セキュリティ衛生:
セキュリティ最前線

バージョン1.3

リリース日:　2021年5月

Securosis, L.L.C.　515 E.Carefree Highway Suite #766 Phoenix, AZ 85085　**T** 602-412-3051

info@securosis.com　www.securosis.com

作者のメモ

このレポートの内容は、いかなるスポンサーとも無関係に独立して作成されました。

元々Securosisブログに投稿した資料に基づいていますが、内容を拡充しレビューしたうえで、プロフェッショナルな編集が施されています。

Chris Pepper氏による編集とコンテンツのサポートに感謝します。

著作権

セキュリティ衛生

目次

セキュリティ衛生が保護に不可欠である理由

セキュリティの専門家として、何十年経っても、同じ問題や間違いを見ることになるのは、気が重くなります。グランドホッグ・デー（春の訪れを占う行事）のハッカー版にはまってしまったように感じられます。起床して、ユーザーまたは管理者によるミスを片付け、新しい攻撃を処理し、コンプライアンス・レポートを作成しても、次の日には、同じ事すべてを再び行う必要があります。もちろん、セキュリティとは非対称な世界です。
攻撃者は1回正しければ十分で、対象の環境に侵入できる可能性があります。防衛側は1回間違えるだけで、攻撃者が大きな足場を得るために利用されてしまいます。公平ではありませんが、そもそも人生は公平ではありません。

セキュリティ・プログラムの構築にあたって、すべての人に言っている最も基本的なアドバイスは、基礎を正しく扱うということです。

セキュリティ・プログラムの構築にあたって、すべての人に言っている最も基本的なアドバイスは、基礎を正しく扱うということです。セキュリティの基礎を覚えているでしょうか。すべての資産の可視性です。これらの資産の強力なセキュリティ構成と状況を維持することです。ベンダーが更新を発行したときに、効率的かつ効果的にパッチをシステムに適用することです。ほとんどの専門家は、基礎が大事という話に同意するものの、敵の組立ラインで作られた最新のマルウェアの仕組みを理解するのに丸一日をかけたり、自社環境での脅威ハンティングに数日かけたり、単純に自分の所では起きないだろうとのんきに構えていたりします。いわゆる楽な仕事です。基礎というのは…退屈です。

しかし実際その基礎がよく効きます。すべての攻撃に効くとは言いませんが、大部分には有効です。そもそも基礎と呼ばれるのはそのためです。このレポートは、それを思い起こすためのリマインダです。すべてのリスクを排除することはできませんが、敵が自社の環境に簡単に足場を作れる状況を放置しているのであれば、反省しなければなりません。最小抵抗経路を塞ぎ、スクリプト・キディや凡庸なハッカーなど、取るに足りない対戦相手に負けないようにするというだけの話です。環境に侵入するために敵に多大な労力をかけさせることができれば、侵入に失敗したり、検知に引っ掛かったり、今後の調査に利用できる痕跡を残させる可能性が高くなります。

多数の失敗事例

長年にわたる数千もの侵害事例のリストを調査した結果、かなりの数の事例は、構成の誤り、既知の脆弱性の未修正、ベンダー・パッチのインストールの失敗などが原因であることが判明しました。セキュリティ衛生の失敗によるマイナスの影響について把握するために、3件の侵害事例について詳しく確認しましょう。

- **Microsoft Exchange:** 最近注目を集めた侵害事例では、オンプレミスにインストールされたExchangeサーバーが攻撃され、攻撃者がサーバーにフル・アクセスできるようになりました。その後何度もランサムウェア攻撃に見舞われたため、これらの重要なコンポーネントを最新の状態に保つ必要性が浮き彫りになりました。

- **Equifax:** Equifaxは、インターネットに接続しているサーバーのApache Struts攻撃に対する脆弱性を、パッチを適用することなく放置したため、リモート・コードを実行できる状態になっていました。当時、パッチはApacheから入手できましたが、同社はすべてのシステムに適用することを怠りました。さらに悪いことに、Opsチームがパッチ未適用のシステムをチェックしたものの、未適用のシステムを検出できませんでした。しかしながら、実際には脆弱なシステムは依然として存在していました。これは、最終的に1億件以上の個人情報が盗まれた代表的な衛生失敗事例です。Equifaxは、責任を取って数億ドルを支払うことになりました。身震いがするような恐ろしい事例です。

- **Citrix:** 主要なテクノロジ・コンポーネントが更新された場合、パッチを適用する必要があります。攻撃者は、パッチをリバース・エンジニアリングして、攻撃の糸口にする脆弱性を見つけないとも限りません。この状況は、特に2020年初頭に起きたCitrixのハッキング事例で問題になりました。攻撃者が、自動検索を実行して脆弱なデバイスを検索できる状態になっていたからです。当然ながら、攻撃者は脆弱なデバイスの検索を実行しました。Citrixが提案した当初の緩和策は、パッチではなく、信頼性も低く、顧客ベース内で幅広く実装されているわけでもないため、多くの組織が攻撃にさらされたままになりました。同時に、広く拡散されたエクスプロイト・コードによって、攻撃が簡単になりました。Citrixがパッチを発行した後、顧客は迅速にパッチを適用し、実質的に攻撃は停止しました。パッチの適用は効果的ですが、効果があるのは実際に適用した場合に限ります。

好き好んで、侵害で苦労した会社をあげつらっているわけではありませんが、そういった会社の間違いから学ぶ必要があります。インフラストラクチャの衛生は、複雑化の一途をたどっています。2020年末に発生したSolarWinds攻撃は、正しく行動して、ツールにパッチを適用しても、攻撃者にアクセスを提供してしまうという例でした。状況を単独で見てみると、「では、なぜパッチなんて適用するのか」と疑問に感じるかもしれません。

そういった疑問は、あなたが誤った教訓を学んだことを示しています。少し前の項に、「すべてのリスクを排除することはできません」と明記されています。サプライ・チェーン攻撃が発生したときに、率直に言って、検知と監視に集中すること以外に、あまりできることはありません。ただし、コンポーネントにパッチを適用していないと、エクスプロイト・コードを持っている人にシステムが開放されます。

選択の余地はない

上記のあらゆる惨状を知った後でも、正しいインフラストラクチャ衛生を実施することに依然として抵抗があるとします。オラクルを信じないでください。高強度で構成された状態を維持し、パッチを適用することができない場合に、あらゆる欠陥を発見（および報告）してくれる監査者に耳を傾けてください。パッチの適用を義務付けている規制要件をいくつかご紹介します。

- **PCI:** 要件2、6、および11でパッチ適用について規定しています。
- **ISO 27001:** 規制A.12.6.1で脆弱性の修正（パッチ適用）について規定しています。
- **GDPR第25条:** 「設計とデフォルトによるデータ保護」と第32条: 「処理のセキュリティ」は、顧客データを保護するシステムを導入する必要があることと、システムの衛生状態が保たれていない場合、当該の顧客データを保護できないことを示しています。

> 好き好んで、侵害で苦労した会社をあげつらっているわけではありませんが、そういった会社の間違いから学ぶ必要があります。

- **NIST SP 800-53 R3:** 構成管理（CM）、リスク評価（RA）、システムおよび情報（SI）では、インフラストラクチャにパッチを適用する必要があることが強調されています。

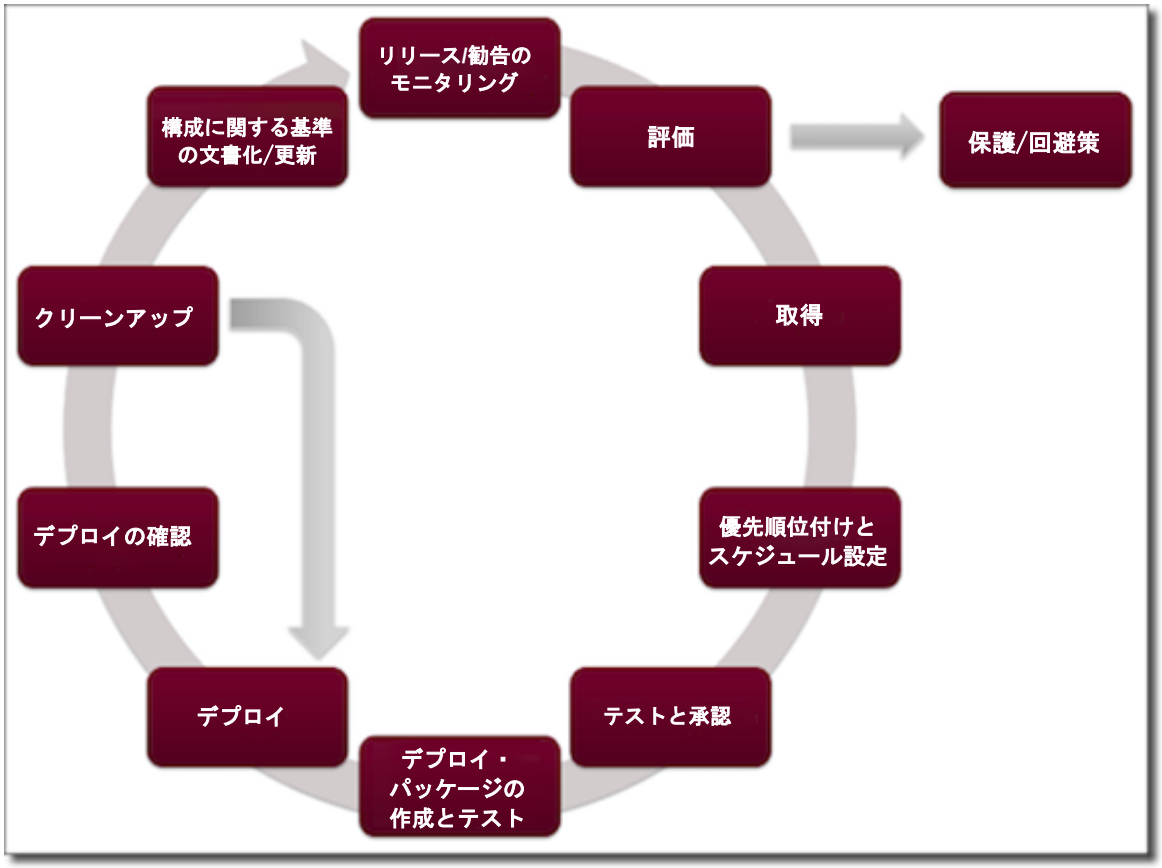
選択の余地はないことは明白です。

脆弱性の修正

前述のように、セキュリティに関する最も基本的なアドバイスは、基礎の正しい実施です。
これは、意志が固く、資金が充実した敵からの脅威を断絶することはできませんが、ほとんどの攻撃者が狙う最小抵抗経路を除去できます。

それだけは十分でなく、攻撃者は多くの攻撃の偵察を自動化しているため、現在では、誤りが許容される余地はほぼありません。つまり、脆弱性を露出させておけば、敵に発見されます。攻撃者は、ボットやスクリプトを使用して、弱いリンクを常に検索しています。

とはいえ、このレポートを読んでいるのは、セキュリティの実施に関する課題を何度も聞くためではないので、フォーカスを問題の修正方法に移しましょう。



迅速かつ完全に修正する

ベンダー自体は、自社の製品で見つかった脆弱性について、更新やパッチを発行します。顧客は、最新の状態と安全性を保つために自社のシステムにパッチを適用します。オラクルは業界として長年にわたってパッチを適用してきました。そして、Securosisでは、同じくらい長い間、パッチを研究してきました。気になる場合は、タイム・マシンで、当初のProject Quantにおけるパッチ適用に関する独創的な仕事を確認してきてください。

上の図は、かつて2009年に当社が定義した詳細なパッチ適用プロセスを示しています。効果的にパッチを適用するには、信頼性の高い一貫したプロセスが必要です。コンポーネントを停止させるパッチのデプロイには深刻なデメリットがあるため、特にテストおよび承認ステップの重要性について説明します。

しかし、堅牢なパッチ適用プロセスを実施するには、数日から1か月程度の時間がかかります。多くの大企業が、リリースから1か月以内にパッチが適用されることを想定しています。しかし実際には、積極的に攻撃されている問題に対処する注目度の高いパッチを適用するには、数週間でも長過ぎる可能性があります。非常にリスクの高い脆弱性に対処するパッチには、優先度の高いパッチ適用プロセスが必要です。本質的な要件は、高優先度（サイクル外）のパッチ適用作業をトリガーする基準と、通常のパッチ適用プロセスのどの部分をスキップするかを確立して同意することです。

または、一時的な仮想パッチを使用することもできます。仮想パッチは、攻撃のシグネチャに基づいて、パッチが適用されていない脆弱性に対する攻撃からの保護を試みます。これは、その攻撃にシグネチャを作成するための識別可能なパターンが確立されている場合に限られます。その場合でも、シグネチャによる攻撃対象領域の保護では不十分です。メリットは、仮想パッチを迅速にデプロイできる点です。しかし、新しいハイリスクの脆弱性に関する情報は大抵希薄で曖昧なことを考えると、できるだけ、回避策に最適な識別可能パターンが組み込まれるようにすることや、攻撃が変形してシグネチャを変えないようにすることは、不可能でないにしても困難です。これはすべて、一時的な仮想パッチを提供するベンダーが、どれだけ完璧にパターンを特定できるかに帰着します。不十分だった場合は、攻撃の大半は引き続き侵入できます。また、ベンダーがパターンを過剰に指定すると、正規のトランザクションをブロックして、アプリケーションが破壊されます。さらに、瑕疵がないことを確認するために多数のテストを実行する必要があります。最終的にパッチ適用に比べて大して時間を節約できないこともあります。考慮事項が増えすぎて、それこそが結局本物のリスクになります。

一時的な仮想パッチのもう1つのデメリットは、脆弱性のあるコンポーネントに向かうすべてのトラフィックは、検査ポイントまたは緩和ロジックを経由して実行する必要がある点です。トラフィックがコンポーネントに直接到達できるのであれば、一時的な仮想パッチは役に立ちません。たとえば、データベースを保護するために境界セキュリティ・デバイスに仮想パッチがデプロイされている場合、データベースに直接アクセスできる内部者は仮想パッチをバイパスして、パッチ未適用のデータベースを悪用できます。このコンテキストでは、「内部者」とは、単純に境界の内側に足場を築いた敵、または管理者の資格情報セットを入手した敵の場合もあります。

攻撃パターンが認識された後に、何をすべきかも明確に理解する必要があります。接続全体を強制終了すると、同じ接続プールを使用している多くの他のユーザーに影響したり、さまざまな予想外の方法でアプリケーションの動作に影響したりする可能性があります。

パッチがまだ作成されていない、ダウンタイムなどのメンテナンス上の問題があるといった理由で、高優先度の脆弱性にただちにパッチを適用できない場合、一時的な仮想パッチが重要な短期間の代替手段として役立つ場合があります。ただし、コンポーネントは修正していないことを忘れないでください。仮想パッチの陰にコンポーネントを隠しているに過ぎません。オラクルでは、30年の経験から、攻撃者が脆弱なシステムを発見しないことを祈ることで、発見されないで済むことは一切ないことを明言します。

検知や一時的な仮想パッチを逃れるために敵が攻撃のシグネチャを簡単に変更できるとすると、完全な検知シグネチャの作成は不可能であり、すべてのトラフィックに検査ポイントを通過させることが困難であるという事実により、ベンダー・パッチのデプロイが唯一の長期的ソリューションとなります。長期的なソリューションについて言えば...

共同責任の完全活用

クラウド革命で最も説得力があったことの1つは、一部のインフラストラクチャ・コンポーネントをプラットフォーム・サービス（PaaS）で置き換えるアイデアです。先程からこれについて示唆してきましたが、共同責任によってインフラストラクチャ衛生をどのように改善できるのかを、少し詳しく掘り下げてみましょう。まず、共同責任はクラウド・コンピューティングの基礎です。各クラウド・プロバイダは固有の責任を担っています。クラウド消費者（あなた）には連結するセキュリティ職責があります。連結しているのは共同責任です。

結局のところ、サービス・プロバイダに責任を移管すると、攻撃対象領域と運用対象領域の両方を削減できます。
そして、これはよいことです。長期的に、PaaSサービスを戦略的に使用することは、テクノロジ・スタックのリスクを軽減する優れた方法の1つです。

具体的な職責区分は、サービスとデリバリのモデル（SaaSまたはPaaS）によって異なりますが、インフラストラクチャ・コンポーネントにPaaSサービスを採用すると、そのコンポーネントの運用業務を免れることができます。セキュリティ・パッチを含め、スケーリングやメンテナンスについて心配する必要はありません。家族から離れ、サーバーやデータベースにホット・フィックスを実施する長い夜や週末が恋しくなるかもしれません。

結局のところ、サービス・プロバイダに責任を移管すると、攻撃対象領域と運用対象領域の両方を削減できます。そして、これはよいことです。長期的に、PaaSサービスを戦略的に使用することは、テクノロジ・スタックのリスクを軽減する優れた方法の1つです。確かにサービス・プロバイダが間違いを犯すこともありますが、リスクはかなり小さくなります。サービス・プロバイダには、脆弱性に対処し、顧客を安全に保つことについて、真摯に懸念し、少なからぬリソースを割り当てるという評判とブランド・エクイティがあります。

サプライ・チェーン

最近のSolarWindsの事例や、第三者請負業者への攻撃から始まった標的型攻撃（2013年から）から学べることがあるとすれば、衛生責任は、自社環境の境界では終わらないということです。前述のとおり、プロバイダおよびパートナーのインフラストラクチャ・コンポーネントを維持する責任はなくなるかもしれませんが、彼らに弱点があったときに自社環境にどのように影響するかについては責任を負います。

えっ、なんですって？もう少し詳しく説明します。外部のビジネス・パートナーが攻撃を受け、攻撃者が自社環境に侵入し、大惨事をもたらし始めたとしたら、どうなると思いますか。それはあなたの責任です。もちろん、パートナーの環境を保護するのはパートナーの責任であり、それに失敗したのだと主張することはできます。しかし、組織の監査委員会の前で、パートナーのリスク・プログラムが不十分だった理由を説明する際に、このような主張は役に立ちません。

運用上の支援を取得し、攻撃対象領域を減らすために共有責任モデルを活用したいのと同様に、リスク管理にさらなるリソースを費やして、何がリスクなのかを把握し、適切な修正アプローチを選択する必要があります。

成功と一貫性

インフラストラクチャのセキュリティ衛生に対する各アプローチは、相互に排他的ではないことを繰り返す必要があるでしょう。パッチはコンポーネントの脆弱性を排除しますが、一時的な仮想パッチによって、当座のリスクを一時的に軽減できることもあります。長期的に最善のソリューションは常に、ベンダーから直接提供されるパッチが関連し、PaaSサービスへの移行が含まれることもあります。ケースごとに最適なアプローチを見つけ、リスク、可用性、自社のアプリケーション・リファクタリング能力のバランスを取る必要があります。

短期間で成功を収める

優先度の高い脆弱性はいつでも発生します。それにどのように対処するかによって、セキュリティ・チームの能力と力量に対する評価が決まります。ここでは例として、地方銀行のような、ある小規模な金融サービス組織について考えてみましょう。顧客のローン・データの処理には自社製のクライアント/サーバー・アプリケーションを使用し、バックエンド処理にストアド・プロシージャを頻繁に使用しています。アプリケーション・チームはフロントエンドWebインタフェースを定期的に更新していますが、バックエンドは大部分が未変更です。これはよくある「壊れてないものは直すな」という状況です。アプリケーションは顧客（Webインタフェースを使用するユーザー）には最新に見え、バックエンドは十分に機能しています。しかし、大抵は、バックエンド・データベースに影響する注目の脆弱性に関するベンダー・アラートや、パッチの緊急リリースの警告を受けています。そのため、セキュリティ・チームは最善かつ最も安全な道を解明する必要があります。

優先度の高い脆弱性はいつでも発生します。それにどのように対処するかによって、セキュリティ・チームの能力と力量に対する評価が決まります。

このプロセスの最初のステップはリスク分析です。脅威データをざっと確認すると、エクスプロイトが出回っています。つまり、何も対処しないという選択肢はなく、時間がきわめて重要です。次に必要なのは、アプリケーションの重要性を認識することです。これは、先程顧客のローン・データを保持することとして説明されているため、ビジネスにとっても、銀行の規制監督の範囲内でも不可欠です。アプリケーションが使用されるのは通常業務時間中であるため、パッチは業務終了後に適用できます。

エクスプロイトが出回っているため、迅速な修正が必要です。なぜなら、セキュリティ研究者が、特定のクエリによってデータベースにアクセスできることを指摘しているからです。セキュリティ・チーム

は、境界IPSデバイスを使用して、特定の危険なクエリを検査してブロックする一時的な仮想パッチを実装します。攻撃パラメータの範囲がIPSブロックには広すぎることがよくありますが、この事例では、一時パッチは機能しました。

別の予防措置として、チームはデータベース周辺の監視を増やし、一時仮想パッチをすり抜ける可能性のある内部者アクティビティが発生したときにアラートを発するようにしました。追加の監視では、既に攻撃に成功したデバイスを使用してアプリケーションをバイパスし、データベースに直接危険なクエリを実行しようとすると検知します。

運用チームは、次回のメンテナンス・ウィンドウ中にベンダー・パッチを適用する必要があります。一時的な仮想パッチで、ベンダー・パッチをテストして、アプリケーションに悪影響がないことを確認するための時間を確保します。ベンダー・パッチ・テストで悪影響がないことが判明し、運用チームは次回のメンテナンス・ウィンドウで無事パッチを適用しました。

最後のステップは、次回のために改善点を特定する、プロセスの戦略的レビューです。ある時点で、アプリケーションはリファクタリングされ、銀行のクラウド・プレゼンスに移動されることになっていますが、今後24か月間には行われません。この優先順位を上げることに意味はあるのでしょうか。おそらくないでしょう。たぶん、次回の脆弱性では一時的な仮想パッチによる緩和が役に立たなくても、オフピーク時の更新で、アプリケーションの可用性に大きな影響を与えることなく、修正できます。アプリケーションのリファクタリングが開始されたら、チームは最初にいくつかのストアド・プロシージャをアプリケーション・サーバー層に移動し、後からデータをPaaSに移行して、アプリケーションの攻撃対象領域と運用対象領域の両方を減らすことを検討します。また、商用SaaSでアプリケーションを完全に置換できるかどうかも検討する必要があります。

組織的整合

上記のシナリオは、インフラストラクチャ衛生のすべてのオプションがうまく連携することで、優先度の高いデータベースの脆弱性のリスクを効果的に軽減できることを示しています。このプロセスには複数のチームが関与していました。セキュリティ・チームは、問題を特定し、修正の代替案を検討し、一時的な仮想パッチおよび監視の追加を決定しました。IT Opsチームは、ベンダー・パッチのテストおよびアプリケーションの管理で重要な役割を果たしました。アーキテクチャ・チームは、今後のアプリケーションのリファクタリングまたはSaaSオファリングへの移行を検討します。

効果的に連携するには、こうしたすべてのチームが、データの損失のないアプリケーションの可用性という目標を一致させ、協力する必要があります。しかし、プロセスの促進において重要な役割を担うもう1つのチームに言及する必要があります。財務です。回避策に使用する境界デバイスや、パッチへのアクセス、特に忘れがちなレガシー・アプリケーションのそれを確保するサポート/メンテナンス契約などの品目の支払いを行うのは財務チームです。インフラストラクチャを最高の状態で維持する上で技術的スキルが不可欠なように、技術者が自分の仕事を行えるようにリソースを確保することも同じくらい重要な仕事です。

このトピックに関する質問がある場合や、貴社の具体的な状況について話し合いたい場合は、
お気軽にinfo@securosis.comまでお問い合わせください。

アナリストについて

Mike Rothman、アナリスト兼社長

Mikeの大胆な見解と恐れることのないスタイルは非常に有効です。なぜなら、企業では、ダイナミックなセキュリティ脅威の現状に対処するための効果的な戦略を決定するためです。Mikeは、ネットワークやエンドポイントの保護、セキュリティ管理、コンプライアンスなど、セキュリティの中でも特に最先端の要素を専門としています。
Mikeは、20年の経験を積み、セキュリティに関してきわめて深い知識を持つ人物です。

Mikeのキャリアは、プログラマおよびネットワーキング・コンサルタントとして始まり、META Groupでアナリストを務めた後、SHYM Technologyを設立し、その後CipherTrustとTruSecureで役員を務めました。Mikeは、2006年にSecurity Inciteを創業し、大々的に宣伝していながら人々を失望させていたセキュリティ業界で良識的な意見を提供しました。eIQnetworksの戦略担当シニア・バイス・プレジデントをしばらく務めた後、Securosisに加わり、セキュリティの現状について再び鋭く評価するようになりました。

Mikeは、2007年に『The Pragmatic CSO』<https://www.pragmaticcso.com/> を出版し、技術指向のセキュリティ専門家を対象に、上級セキュリティ専門家になるために必要な資質の詳細を紹介しました。また、コーネル大学のオペレーションズ・リサーチおよび生産工学で非常に貴重な工学の学位を取得しています。彼の身内は、彼が日常では自分の教育レベルで話をしないことに喜んでいます。

Securosisについて

Securosis, LLCは、ソート・リーダーシップ、客観性、および透明性を専門とする独立系調査・分析会社です。Securosisのアナリストは、全員が役員レベルの地位を有しており、価値の高い実用的なアドバイザリ・サービスの提供に取り組んでいます。Securosisのサービスは次のとおりです。

- **一次調査出版**: Securosisの調査の大半は、ブログを通じて無料で公開し、年次ベースで配布ライセンスを有する論文集としてパッケージ化しています。公開されたすべての資料およびプレゼンテーションは、Securosisの厳格な客観性要件を満たし、当社のTotally Transparent Research（完全透明調査）ポリシーに従っています。

- **クラウド・セキュリティ・プロジェクト・アクセラレータ**: Securosis Project Accelerators（SPA）はパッケージ化されたコンサルティング・サービスです。Securosisの応用研究と実戦テスト済みのフィールド・エクスペリエンスをお客様のクラウド・デプロイメントに適用します。これらのきめ細かいプログラムは、評価、カスタマイズされたワークショップ、継続的なサポートを組み合せ、クラウド・プロジェクトを適切かつ迅速に保護できるようにします。これらのプログラムは、プロジェクトを数か月から数年短縮しながら、最先端のクラウド・セキュリティ・プラクティスを既存の業務に統合できるように設計されています。

- **クラウド・セキュリティ・トレーニング:** Securosisは、Cloud Security AllianceのCCSKトレーニング・クラスと、当社独自のAdvanced Cloud SecurityおよびApplied SecDevOpsプログラムを構築したチームです。パブリック・クラスにご参加ください。または、ご依頼に応じて、カスタマイズしたプライベート・クラスを開催します。

- **ベンダー向けアドバイザリ・サービス**: ベンダー・クライアントが、重要な市場要件に気が付く正しい方法で、適切な製品/サービスを市場に提供できるよう支援する、いくつかのアドバイザリ・サービスを提供しています。Securosisは、お客様が聞きたいことではなく、お客様が聞くべきことを明確に指摘することで知られています。クライアントは、通常、1日の戦略的エンゲージメントから始まり、継続的なサポートのために有料で当社に相談できます。Securosisのアドバイザリ・サービスの一部として提供されるサービスには、市場および製品の分析と戦略、テクノロジ・ロードマップのガイダンス、競争戦略などが含まれます。ただし、当社は、すべての関わりにおいて、厳格な客観性および機密要件を遵守する点に注意してください。

- **カスタム調査、講演、アドバイザリ**: 新しいテクノロジまたはセキュリティの問題に関するカスタム調査レポートが必要ですか。社内向けまたは公開のセキュリティ・イベントの高評価の講演者が必要ですか。合併または買収のデュー・デリジェンスに関する外部専門家が必要ですか。セキュリティ戦略を評価し、ギャップを特定し、ロードマップを進める専門家が必要ですか。これらの定義済みプロジェクトは、必要なサービスが、1日の戦略的エンゲージメントよりは長く、長期的なコンサルティング・エンゲージメントよりも短い場合にギャップを埋めます。

Securosisのクライアントは、ステルス・スタートアップから、一流の有名テクノロジ・ベンダーやエンド・ユーザーまでさまざまです。大型金融機関、機関投資家、中小企業、および大手セキュリティ・ベンダーのクライアントもいらっしゃいます。Securosisの詳細については、当社のWebサイト: <https://securosis.com/>を参照してください。