eBook

TLSのベストプラクティス

digicert®

digicert®

ベストプラクティスで最善なビジネスのスタートを切る

単なる証明書の失効が手間の掛かる面倒な作業になることがあります。残念ながら昨年1年間で企業の60%が、重要なビジネスアプリケーションに影響が出るような証明書関連の障害を経験しています。このeBookは、証明書管理のベストプラクティスに準拠するための詳細なフレームワークを分かりやすくご紹介しています。ビジネスへの影響を緩和することにご活用ください。

目次

識別

発行されたすべての証明書を把握する

各証明書のインストール場所を特定する

すべての証明書とドメインの所有者を明確にする

WebサーバのOSとアプリケーションのバージョンを認識する

Webサーバの暗号スイートとSSLバージョンを特定する

改善

脆弱な暗号鍵、暗号スイート、ハッシュを削除する

ワイルドカード証明書の発行と配布を管理する

適切な証明書タイプを実装する

すべてのベンダーデフォルト証明書を管理する

すべてのWebサービスに必ず最新のパッチをインストールする

保護

発行と更新のプロセスを標準化/自動化する

すべての証明書のインストールと更新をタイミング良く行う

証明書の更新時に秘密鍵を再利用しない

証明書と秘密鍵を安全な方法でインストールする

利用停止の際に証明書の削除/取り消しを行う

監視

ネットワークで新しいシステムと変更を検出する

CT（Certificate Transparency）ログで不正な証明書をチェックする

承認されていない証明書申請をCAAで防止する

発行されたすべての証明書を把握する

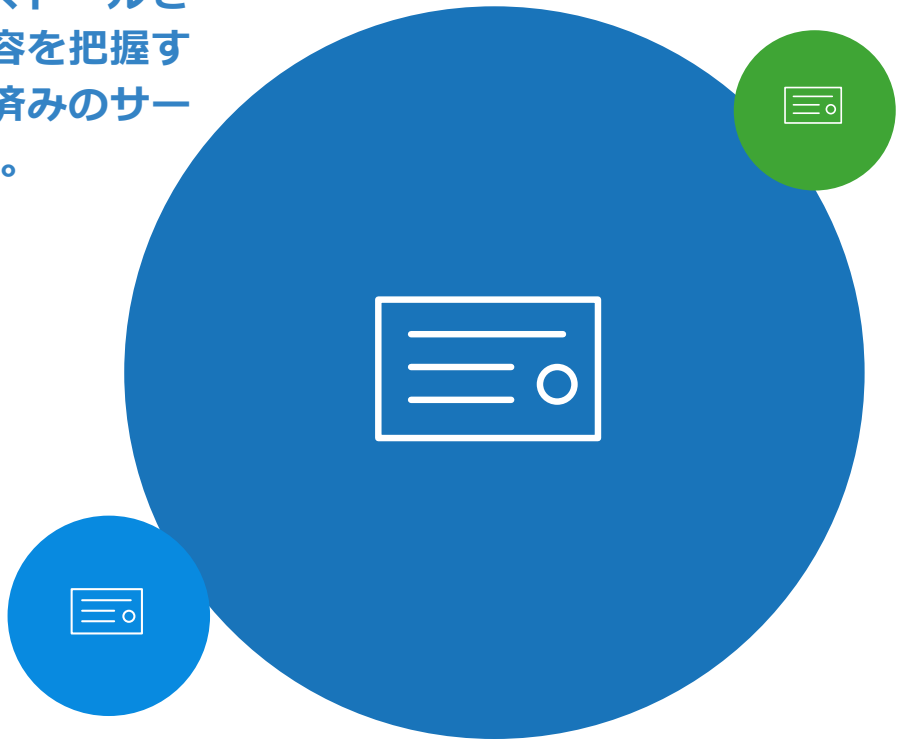
まずは利用中の証明書の全容を把握することが必要不可欠となります。これには、証明書の所有者、場所、ドメイン、OSとアプリケーションのバージョン、暗号スイート、TLSバージョンの認識が含まれます。

利用中の証明書を網羅したリストが無ければ、有効期限切れの証明書、脆弱な暗号鍵、ハッシュなどのセキュリティリスクに目を向けることはできません。リストには証明書情報の詳細が記載されている必要があります。このリストには、すべての認証局から発行された証明書のタイプ（DV、OV、EVなど）発行者に関して確認されている問題、鍵長、アルゴリズム、有効期限、証明書のその他の要素を含めます。

認証局から発行された証明書のリストを取得することから始めると良いでしょう。しかし、すべてを取得できたかどうかはどうすれば分かるでしょう。独自の認証局やSSLサーバ証明書を持つネットワークデバイスは確認できますか？SSLサーバ証明書を検出するには、ネットワークスキャナを使用するのが最適な方法です。ほとんどの組織は、オンライン上に気づいていなかった証明書が多数存在していることに驚きます。

各証明書のインストール場所を特定する

証明書を発行しただけでは、正常にインストールされ、正しい場所にあるとは限りません。全容を把握する証明書のリストを準備するときは、確認済みのサーバの場所をリストに追記する必要があります。



これは手動で追跡できることもありますが、大規模な組織では、定期的な証明書検出スキャンを使用して、証明書が適切な場所にインストールされていることを検証する方が有効です。不正な証明書がインストールされている場合、暗号化されたトラフィックが知らないうちにネットワークから送出される可能性があります。

すべての証明書とドメインの所有者を明確にする

他に誰が証明書を購入しますか？
その人は更新も行いますか？
管理体制に違いはありますか？

証明書の所有者が退職した場合、その証明書は期限切れになり、大きな損害を引き起こす可能性があります。そのため、各証明書の所有者を指定し、所有権の更新と譲渡のプロセスを確立しておくことが不可欠です。認証局が公開されたドメイン向けに証明書を発行するためには、ドメイン所有権も検証することになります。

WEBサーバのオペレーティングシステムとアプリケーションのバージョンを認識する

証明書リストには、Windows、Linuxなどのオペレーティングシステムやapacheなどのアプリケーションの詳細も含める必要があります。

OpenSSLなどの特定のバージョンを狙った攻撃（Heartbleedなど）に脆弱になる可能性があるため、これは重要です。

WEBサーバの暗号スイートとSSLバージョンを特定する

最後に、証明書のリストにはWebサーバ暗号スイートとSSLバージョンを含める必要があります。

これらの項目は、通常、Webサーバで設定されています。SSLに特化した攻撃の多くは、旧バージョンのSSL（例えば、SSL 3.0に対するPOODLE）や、安全でない暗号スイート（例えば、RSA暗号化に対するROBOT攻撃）に焦点を合わせています。

暗号スイート：セキュアなSSL/TLSネットワーク接続を支援するためにWebサーバに構成されているアルゴリズムの一式

まとめ

SSL/TLSサーバ証明書リストには次の情報が必要です。

発行された証明書
- 証明書タイプ
- 鍵長
- アルゴリズム
- 有効期限

証明書の場所

証明書の所有者

Webサーバ構成
- OSバージョン
- アプリケーションバージョン
- TLSバージョン
- 暗号スイート

脆弱な暗号鍵、暗号スイート、ハッシュを削除する

TLS/SSLサーバ証明書リストから未解決の問題が明らかになることがあります。そのときが問題の改善に取りかかるチャンスです。

証明書には、攻撃を受けやすい公開鍵と署名が含まれています。2048ビット未満の鍵長や、MD5やSH A-1のような古いハッシュアルゴリズムを使用している証明書は、公開用Webサーバではすでに利用できなくなっています。しかし、内部Webサイトには存在している可能性があります。その場合は、アップグレードが不可欠です。

脆弱な暗号鍵やハッシュを持つ証明書の認識より重要なことは、Webサーバでサポートされるバージョンと暗号スイートを確認することです。

次のバージョンは、古くて脆弱であるため無効にする必要があります。

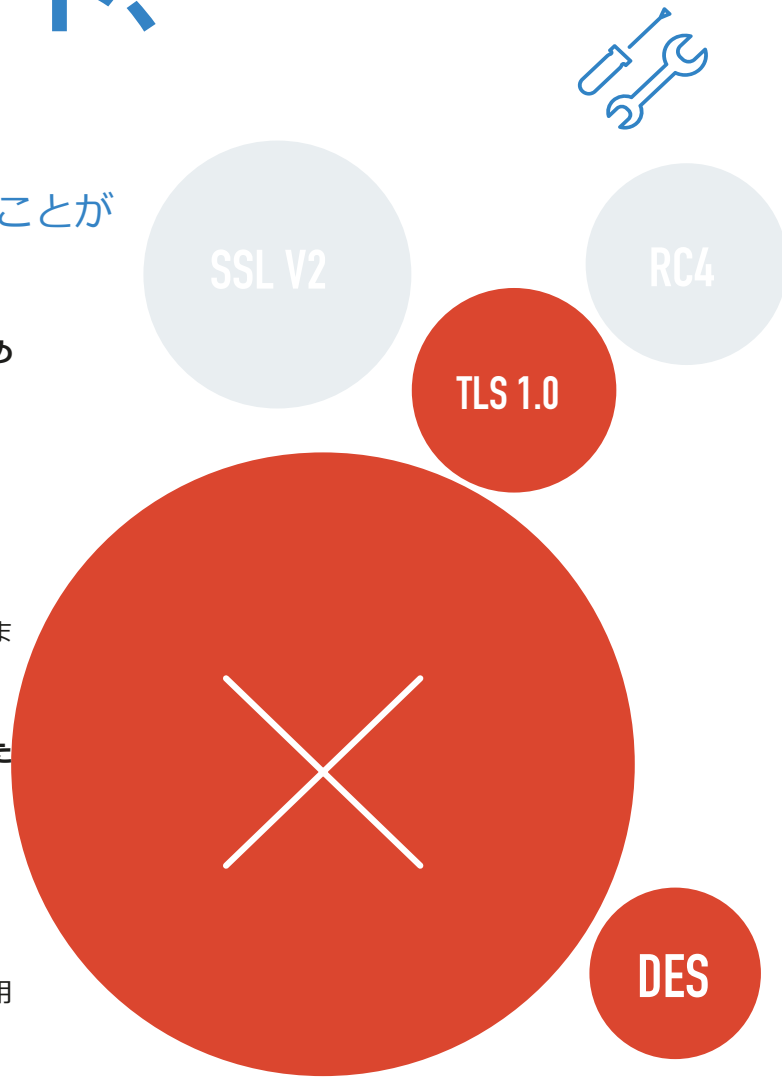
- SSL v2
- SSL v3
- TLS 1.0
- TLS 1.1

代わりに、TLS 1.2とTLS 1.3を有効にします。

次の暗号スイートは、古くて脆弱であるため無効にする必要があります。

- DES
- 3DES
- RC4

代わりに、AESのような最新の暗号を使用します。



ワイルドカード証明書の発行と配布を管理する

ワイルドカード証明書には、一定の条件を満たすことで1つのドメイン名で複数のホストに対応できるという大きなメリットがあります。

ただし、ワイルドカード証明書に関して注意が必要ないくつかの問題点があります。1つには、ワイルドカード証明書の秘密鍵が盗まれた場合、攻撃者はそのドメインスペース内のどのシステムにでもなりすますことができます。例えば、盗まれたワイルドカード鍵はDNSポイズニング攻撃や、ネットワーク内に不正な無線アクセスポイントを作るために使用されています。

もう1つの問題点は、ワイルドカード証明書の鍵が漏洩した場合、全サーバにインストール済みの、証明書のコピー全てを破棄し、再発行する必要があります。コピー数が多くなれば、それだけ問題も大きくなります。完全に文書化されていない場合、すべてのコピーが置き換えられたことを確認できない可能性もあります。

これを防ぐための最善の方法は、ワイルドカード証明書のコピーにそれぞれ異なる秘密鍵を使用して発行することです。こうすれば、1つのワイルドカードが漏洩した場合、すべてのコピーを取り消す必要はありません。

前述の問題により、EV（Extended Validation）ではワイルドカード証明書を使用できません。ただし、十分に管理され、文書化されたプロセスがあれば、ワイルドカード証明書を使用することに何の問題もありません。

適切な証明書タイプを実装する

作成されるSSLサーバ証明書はすべて同等ではありません。

プライベートSSLサーバ証明書は内部システムに使用できますが、プライベートルート証明書はユーザーに正しく配布、設定される必要があります。パブリックサイトを保護する場合は、OV証明書かEV証明書を推奨します。機密情報を処理するサイトには、DV証明書は推奨しません。

EV（Extended Validation）
- ドメイン利用権
- 厳格な組織審査
- 最も高い水準の信頼性

企業認証（OV）
- ドメイン利用権
- 有効な企業の実在性
- 高い信頼性

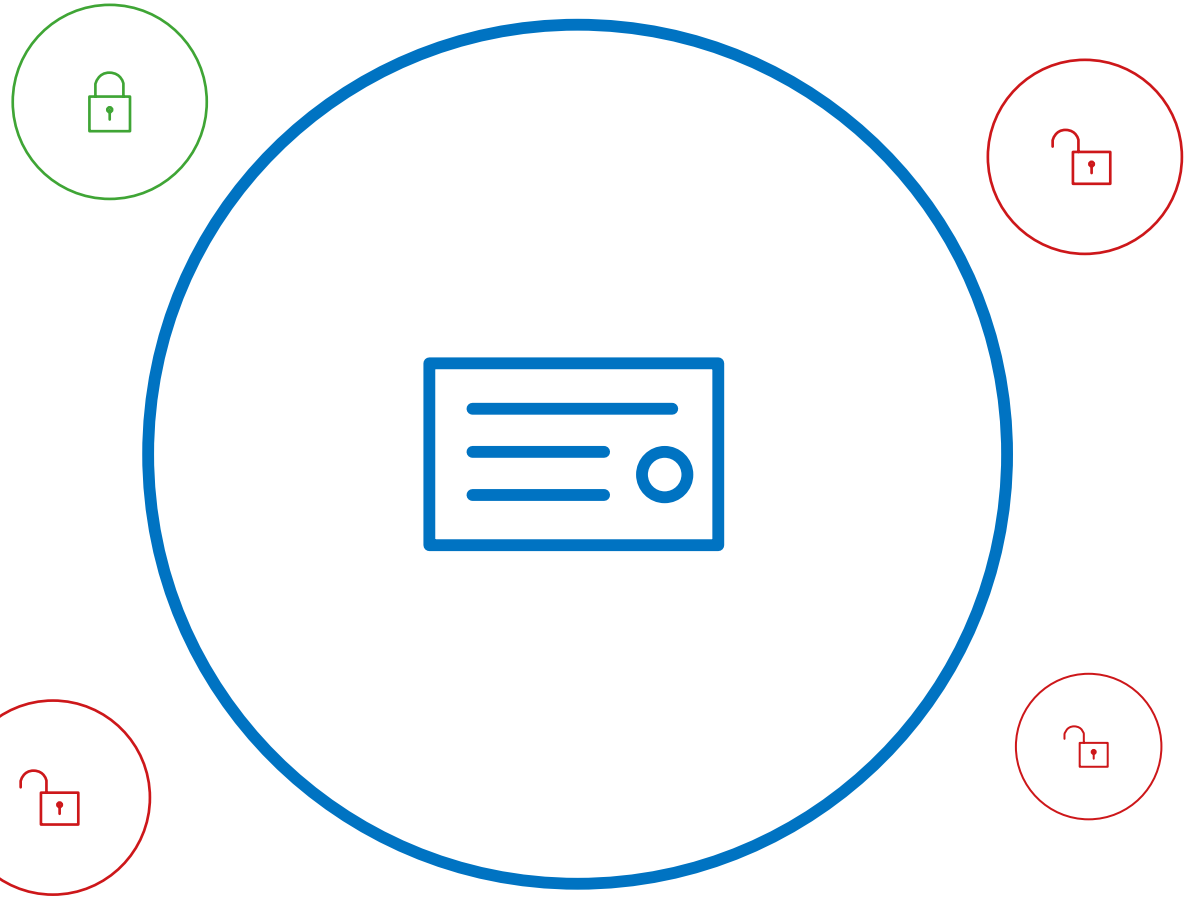
ドメイン認証（DV）
- ドメイン利用権
- 低い信頼性

プライベートSSLサーバ証明書
- プライベートルート証明書のユーザーへの配布、設定が必須

自己署名SSLサーバ証明書
- 信頼性なし

すべてのベンダーデフォルト証明書を管理する

ベンダー証明書は使いやすさが優先され、必ずしもセキュリティ重視ではありません。



問題は、これらのベンダーがこのタイプの証明書を公開ネットワークでの利用を想定していなかったことにあります。通常、ベンダー証明書は、自己署名され、期限切れになるか、脆弱な暗号鍵を使用していて、ブラウザから信頼されません。ほとんどの企業は、何千というベンダー証明書を所有していることに気づいていません。このような証明書は削除し、信頼できる証明書（最低でもプライベートSSLサーバ証明書）に交換する必要があります。このプロセスを効率化するには、ACMEプロトコルを含む最新の自動化ツールを使用して、交換とインストールを行います。

すべてのWEBサービスに必ず最新のパッチをインストールする

Webの最も破滅的な攻撃を回避するには、オペレーティングシステムにパッチを適用することが重要です。

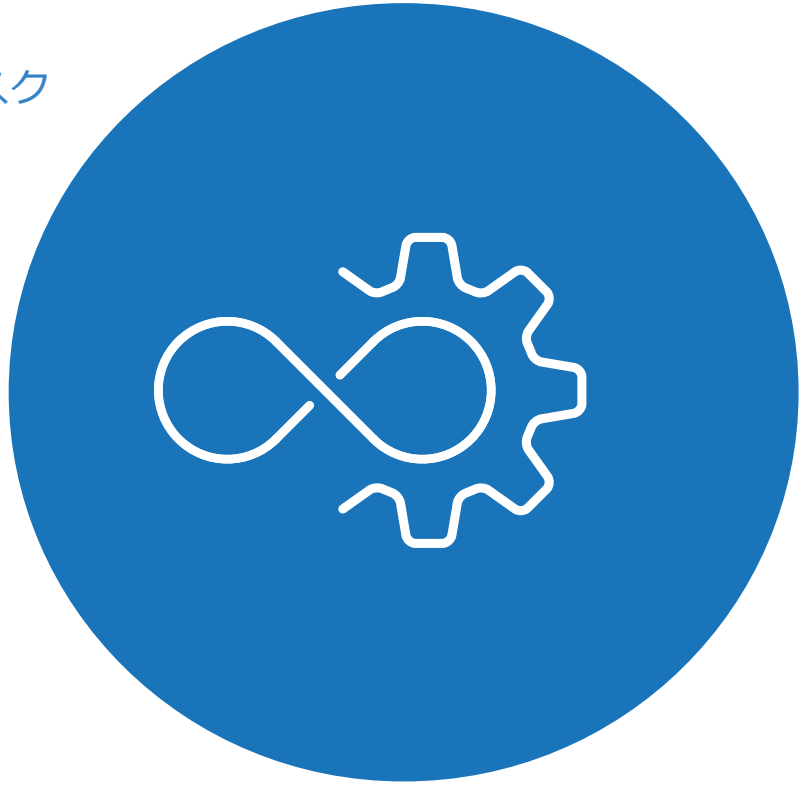
これは、オペレーティングシステムだけでなく、Webサーバにも当てはまります。

まとめ

- 可能な限り、脆弱な暗号鍵とハッシュを取り除く
- SSL v2、v3、TLS 1.0、1.1を無効にする
- TLS 1.2、1.3を有効にする
- TLS 1.2から脆弱な暗号スイートを無効にする
- ワイルドカード証明書を管理する
- 必ず適切な証明書タイプが実装されるようにする
- ベンダー証明書を置き換える
- Webサーバに必ず最新のパッチを適用する

発行と更新のプロセスを標準化/自動化する

あらゆる問題を特定して改善したら、将来のリスクを緩和するためにポリシーと手順を設定します。



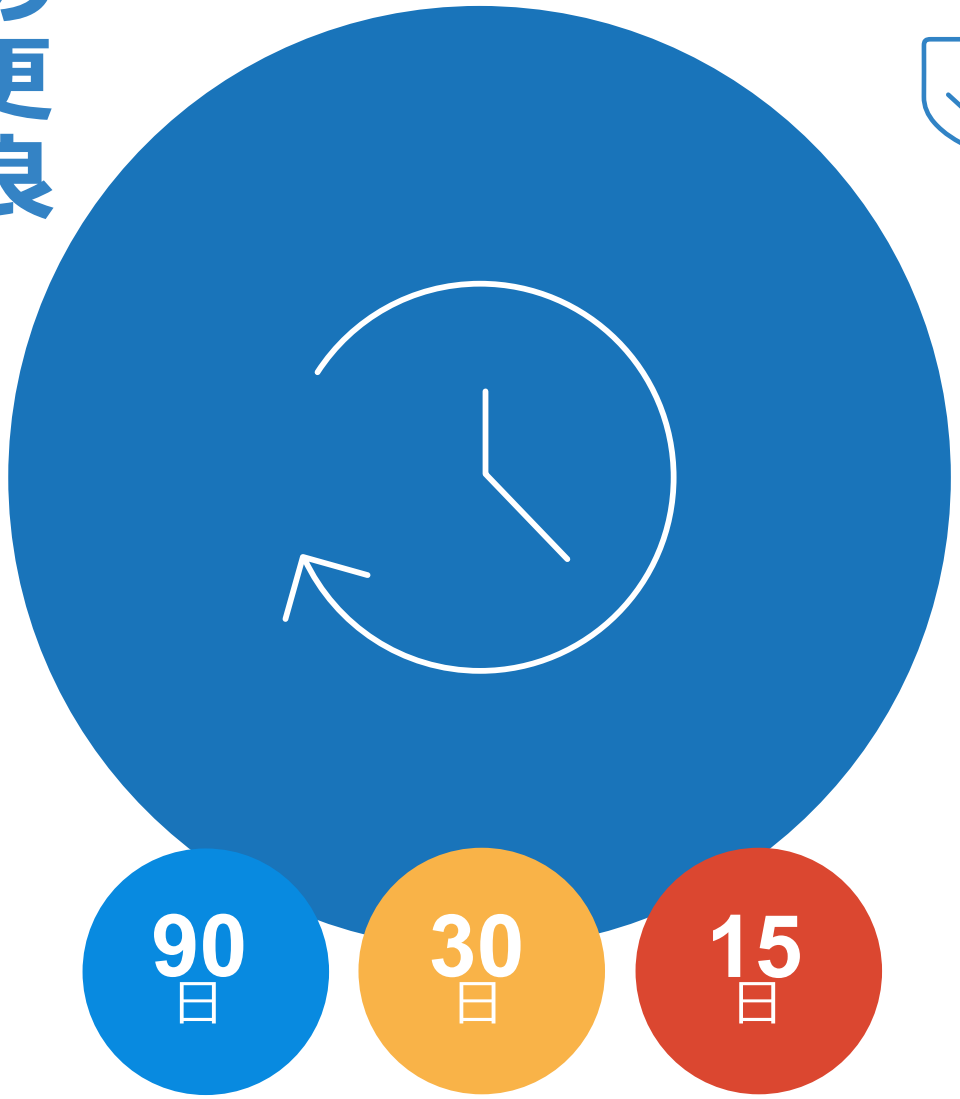
証明書の発行と更新の標準的なプロセスを作成すると、負荷の分散、ユーザーエラーの回避、SSLプロセスの自動化導入に役立ちます。例えば、事実上どのクライアント/サーバタイプを使用していても、DigiCert® CertCentralではACMEプロトコルの実装を自動化できます。

すべての証明書のインストールと更新をタイミング良く行う

組織によっては、様々な時間の制約を受けて仕事をすることになります。

証明書は、有効期限の少なくとも15日前までに更新し、テストを行い、何か問題が発生した場合には前の証明書にロールバックできる時間を残しておくことをお勧めします。変更管理プロセスが長い場合は、標準で30日が適切だと思われます。

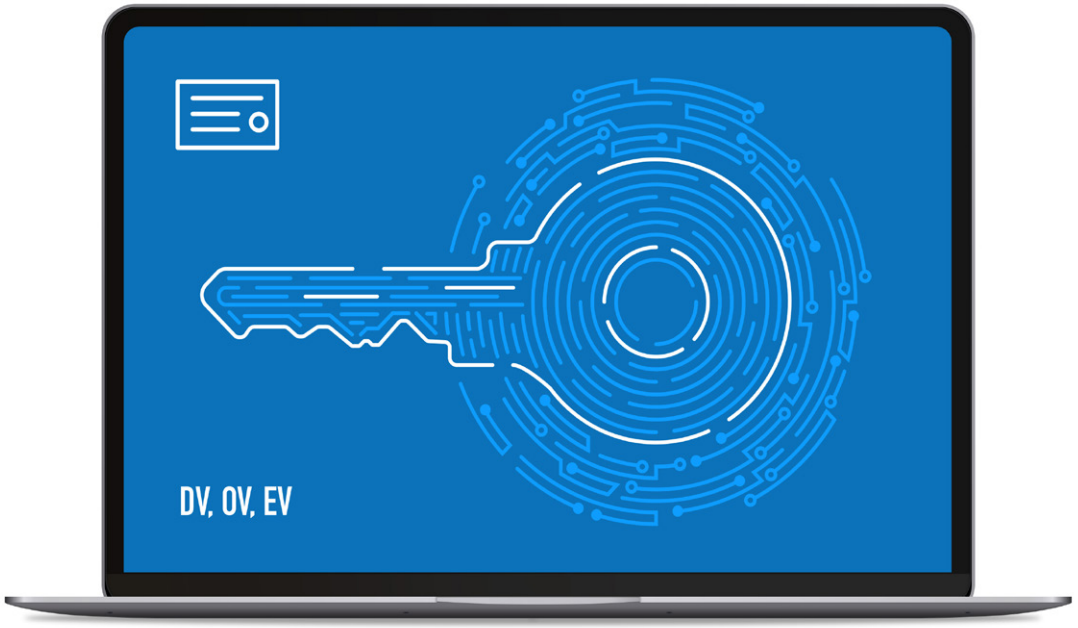
使用しているシステムに関係なく、期限切れになる証明書に関して、ユーザーに警告通知を送信する必要があります。有効期限切れの前に、システムが自動的に一定間隔で（例えば、90日、60日、30日、15日など）ユーザーに通知します。



証明書の更新時に秘密鍵を再利用しない

秘密鍵を再利用することで、鍵が漏洩するリスクが増します。

ベストプラクティスとしては、必ず新しい鍵ペアを作成します。同様に、CSRを再利用しないようにします。これは、秘密鍵が自動的に再利用されるからです。



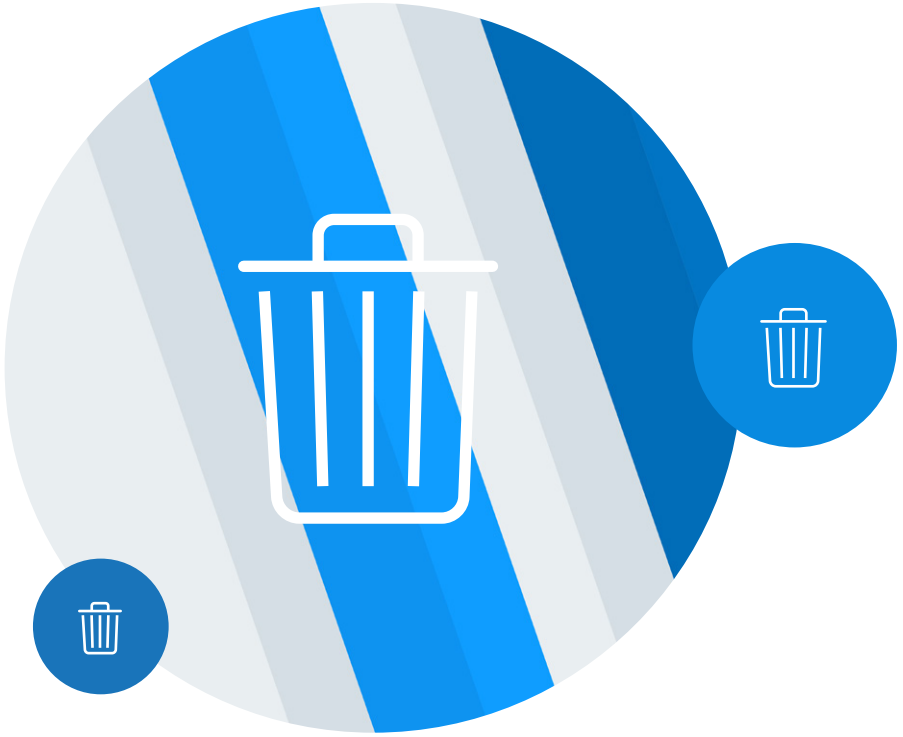
証明書と秘密鍵を安全な方法でインストールする

ほとんどの組織は、安全な方法で秘密鍵を作成し、保存することに長けていません。

- 安全で信頼できるコンピュータで秘密鍵を作成する
- 絶対に必要なときのみ秘密鍵へのアクセス権を付与する
- 所有者が退職したときは、必ず新しい秘密鍵を生成する
- 証明書と秘密鍵の配布には、暗号化した電子メールを使用する
- 利用している電子メールシステムがメールを自動的に削除し、破棄できることを確認する
- これらのシステムへのアクセスには、二要素認証を必須にする
- 秘密鍵のエクスポートや移動を行うために、プロセスを文書化する

利用停止の際に証明書の削除/取り消しを行う

寿命を迎えるシステムの変更管理と取り消しプロセスの一貫として。



まとめ

- 発行と更新のプロセスを標準化/自動化する
- 証明書の更新とインストールをタイミング良く行う
- 証明書と秘密鍵を安全な方法でインストールする
- 秘密鍵は再利用しない
- 廃止プロセスで証明書の削除/取り消しを行う

ネットワークで新しいシステムと変更を検出する

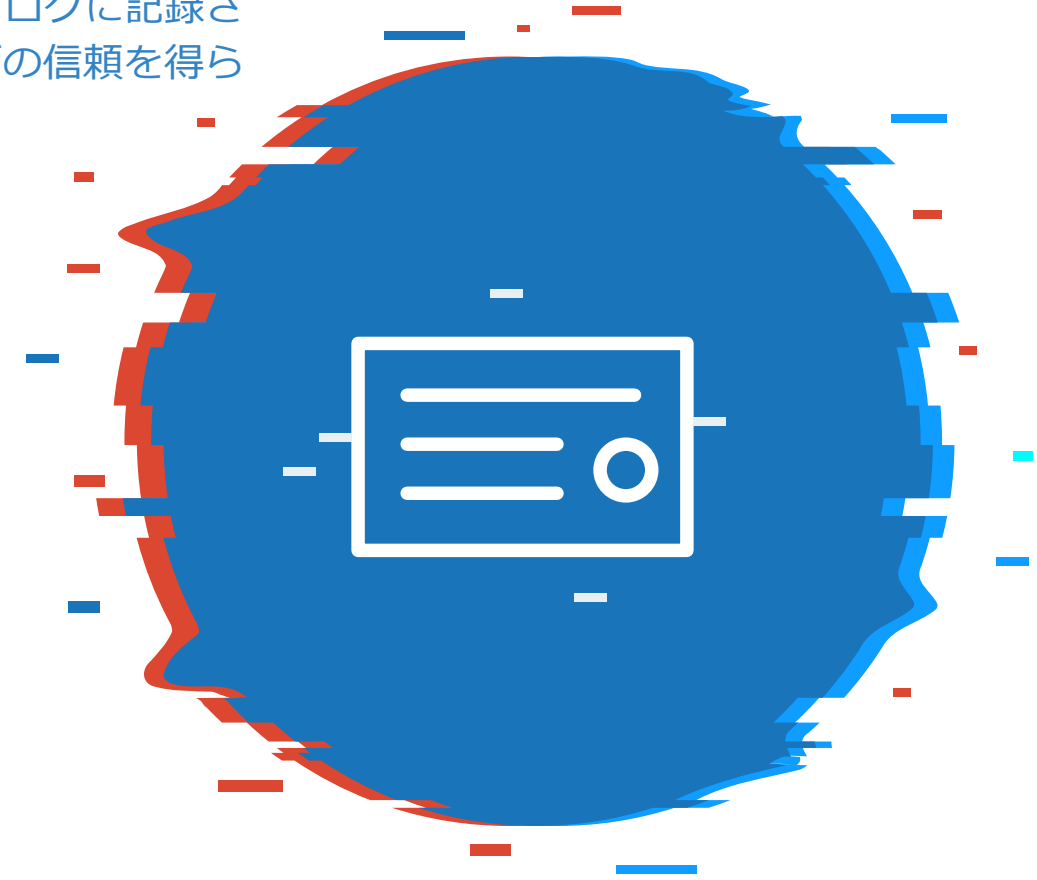
SSLサーバ証明書を手作業で管理する代わりに、発生する可能性のあるリスクの定期的なチェックを行います。

すべてのネットワークは動的で、常に変化しています。そのため、新しいシステムや変更を絶えず監視する必要があります。これを行う最良の方法は、ネットワークスキャンツールを使用することです。これらのツールは、SSLのセキュリティ上の問題、証明書の有効期限、ネットワークのその他の変更を検出します。

CTログで不正な証明書をチェックする

パブリックCT（Certificate Transparency）ログに記録されていないパブリック証明書は、ブラウザの信頼を得られません。

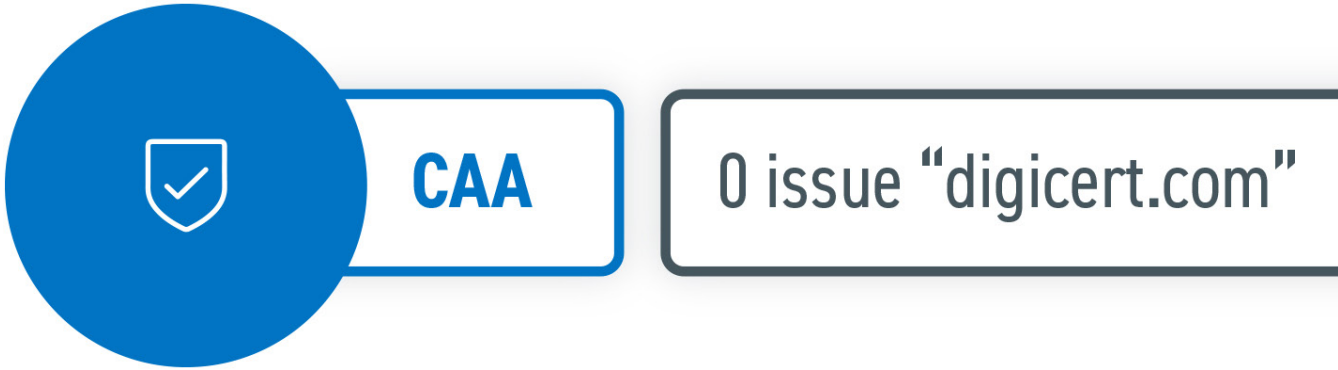
CTモニタを使用して不正な証明書を検出すると信用情報のように、不正な証明書を迅速に識別して修正することができます。



承認されていない証明書申請をCAAで防止する

Certificate Authority Authorization（CAA）は、特定のドメインに対して証明書の発行を許可されている認証局（CA）を指定するために使用されるDNSレコードです。

2017年に、CA/ブラウザフォーラムはBallot 187を発表しました。これは、すべての認証局にCAA DNSレコードをチェックし、対象ドメインに関して見つかったすべてのエントリーに準拠することを求めています。この目的は、ドメイン所有者がそのドメインに対して証明書の発行を許可されている認証局を宣言できるようにすることにあります。CAAは、承認されていない認証局からの証明書がリクエストされた場合に通知を受け取る方法も提供しています。



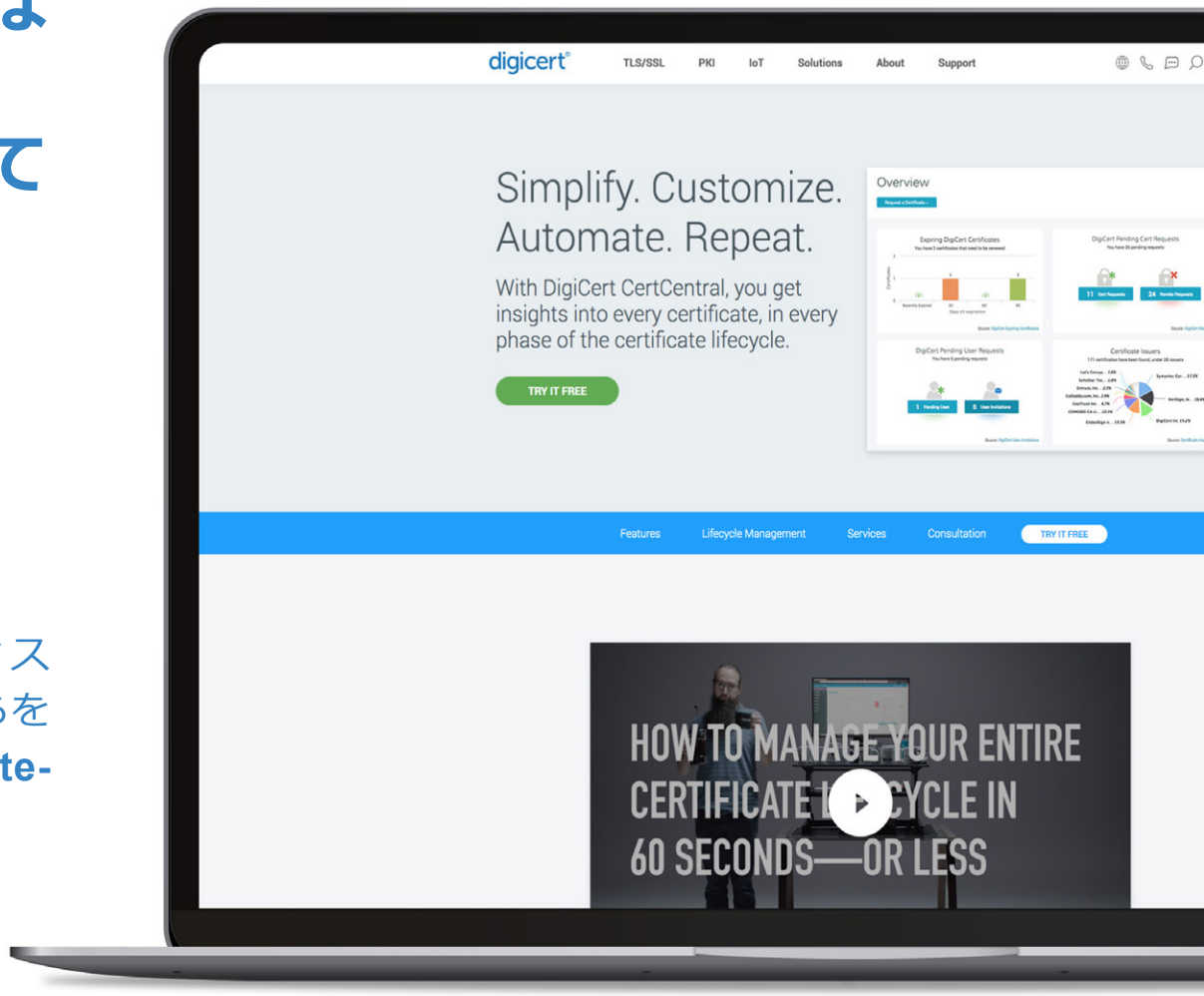
まとめ

何をすべきかが分かったので、次はそれを実行するための最も簡単で最も迅速な方法です。

DigiCert® CertCentralは、証明書エコシステム全体の把握、改善、保護、監視だけでなく、カスタマイズや自動化の機能も提供します。

- ネットワークで新しいシステムと変更を検出する
- CTログで不正な証明書を監視する
- 承認されていない証明書リクエストをCAAで検出/防止する

証明書管理ツールの利用は一般的です。同じプラットフォームで全てを管理するのは特別です。



DigiCert® CertCentralでベストプラクティスを簡単に導入する方法については、こちらをご覧ください：**digicert.com/jp/certificate-management/**