだフォーマットに前向きではありませんでした。ほとんどの企業は、FLast、First.Lastなどの一般的な形式を使用しています。ユーザー名の形式を検証するために、オンプレミスのSkypeサーバーで使用されるLyncサービスのタイミングの脆弱性を利用するlyncsmashと呼ばれるツールを使用していました。FLastとFirst.Lastをテストしても何も見つかりませんでしたが、3回目の試行で、LastFを使用するつもりでした。かなりの量の有効なユーザー名がスクロール表示さるのを見始めました。

有効なアカウントのユーザー名のリストを解析したところ、482のうち、230の有効なアカウントが見つかりました。これはまだ良い数字だったので、lyncsmashを使用してパスワードスプレーを使い続けました。4回目の試行の後、私はいくつかの手がかりをヒットしました。11個のアカウントが戻ってきましたが、すべて同じ弱く推測可能なパスワードを使用しています。

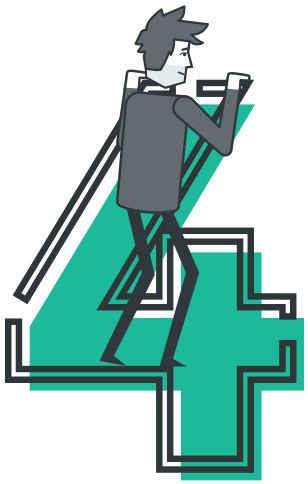
外部アクセスを検証するために11個のアカウントを持つことに懸念はありませんか？

OSINTの実行中に、ターゲットが電子メール、VPN、およびその他のリモートアクセスに使用するいくつかの外部サービスを見つけました。しかし、その後は低迷しました。彼らはすべての外部ポータルで多要素認証（MFA）を使用していました。そして、私はそれについて考え始め、彼らのVPNがいくつかの異なる認証プロファイルを持っていることに気づきました。そこで、私はそれらすべてをテストしました。残念ながら、運がありませんでした。

OSINTの実行に戻り、これらのアカウントでログインできるものを見つけたいと思いました。そして、暗闇の中で、私はついに見つけました。2番目のVPNエンドポイントです。また、そこには他の認証プロファイルにはない追加の認証プロファイルがあり、スマートフォンがネットワークにアクセスするために使用されていました。そして、たまたまそのプロファイルはMFAを必要としませんでした！

スコア獲得！

私は信頼できるSSL VPN Linuxツールopenconnectを破壊して、内部ネットワークに飛び込んだのです。

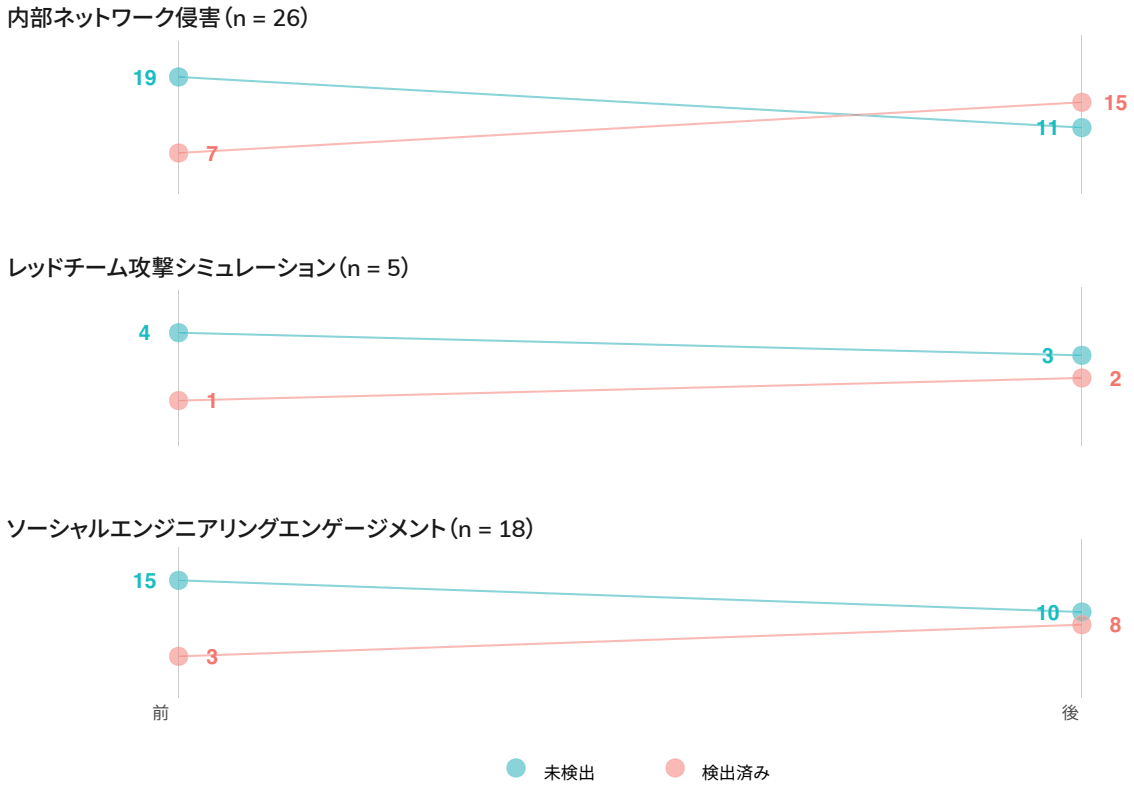


検知と対応および防御

予防技術が何らかの形で最終的には失敗することが認識されているので、検知と対応（D&R）は、成熟したセキュリティ体制にとって重要なコンポーネントです。とはいえ、D&Rは特定の侵入テストのエンゲージメントの対象外である場合があります。これは常にレッドチームのシミュレーションの範囲内ですが、これがなければほとんどのシミュレーションは成り立ちません。しかし、通常、基本的な内部ネットワークに対するコンプロマイズに対する実際的な防御策にはなりません。D&Rは内部ネットワークの侵害の際に警告することができます（すべきです）が、通常、侵入テスト担当者は「つかまえられた」後、テストを再開することができます。ただし、エンゲージメントタイプ全体での検知の割合は興味深いものです。

図24：内部アクセス前または後に、ブルーチーム、SOCあるいは物理セキュリティによって検出、捕獲またはブロックされましたか？

侵入が対象となっており、回答を得られたエンゲージメントのみが含まれます。



出典：Rapid7

図24は、範囲内の資産に対して内部アクセスが成功する前（左側）と後（右側）にペンテスターが気づいた頻度を示しています。アクセス権が取得される前にD&Rが起動した場合（左側の赤）は、D&Rがプライマリコントロールとしての役割を果たしていると確信できます。アクセス権を取得した後にアラートが出た場合（右側の赤）は、2番目のコントロールとしての役割を果たしています。レッドチームのサンプルサイズは小さいですが、通常、エンゲージメント全体で検知がされていないことがわかります。一方、標準の内部ネットワークの侵害は、侵害後にいくらか頻繁に検知されます。つまり、組織は少なくともある程度の確率で気づくことが可能となり実行されたセキュリティ侵害のタイプに対してインシデント対応が可能になります。

基本的な予防

自動化されたD&Rは成熟したITセキュリティプログラムの重要なコンポーネントですが、すべての組織が次回の侵入テストの前にレビューしておくことをお勧めします。おおまかに言って、基本となるのは次の内容です。

- **パスワード管理戦略の確認：**侵入テストの多くで、他にどのような優れたセキュリティ制御が行われているかに関係なく、パスワードのリストや人間が生成したパスワードに行き着きます。資格情報の管理が、組織のITセキュリティプログラムにとってフルタイムの機能になっている必要があります。自動生成され、自動的にローテーションされるパスワードに移行できるユーザーとサービスが多いほど、優れています。最初はマシン制御のパスワード管理を採用するのが面倒ですが、しばらくすると問題は二の次になります。ユーザーに対してパスワード管理ソリューションのトレーニングとサポートを行うことで、ソーシャルエンジニアリングを通じて誤ってパスワードが漏えいすることがほとんどなくなります。また、盗み見られたNTDS.ditハッシュファイルは役に立たなくなります。
- **パッチ管理戦略の確認：**内部ネットワーク評価の最適な使用方法は、パッチ管理戦略がどこで失敗しているかを把握することです。つまり、ITインフラストラクチャの暗い、クモの巣の張った中では、パッチやアップデートの定期的なレビュ ーが行われていないということです。企業が正しく仕事を行うためにユーザーに依存していて、更新のためにそれらのナグ画面をクリックする（「後で」何度も何度もクリックするのではなく）場合、ペネトレーションテスターはほとんど間違いなく、非常に多くの悪用可能な脆弱性に圧倒されます。それぞれのメンバーが最新情報を入手できなくなることが最も悪質な例です。
- **可能な限りのネットワークセグメンテーションの採用：**小規模で管理可能なネットワークセグメントは、内部違反を封じ込めるという点で、全体に対して良い影響を与えます。ペネトレーションテスターは、多くの時間とエネルギーを、最初のシェルを取得するためにではなく、次にどこへ行くかを決定するために費やします。巨大でフラットなネットワークに対して、何百もの横移動のチャンスをもたらす可能性があります。また、高価なD&Rプログラムでは管理が困難になっている傾向があります。攻撃者（犯罪者と侵入テスターの両方）驚かせ、コンプロマイズの出発点と、次のコンプロマイズのターゲットを求めて資産から資産へ移動するのを困難にすることです。

これら3つの基本的な情報セキュリティプラクティスの1つが不足しているサイトは、経験豊富な攻撃者によって簡単に侵害される可能性があります。正直に言うと、これらのプラクティスの1つ以上が実行されていないときに侵入テストに時間を費やしお金を払うのは、おそらく無駄なことです。[12]

適切なパッチ管理とすべてのユーザーとサービスアカウントのマシン生成パスワードを備えたセグメント化されたネットワークなら、ペネトレーションテスターが現れたときに最も安全です。これにより、クライアントは攻撃的なセキュリティの見返りを確実に得ることができます。侵入テストは、実際にはテクノロジーとポリシーのテストに関するものではありません。それはQAとHRの仕事です。代わりに、ペンテストにおいては、それらのテクノロジとポリシーに関するテストの前提について考えておくことをお勧めします。

侵入テストは、ほとんどの場合、そもそもプロセスが完全に欠けていることを指摘するのではなく、既存のプロセスとテクノロジーを段階的に改善することです。最も影響力のある侵入テストは、1つの孤立したシステムまたは忘れられたネットワークが、何らかの理由で社内スタッフによって見過ごされ、それらをどのように見つけ、それを侵害するために何をしたかを正確に報告するものです。結局のところ、侵入テストは、非常に複雑な情報技術の実行に関して、想定が現実と一致しない場所を知るための最も効果的な方法であり、最高の侵入テスト担当者は、これをすべての後処理の焦点にします。クライアントがそれに応じて想定を調整できるようにレポートします。

[12]とはいえ、これらの面の1つに取り組むにあたりビジネス上の正当性が必要なのであれば、侵入テストは、時間とリソースを費やすべきだと管理者に納得させるのに非常に役立つ手段でもあります！

お問い合わせ

japansales@rapid7.comまでメールにてご連絡ください。

RAPID7