ポリシーガイド

効果的なコードおよびソフトウェア署名ポリシー作成ガイド

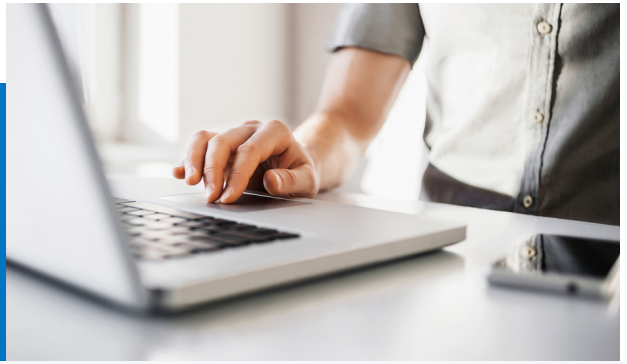
digicert®

ソフトウェアとコードサイニング：文書化された強力な署名ポリシーの重要性

コードサイニングはソフトウェアを開発し保護するプロセスに欠かすことのできないステップです。しかし、多くの開発チームやエンジニアリングチームは、一貫した署名の実装に苦労しています。その主な理由は、署名がアジャイル開発におけるタイムトゥマーケットの目標を阻むという誤解です。さらに、当社の調査によれば、多くのソフトウェア開発チームは署名プロセスを管理する明文化された包括的なポリシーがないまま、一般化されたガイドラインと暗黙的な慣行に則った運用をしています。

より高度な署名要件のニーズが高まる中、組織と開発チームはソフトウェアを保護するためにしっかりとした署名ポリシーおよびプロセスを作成し、運用する必要があります。正式なコード署名ポリシーを作成して文書化することで、開発者とエンジニアは開発したソフトウェアを保護するために必要なベストプラクティスに従って行動できるようになります。

本ガイドは、署名手法においてエンドツーエンドのセキュリティを実現するためのソフトウェア、プロセス、ポリシーを組み合わせた戦略を策定するのにお役立ちいただけます。本ガイドを作成するにあたり、デジサートでは数十人の DevOps エキスパート、チームリーダー、ソフトウェアセキュリティの専門家にアンケートと取材を実施しました。さまざまな組織規模、業界、地域からのエキスパートの皆様のおかげで、事実上どのような組織にも、どのようなソフトウェア開発チームにも役立つ署名のベースラインを作成することができました。



本ガイドの使い方

パート I では、チームのニーズに応じた署名ポリシーの土台を作成するために役立つ主なコンセプトを紹介します。コードサイニングを実施することで、いかにチームが開発するソフトウェアにとって最善の状態を実現できるかについて理解を深めます。

パート II では、私たちがご意見を伺ったソフトウェアセキュリティおよび DevOps エキスパートが重要視している手法や手順を交えながら、明文化されたポリシーの作成についてまとめます。

最後に、オリジナルのポリシーを作成する際に参照できるリストを掲載します。

パート I

適切なソフトウェアセキュリティをどのように考えるのか

開発チームには明確な目標のビジョンが必要

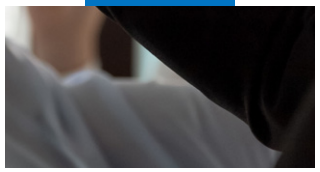
御社のソフトウェア開発チームのミッションとは？

この質問への答えは思ったほど明白ではないかもしれません。組織のミッションや、別のチームまたは事業部門のミッションが、チームのミッションの代わりになってしまっていることが多くあります。チームのミッションは組織のミッションを実現するためにあるべきですが、そのミッションは同じではありません。人事部のガイドラインに従って財務部を運用してもうまくいかないのと同じように、営業部のシステムを使用して人事部を運用することができないのと同じように、ソフトウェア開発チームを別のチームまたは組織全体の原則やガイドラインに従って運用しても、最高のパフォーマンスを発揮することはできません。

ソフトウェア開発チームが最高の成果をあげるには、チーム全員がソフトウェアにとって最善な行動を取れるようにチームの行動指針を定める強力かつ明確なミッションステートメントを掲げ、それに向かって動く必要があります。セキュリティが関係する場合は特にそうです。ソフトウェア開発チームのパラメーターおよび機能を用いてミッションを定めると、短期的なビジネスまたは売上の目標を達成するための受け身の作業から、長期的な目標を達成するための能動的な作業へシフトしていきます。

このガイドを読みながら、ぜひ御社のソフトウェア開発プロセスについて考えてみてください。ソフトウェア開発チーム独自のミッションステートメントがまだない場合は作成してください。ミッションステートメントを考えるときは次のことを覚えておいてください。強力なミッションステートメントは、チームの働き方を定めるものでもなければ、組織または顧客が期待するメリットを定めるものでもありません。強力なミッションステートメントは、御社が開発するソフトウェアにとって何が最善であるのかと、その理由を定めるものです。

優れたミッションステートメントは、シンプルで覚えやすく、専門用語は使いません。役立つルール：読点と箇条書きの使用は避けましょう。

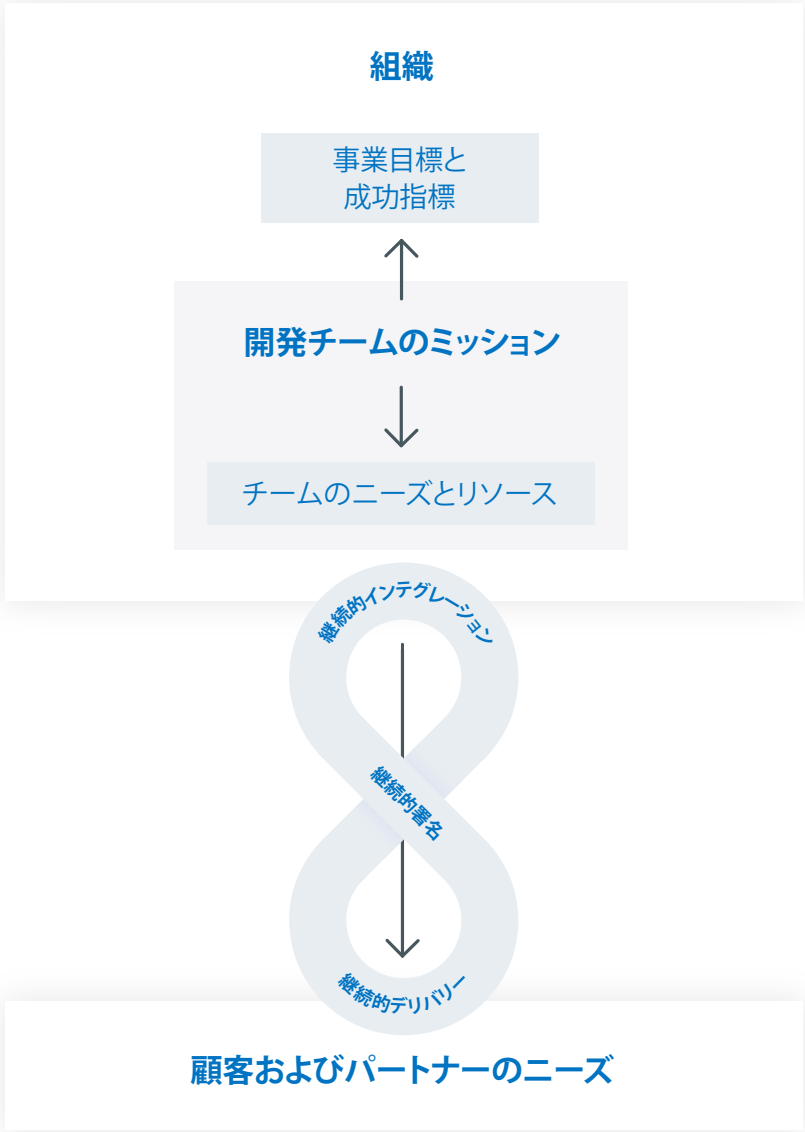


セキュリティは強力なミッションから生まれる

昔から、ソフトウェア開発におけるセキュリティ（特に署名）は多くの時間と手間を要する作業でした。コードとソフトウェアのセキュリティを適切に保護するための余分な手順が、短期的なビジネス目標達成の障害となった例は数えきれません。開発者とエンジニアはソフトウェアにとっての最善を望んでいますが、署名は（プラットフォームやソフトウェアサプライチェーンによって義務づけられていない限り）将来を見据えた長期的目標の範疇にあります。ソフトウェア開発、特にアジャイル開発の采配において、目先の短期的なニーズや市場の反応を優先するあまり、長期的目標がないがしろにされることは珍しくありません。

そこで力を発揮するのが強力なソフトウェア開発ミッションステートメントです。強力なミッションステートメントは、ソフトウェア開発のミッションを完遂するために必要なあらゆるステップとリソースを把握します。このミッションステートメントは対内的に働き、開発チームのニーズを満たします。さらに、組織構造をさかのぼって働き、ビジネス目標と成果指標を達成するのにも役立ちます。そして最後に、対外的に働き、パートナーとソフトウェアを使用する顧客のニーズを満たすのにも役立ちます。

強力なソフトウェア開発ミッションステートメントを定めることで、署名をはじめとするあらゆるセキュリティ手順は、完全な署名付きのコードを期限内にデリバリーすることで開発者とエンジニアが短期的な事業目標を達成できる、将来を見据えた長期的な開発プロセスに欠かせない要素になります。作成するあらゆるポリシーガイドは、ソフトウェア開発のミッションステートメントをよりどころにする必要があります。



文化と役割を理解する

ソフトウェア署名ポリシーにおいて、ガイドはただの規則集であってはなりません。開発チームが開発したソフトウェアにとって最善のことをするうえでソフトウェア署名がどのように役立つかを説明するのが最高のガイドです。先ほどから述べている文脈において、強力なソフトウェア署名ポリシーは、目標を遂行してチームのミッションを達成するために署名を正しく使用する方法をチーム全員に周知するのに役立ちます。

そのためには、チーム内の役割の関係性を知り、これらの役割によって署名ポリシーのコミュニケーションと実装がどのように定義されているかを理解することが重要です。ポリシーが明確に定められ、そのベネフィットおよびソフトウェア開発プロセスの全体的な取り組みが理解されると、コードのセキュリティを保護することは、チェックリストのただの 1 ステップやマネージャーと開発者/エンジニア間の火種ではなく、チームの文化の一部となることが当社の調査で分かっています。

まず、以下の 3 つの原則から始めましょう。

1 ルールは管理者のため、プロセスは開発者とエンジニアのため

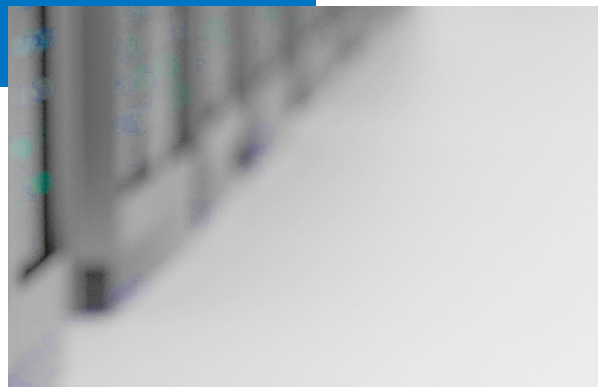
いつ、どのように署名を実施するかを定める明文化されたポリシーを作成することは重要ですが、そのルールを根本に据える必要があります。管理者とマネージャーは、日々の業務での署名の使用方法を説明したポリシープロセスを開発者とエンジニアに徹底させる必要があります。

2 署名をただ強制するのではなく、可能にする

大抵の場合、多くの人々は正しいことをしたいものです。開発者とエンジニアは、自身が作成したコードについて正しいことをしたいと考えています。ソフトウェア署名ポリシーは、セキュリティ規則やコンプライアンス要件を強制するための取り締まりツールとして運用すべきではありません。それは開発者とエンジニアが一生懸命作成した成果物を保護し、支持するのを楽にするものであるべきです。署名ポリシーは優れたセキュリティ手法を可能にするものです。

3 コミュニケーションが重要

ポリシーが曖昧で紛らわしかったり、非現実的、または厳しすぎる場合、ミッション達成の障害となります。ソフトウェア署名ポリシーが明確で、論理的で、分かりやすく、実施しやすいものになっているかを確認し、期待された通りに署名を遂行する方法についてポリシーを利用する人全員に定期的に周知させます。コミュニケーションは双方向であることに注意してください。ポリシーを効果的なものにするには、実際にポリシーを運用する人全員の意見を取り入れ、定期的にレビューしてアップデートします。



潜在的な障害となり得る点を見分ける

人間は単純化したがる生き物です。何か問題が発生すると、見逃した 1 つのエラー、ある 1 人のミス、ある一瞬の不注意、1 つの異常など、何か 1 つの原因を見つけようとします。しかし往々にして、チェーン（または網目）のように要因が複雑に絡み合ってあるとき臨界点に達したというのが真相であることがほとんどです。

ニュースの見出しを見ると、ソフトウェアサプライチェーンのセキュリティ侵害がある 1 人の怠慢な従業員、ある 1 人の極悪なハッカーによって引き起こされたように報じられるものばかりですが、本当は、ソフトウェア開発の脆弱性は複雑で、ごく一般的に存在しています。近年、より多くのソフトウェアが開発され、より多くの場所で導入されるようになったことで、この複雑性はさらに深刻化しています。複雑性を解決するソリューションがない限り、システムの脆弱性は倍々に増加し、個々のエラーの数が増え、それが見つかって利用されるのは単に時間の問題に過ぎなくなります。

不注意、怠慢、ごくまれに起きるヒューマンエラーがセキュリティ障害の原因であるという考え方から脱却しない限り、ソフトウェアへの攻撃の数と規模が拡大することは間違いありません。確かに単純なエラーの場合もありますが、ソフトウェアセキュリティの脆弱性の多くは、遡ればポリシーやプロセスの組織的な問題に行き着きます。そしてこれらの問題は通常、誤った情報、非現実的な期待、見落とし、使いづらいツール、またはツールの不在によって引き起こされています。

セキュリティ障害は、通常、単純なヒューマンエラーによって起きるわけではありません。開発するソフトウェアにとって正しいことをするために必要なツールと指針がソフトウェア開発チームに与えられていないという組織的な問題の結果として起きるのです。

優れたセキュリティは
優れたセキュリティ文化から

強力なソフトウェア署名ポリシーとは、優れたセキュリティ文化を明文化したものです。ソフトウェアにとって正しいことを行う権限をソフトウェア開発チームに与える文化は、トップダウンのルールや強制ではなく、強力なソフトウェア開発ミッションと共同作業と能力の付与に基づきます。また、セキュリティを組織的なものとして捉え、優れたツールを使用した適切なセキュリティ手法を促進します。

「セキュリティは企業に所属する 1 人 1 人にとって重要です。セキュリティの文化を育むには、好奇心を刺激して、いつまでも学び続けたいという気持ちを深い部分から掘り起こします」

DigiCert 最高技術責任者 (CTO)、
Jason Sabin

弱い文化

- 短期的、場当たり的な姿勢
- トップダウン
- 強制的な考え方
- 非現実的な個人責任
- 手動ツール

強い文化

- ミッションに基づく先取的な姿勢
- 相互にプラスになる関係
- 可能にするという考え方
- 組織的なサポート
- 自動化ツール

パート II

ソフトウェア署名ポリシーの作成

専門知識を集結する

ここまでで、独自のソフトウェア署名ポリシーと優れたセキュリティ文化のパラメーターを決める方法について考える機会があったはずです。ビジョンを思い描き、チームの明確なミッションステートメントを作成しました。ここからは、詳細項目を埋めて、チームメンバーに伝えられるポリシーを作成していきましょう。次のステップには、独自のミッション、プロセス、ニーズ、ツールをもつあなたのチームが組織全体の中でどのような位置付けにあるかを決定することが含まれています。

世界中の数百の業界にわたるソフトウェア開発者と関わる中で、ソフトウェア署名ポリシーが組織内の別のチームや部門によって伝えられているケースも少なくありませんでした。ソフトウェア署名ポリシーの作成元として最も多く名前が挙がったものを以下に示します。

- 最高情報セキュリティ責任者（CISO）
- セキュリティ運用/情報セキュリティ部門
- リスク管理部門
- 製品管理部門
- コンプライアンス部門
- 顧客およびパートナー
- 品質保証およびサイトリライアビリティ部門

リソースについて相談する場合、以下を考慮すると役立ちます。

1

一体性
署名について先取的で成熟したセキュリティ体制を実現するために、その部門はどのように役立つでしょうか?その部門が作成したポリシーとプロセスのどの部分があなたのチームのミッションを実現するのに役立ちますか?

2

相互にプラスになる関係
あなたのチームのポリシーとプロセスは、その部門のポリシーとプロセスにどの程度あてはまりますか?あなたのチームのミッションを実現する過程で署名ポリシーおよびプロセスに違反することなく共同作業するにはどうすればよいですか?

3

アクション
その部門が実行可能かつタイムリーな情報を提供するために、あなたのチームのリソースをどのように有効化できますか?どの情報が重要で、どの情報が非現実的な情報または参考用の情報ですか?

フォレンジック調査の重要性

最高のセキュリティは先回りして未然に防ぐことです。しかし、過ちは大切な学びの機会を与えてくれます。最近のセキュリティ障害や失策について定期的にフォレンジック調査を実施しましょう。自社で侵害や失策がなかった場合は、他社で発生した侵害を調査しましょう。エラーや脆弱性は、脅威のランドスケープを縮小して、コードをより安全に保護する署名ポリシーを作成するのに役立ちます。

ソフトウェア署名および使用方法の基礎

署名システムの個々の構成やチーム構成は組織によって異なることがありますが、当社の調査によって、チームの規模、ソフトウェアの種類、業界を問わず、開発チームが挙げた強力なソフトウェア署名ポリシーの構成要素にはいくつかの共通項があることが分かりました。これらの要素は署名のベストプラクティスの基礎となります。

鍵の発行

これは鍵管理の機能ですが、当社の調査で DevOps 専門家が最も重要視していたのがこの要素でした。管理プロトコルを定めるときは、誰が、いつ鍵を発行できるかについて管理策と手順を設定してください。チームメンバーは自身の役割に従って、鍵を発行するのに適切なタイミングを知る必要があります。発行する鍵の種類や、鍵に関連付けられる属性も管理する必要があります。そうすることで、攻撃ベクトルの元となる脆弱なアルゴリズムで新しい鍵を発行せずにすみます。

鍵の管理

ユーザーの役割に則った管理策と手順を設定します。誰が鍵を管理するのか。鍵の管理はチーム、組織、制作段階における役割に基づいて、どのように分担されるか。これらの手順には、組織全体のすべての鍵についてのロケーション追跡が含まれる必要があります。

鍵の保管

鍵は保護されなければなりません。使用中および非使用時の鍵の保管について管理策と手順を設定します。これには HSM、トークン、USB、多要素認証を使用した鍵へのアクセスが含まれます。チームメンバーは、鍵を紛失したり、不適切に保管したりしないようにする方法を知っていなければなりません。

署名の権限

チームメンバーは、誰が署名の権限を持っているか、いつ署名を行うべきかを知っていなければなりません。また、署名の権限が付与されないケースと、自身の役割に権限がない場合に署名をリクエストする方法も知っている必要があります。

鍵の使用方法

鍵をどのように使用するか（使用しないか）は、適切な署名手順の重要な要素です。鍵は、適切な人の手で適切なタイミングで使用されなければなりません。

鍵の共有の防止

鍵の共有は珍しいことではありませんが、最も危険な慣行でもあります。レポジトリ内、または内部サーバー、システム、ネットワーク内であったとしても、チームメンバーは決して鍵を共有してはなりません。適切な個人が、その役割に基づいて、一意の鍵を発行、管理、保管する必要があります。

継続的署名

コードおよびソフトウェア署名は、後から思いついたように付け加えたり、面倒なコンプライアンス手順として取り扱ってはいけません。あらゆる CI/CD パイプラインに 継続的署名（Continuous Signing: CS）を含め、コードのセキュリティが適切に一貫性をもって守られるようにしてください。

社内用途の署名を軽視しない

ソフトウェアセキュリティにまつわる 1 つの真実は、ベストプラクティスとプラットフォームによる義務化は相性が良いということです。アプリストア、オペレーティングシステム、規定されたブラウザーにデプロイすることを目的にソフトウェアを開発する場合、コード署名がより一貫して見られることが分かっています。ソフトウェアを実行するために署名が必要であれば、そのコードには署名が付けられます。しかしながら、こういった要件はおまじないで設けているわけでも、ただやりたいからやっているわけでもありません。署名には、実際にソフトウェアを破損、侵害、その他の悪意から守る効果があります。各種プラットフォームが署名を義務化するのは、高い理想を押し付けているのではなく、脅威への現実的なソリューションを求めているのです。

では、デプロイの必須要件になっていないとき、ソフトウェア署名があっという間にめったに行われないオプションと化してしまうのはなぜでしょうか。署名が、ベストプラクティスとして現実的な安全策であると分かっているなら、プラットフォームで署名が義務づけられていなかったとしてもソフトウェアは署名されるのではないでしょうか。

社内ソフトウェアのセキュリティに関する誤解

ほとんどの場合、組織は既知の開発者チームと、自社システムを取り巻くセキュリティを信頼しているため、社内のソフトウェア署名は無視されています。コードがひとたび世に出ると、ファイアウォール外の空間には脅威が潜んでいます。組織内では、チーム間で授受されたソフトウェアや、社内用途でサーバーにデプロイされたソフトウェアは、シールドされていると思いがちです。

しかし実際には、社内のソフトウェアも攻撃に対して脆弱です。誰かがシステムへのアクセスに成功した場合、未署名のコードは組織全体のインフラストラクチャーに害をなすのに使用できる格好のターゲットになります。ベストプラクティスとしては、社内ソフトウェアへの署名は理想ではなく、悪意のある第三者に利用されないように保護を実装するのと同じように、ソフトウェアと組織を守るための現実的な方策なのです。

自動化による解決

義務化が署名を促すというのがソフトウェアセキュリティの 1 つの真実だとしたら、もう 1 つの真実は時間と手間がかかるセキュリティ手順は DevOps の哲学の対極にあるということです。平均的な組織の各種社内システムを取り巻く保護の多さを考えれば、リスクが低く思える場合に開発者とエンジニアが足手まといになる手順を飛ばしたくなるのは理解できます。ハッキングや改ざんほどダメージは大きくないものの、手作業によってコード署名を行うためにかかる時間と手間自体が問題になることもよくあります。

それを解決するのが自動化と説明責任の組み合わせです。継続的署名を利用するシステムにより、どこで開発/デプロイされたかを問わずソフトウェアを保護するプロセスが実現され、開発者はいつでもすぐに署名を簡単に実行できるようになります。

最新のマネージドソフトウェア署名ツールはセキュアな署名を容易にします。「これらのツールを使用すると、何の負担もなしにパイプラインに署名を追加できるため、セキュリティのレベルを引き上げることができます。念には念を入れた安全対策を現実にする良い例です」

DigiCert エンジニアリング担当バイスプレジデント、Wade Choules

教育、トレーニング、フィードバック

ソフトウェア署名ポリシーの作成が終わったら、開発者とエンジニアに研修と周知を行うための体系が必要になります。以下のカリキュラムは、安全なソフトウェア署名を実践する方法をチームに教える際のテンプレートとして使用できます。

トレーニング

初期のトレーニング体系には、以下の構成要素が含まれます。

- ソフトウェア開発チームのビジョンとミッション
- ソフトウェアセキュリティの利害関係者
- ポリシーの紹介
- 署名が開発プロセスの他のステップと同じくらい重要である理由
- 組織内の署名の役割
- 署名プロセスの中核的原則
- 誰が、いつ、どのように署名するか
- 適切な鍵の使用方法と鍵の属性
- タイムスタンプを付けるタイミング
- 鍵のセキュリティと管理
- 問題をレポートする方法

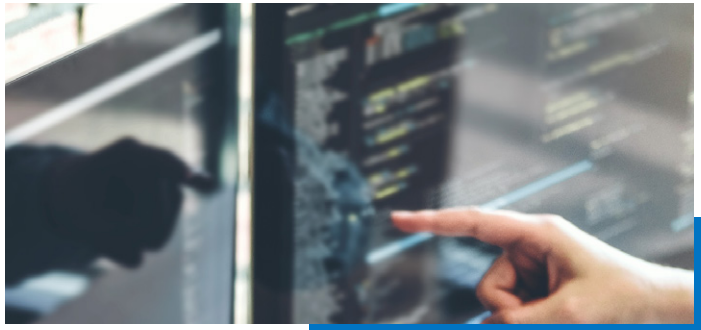
トレーニング体系にはこれらの構成要素に加え、理解度チェックを含めるようにしてください。また、新しいチームメンバーに意見や感想を述べたり、トレーニングに含まれない内容について質問する機会を設けることをお勧めします。いただいた意見や質問を元に、トレーニングモジュールを改訂してより万全な内容にしていきましょう。

教育

継続的な教育は知識を補足して理解を深めるのに役立ちます。定期的にポリシーとプロセスのレビューを行うことをお勧めします。さらに、プロセスのステップの重要性について、その文脈を提供すると役立ちます。ぜひ以下のことを検討してください。

チームメンバーからの意見や懸念事項を募るための場を含む、署名ポリシーおよびプロセスの定期的なレビュー。これにはトレーニングモジュールを割り当ててもよいですし、質疑応答ありのミーティング、その他の形式のディスカッションややり取りでも可能です。

現実世界におけるソフトウェア署名の重要性に関するニュースやお知らせを随時配信。ソフトウェアセキュリティに関するディスカッションフォーラム。ソフトウェアセキュリティに関するチームメンバー間で提案や懸念事項を共有するための専用のスペース（メールアカウント、プライベートの Slack チャンネルなど）。



コミュニティが重要

セキュリティの特性からして、情報を共有すると攻撃にさらされるような気がするかもしれません。しかし、サプライチェーンが拡大して複雑さが高まる中、共同作業と情報共有は世界中のソフトウェアセキュリティにとって不可欠になりつつあります。

攻撃の数と規模が拡大する中、ソフトウェア署名ポリシーについては、ソフトウェア開発者とセキュリティ専門家がベストプラクティスを共有し、有効な対策情報を交換し合うことが重要です。個々のセキュリティ構成や知的財産は公開すべきではありませんが、あなたの会社のソフトウェアを守る強固な情報がサプライチェーンの他社を守るかもしれませんし、その逆もあり得ます。

自社のソフトウェア署名ポリシーガイドを作成し、実施した後は、つながりのある他の専門家と、ポリシーとプロセスによってセキュリティを強化した方法について話し合います。他の会社でどのような施策を行っているかを聞き、コミュニティ全体でより強固なセキュリティ体制を採用する機運を盛り上げましょう。ただ 1 つの組織も、セキュリティに対して先取的なアプローチを導入しないという選択肢を取ることはできません。チェーン全体を安全にするには、各組織の各チームが各自のコードを守る必要があります。

ソフトウェア署名ポリシーの作成

1

ソフトウェア開発チームのミッションステートメントを作成する

ビジョンを描くこと。ミッションステートメントには、ソフトウェアにとって何が最善かとその理由を定義します。

2

ポリシーを定義して施行する上で、各役割がどのような貢献をするか検討します。

署名ポリシーは相互にプラスになる関係と共同作業をベースに作成し、明確で分かりやすく実践しやすいものにします。

3

適切なセキュリティ手法の組織的ランドスケープにおける署名の役割を計画する

CI/CD パイプライン全体で継続的署名が（強制されているのではなく）可能になっていることを確認します。システムは署名ベストプラクティスをソフトウェア開発プロセスのシンプルな一部にする必要があります。

4

内部および外部の情報源から情報を集める

内部利害関係者および顧客のノウハウとニーズを活用して、成熟したセキュリティ機能を含む優れたソフトウェアを期日までに提供することを可能にする署名手法を策定します。障害のフォレンジック調査を将来のセキュリティ上のミスを防止するための指針とみなします。

5

リソースおよびツールの利用可用性と使用方法を文書化する

署名プロセスと開発者およびエンジニアが署名に使用するツールを監査します。可能な場合、署名を高速化し、サプライチェーン攻撃を減らすために自動化を実装することを検討します。

6

署名および署名コンポーネント（外部および内部）の使用法と手法を定める

日常の開発プロセスと署名コードがどのように適合するかを考えます。鍵の発行、管理、使用、保管、および役割。鍵を共有してはならないことをチームメンバーに徹底します。

7

管理者、マネージャー、開発者、エンジニアからフィードバックを募る

ポリシーの公開および導入の前に、ポリシーを指針として使用する人に確認します。フィードバックを取り込み、ポリシーが各関係者の日常のソフトウェア開発要件に沿うようにします。

8

定期的レビューを含む教育およびトレーニングプログラムを開発する

セキュリティの強固さを保つには、導入時にソフトウェア署名ポリシーと使用方法に関するトレーニングを行い、定期的にチームメンバーが署名手順をレビューする必要があります。

9

ミッションとポリシーガイドを定期的にレビューし、必要に応じて更新、改訂する

組織の目標が変わったり、プロジェクトが変化したり、チームが拡大、縮小、増加することがあります。開発者とエンジニアからのフィードバックを含むポリシーのレビューを定期的に実施することで、変化がポリシーに取り込まれ、手順が最新の状況に合った明確で有用なものであり続けます。

10

情報共有と共同作業を通して、自社のセキュリティおよびサプライチェーンの他社のセキュリティを強化できる機会を探る

1 つの組織の署名機能が成熟することで、ソフトウェア界全体のセキュリティを向上できます。ベストプラクティスの実装に関するアイデアを共有することは、組織、サプライチェーン、エンドユーザーを保護するのに役立ちます。

DigiCert® Secure Software Manager は、CI/CD パイプラインで自動化可能なエンドツーエンドの継続的署名を提供します。追跡、監査、強力かつシンプルな一元的鍵管理を実現します。DigiCert Secure Software Manager は、デジサートのフレキシブルなアーキテクチャと API により、御社の環境を含む任意の開発環境と簡単に統合できます。

コードおよびソフトウェア署名の自動化の詳細については、
当社までお気軽にお問い合わせください
jpn-info-pki@digicert.com

デジサートと信頼の OS

デジサートのルーツは、コネクテッド化社会のセキュリティをもっと簡単にしたいと考え、ひたむきに探求したことから始まります。デジサートは、TLS/SSL、アイデンティティ、サーバー、ネットワーク、鍵、コード、署名、文書、IoT デバイスの管理がより簡単に、より高い信頼性で行える OS を開発しました。サービスとサポートはお客様から業界最高の 5 つ星の高い評価をいただいており、Fortune 500 企業の 88%、世界の銀行トップ 100 のうち 97 行、世界的 e コマース企業全体の 81% の信頼を得ています。PKI は毎日利用されています。より効果的に PKI が動作できるオペレーティングシステムが必要ではないでしょうか。