digicert®

TLS Best Practices Guide

Reduce overhead and risk with the five pillars of certificate lifecycle management.

GUIDE

Successful certificate lifecycle management is crucial as the number and types of PKI use cases continue to proliferate, with the number of webpages, devices, systems, servers, and users requiring digital identities and security growing exponentially. As a result, best practices that improve the efficiency and effectiveness of certificate management programs deliver the protection and cryptographic agility that organizations require. This eBook gives you a detailed-but-simple framework for staying ahead of the digital security curve and remaining fully compliant—now and in the future.

Contents

Discover

Manage and report

Notify

Automate

Unify

Discover

Get a complete inventory of you cryptographic assets

Discovery: The foundation of PKI best practices

If you don't have a thorough inventory of your certificate landscape, your organization could be open to security risks you don't even know exist. Using a discovery service to detect and remediate vulnerabilities like rogue or expiring certificates, weak keys and hashes, or outdated server software is one of the most effective means of preventing potential outages and disruptions.

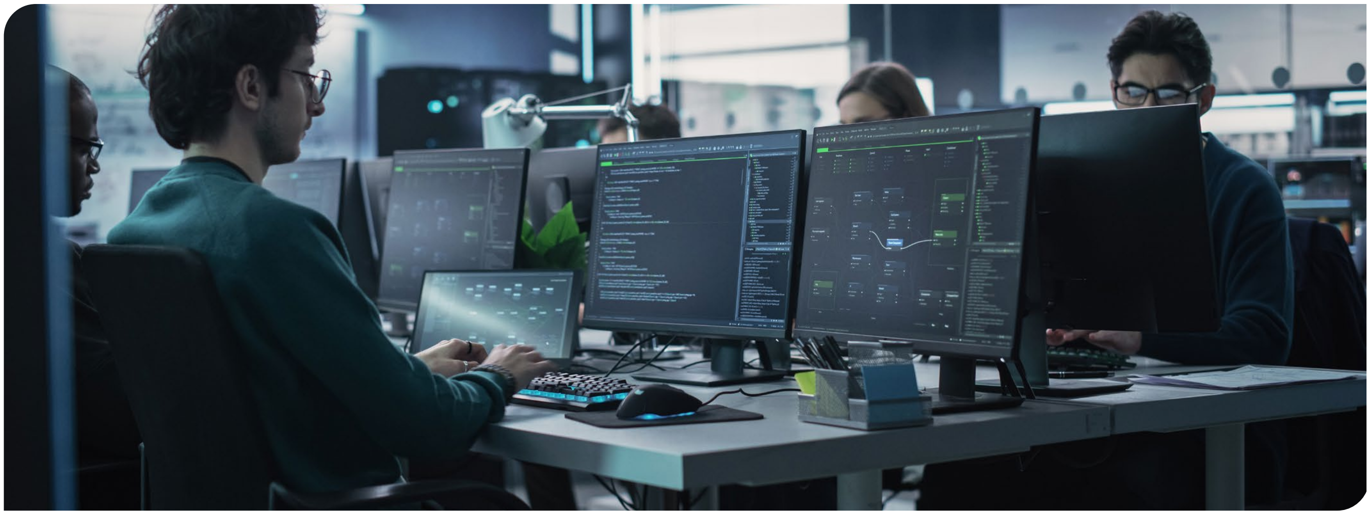
A good place to start is assembling a list of issued certificates from your CAs. But how do you know you've captured everything? What about your internal CAs and any network devices with certificates? Network scans for certificates are a good place to start. Organizations can add to their inventory by also inspecting servers for certificates, keys, user certificates and algorithms; importing private roots and identifying certificates issued off of those roots; and importing data from other discovery tools.

Takeaway:

Discovery services lay the foundation of best practice approaches to PKI management, providing a complete and detailed picture of your organization's cryptographic assets. Scans, inspections, and other discovery methods can be employed at the frequencies that align with the desired risk mitigation strategy, allowing for continuous detection and remediation of vulnerabilities.

Build a unified view of your inventory:

- Network scans
- Server and file inspection
- Private root discovery
- Import from third party tools



Identify exposure to known exploits

Safeguard against system-targeted attacks.

Your inventory information should also include details of the operating system such as Windows or Linux, and applications such as Apache. Each system should be checked to make sure they are all updated to the most recent version to make sure they are not vulnerable to exploits.

This is important because your organization could be vulnerable to critical exploits like Heartbleed, POODLE (SSLv3), FREAK, LogJam or DROWN. You should also assess security factors surrounding your web server operating systems and certificates.

Takeaway:

By identifying the versions of servers, load balancers, application frameworks, cloud infrastructure, databases and other IT infrastructure, you can proactively address exploits and vulnerabilities.

APACHE Windows 11

Scan cipher suites and TLS versions for vulnerabilities

Review cipher suites and TLS/SSL versions.

These items are typically configured on your web servers. Many TLS/SSL-specific attacks focus on older versions of SSL (e.g., the POODLE attack on SSL 3.0) or insecure cipher suites (e.g., the ROBOT attack on RSA encryption). We recommend using the most up to date versions of TLS including TLS 1.2 and 1.3.

What is a cipher suite? .

A set of algorithms configured on a web server that helps to secure TLS/SSL network connections.

Manage and Report

Protect your private keys

Reusing a key is like reusing a password—and saving time isn't worth increasing risk.

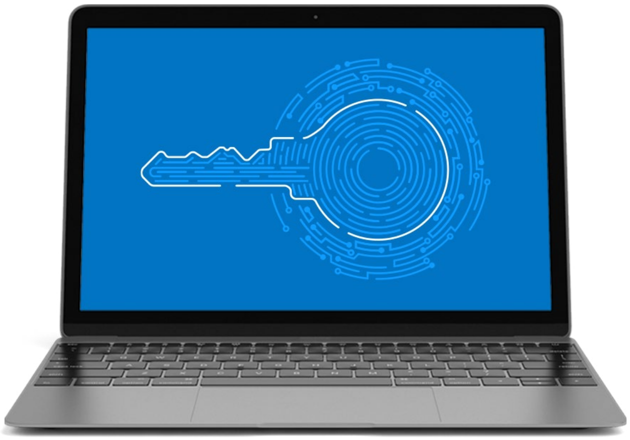
The private key is a separate file that's used in the encryption/decryption of data sent between your server and the connecting clients. It's created by the certificate owner during the Certificate Signing Request (CSR). The Certificate Authority (CA) providing your certificate does not create or have your private key. In fact, no one outside of your administrators should ever be given access to this material. As a best practice, make sure you create a new key-pair with every certificate. Likewise, you should never reuse a CSR as this will automatically reuse the private key.

In addition:

- Check for weak keys
- Check for known compromised keys
- For high security, use a vault, token, or HSM for secure key storage

Takeaway:

It can be tempting to reuse CSRs to save time, but this duplicates keys and multiplies risk. Automating certificate requests and renewals can significantly reduce the manual effort required to generate CSRs and provision certificates



Prioritize remediation

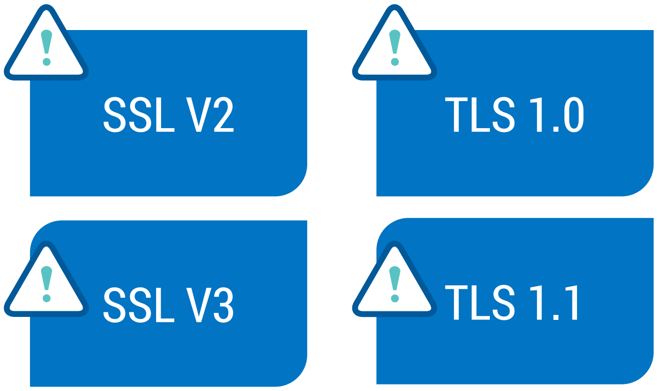
Remediate weak keys, ciphers, hashes, and any outdated assets identified in your inventory.

Certificates contain public keys and signatures which could be vulnerable to attacks. Certificates with key lengths less than 2048-bits, or that use older hashing algorithms like MD5 or SHA-1, are no longer permitted on public web servers. However, you might find these on your internal websites. If so, it's vital that you upgrade them immediately.

Even more important than identifying certificates with weak keys or hashes is reviewing TLS/SSL versions and cipher suites supported on your web servers. You should always enable the most up-to-date versions of TLS including TLS 1.2 and TLS 1.3. For cipher suites, use the most modern ciphers like AES or review this list to see obsolete and retired cipher suites.

Outdated and vulnerable TLS/SSL versions:

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



Control issued and distributed wildcard certificates

Wildcard certificates simplify certificate issuance but introduce other security concerns.

While wildcard certificates provide a clear benefit to administrators, there are security risks to securing multiple domains with the same private key. For example, if the private key is lost or stolen, it acts as a single point of compromise for every server the certificate is issued to. Using the same private key across the network or sharing it with different departments could lead to the key being misplaced or stolen, thus requiring all related certificates to be replaced. And if the wildcard certificate is revoked for any reason, the private key will need to be updated on all servers that use that certificate. All these updates will have to be done at one time to avoid disruption of data moving across the network.

This same process applies to wildcard certificates upon renewal. Although organizations may initially use wildcards to save time and money, managing their renewal or having to unexpectedly replace a wildcard certificate can lead to significant work.

Takeaway:

It's important to protect your private key when using a wildcard certificate. To minimize the risk of sharing a single private key across wildcard uses, you could instead use a separate private key for each copy of your wildcard certificate and have a secure system in place to protect your private keys. We also recommend using automation on every server the wildcard certificate is installed on to save time, reduce human error, and minimize misplacement of keys.

Deploy appropriate TLS certificate types

Choose the level of assurance that supports your business.

If you are securing a public website that collects user information in forms, or logins and passwords, we recommend a high-assurance certificate to protect your brand by preventing imitation of your brand and company's website.

The certificate that provides the highest brand and identity assurance is an Extended Validation (EV) TLS certificate. EV certificates are often used by governments, global businesses, banks, and financial services organizations. EV certificates Have the most demanding validation process involving 16 steps to verify details such as the certificate requestor's contact info, job title, and employment, as well as a contact blocklist check, domain fraud check, organization registration number, jurisdiction, and registered agent check.

The second highest assurance certificate is an Organization Validated (OV) TLS certificate. OVs validate the domain owner and all of the contact information of your organization, including a confirmation of your organization's physical address, a phone call to authenticate the certificate request, a fraud and blocklist check, and malware checks.

Lastly, Domain Validated, or DV TLS certificates provide the lowest level of identity assurance and are never recommended for sites transacting sensitive information as they could easily be impersonated. As for private TLS certificates, they are often used for internal systems, but the private root must be successfully extended to users to remain secure.

Certificate use cases:

Extended validation (EV)

- Banks and financial services
- Fortune 500 companies
- Global 2000 companies
- E-commerce
- Compliance (e.g., HIPPA, PCI)\

Organization validation (OV)

- Log-in Screen
- Business sites
- Compliance (e.g., HIPPA, PCI)

Domain validation (DV)

- Blogs
- Personal websites
- Any site that doesn't conduct transactions or gather information

Control all vendor-provided certificates

Vendor certificates are designed for ease of use, but not necessarily security.

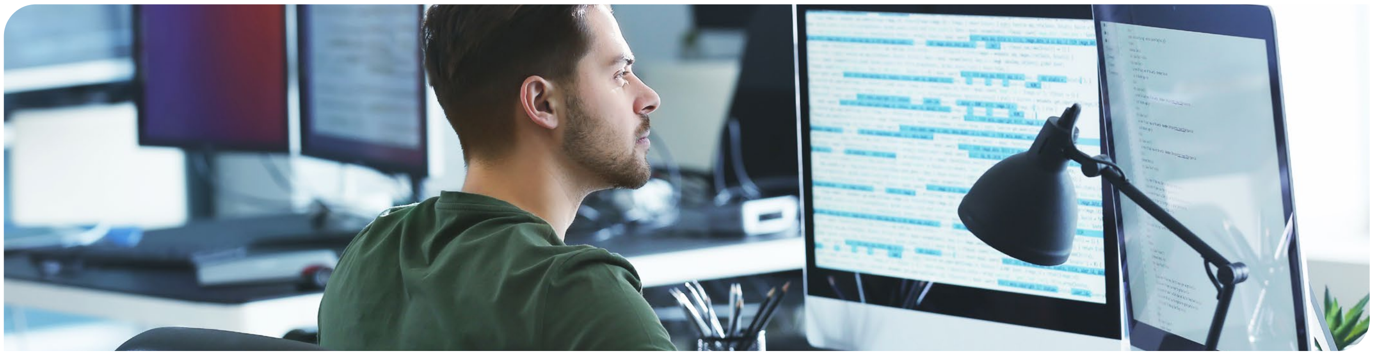
Vender certificates are certificates issued by third party hardware vendors and come preinstalled on their devices. The problem is that these third parties never intended these types of certificates to be put on a production network. Default vendor certificates are typically self-signed, expired or using weak keys, and are therefore not trusted by browsers. Many organizations have thousands of vendor certificates they're not aware of. Each of these certificates should be removed and replaced by a certificate with known trust (at a minimum a private TLS/SSL certificate). To streamline this process, use the latest automation tools like APIs or ACME URL to help you with the replacement and installation.

Ensure all system patches are current

Patching systems is important to avoid some of the web's most devastating attacks.

Patches are updates to operating systems, servers, application frameworks, databases, and other softwar and systems that provide security against vulnerabilities in a product. For example, this could apply to Windows and Linux operating systems or to your web servers and load balancers.

These updates help prevent attacks like the Heartbleed Bug which was a vulnerability found in OpenSSL's cryptographic software library. Anyone who used the vulnerable version of the OpenSSL software had a backdoor for attackers to read memory from any of their systems secured by it—allowing attackers to eavesdrop on communications or steal data from the services and users.



Secure access to certificate management systems

Take advantage of common tools like two-factor authentication and single sign-on (SSO).

Two-factor authentication (2FA), or multi-factor authentication, requires the use of more than one security method to authorize logins—typically something you have, and something you know. Protect your certificate management platforms with an extra layer of security to avoid breaches when controlling your diverse inventory of certificates.

Examples of two-factor authentication include:

- Mobile device 2FA authentication
- Authenticator app 2FA
- Token generator 2FA

Integrate with ITSM systems

Certificate management platforms can integrate with other software solutions like ServiceNow to interface seamlessly with IT operations.

For more complex IT environments, integrate your certificate management platform with Information Technology Services Management (ITSM) like ServiceNow to create the certificate lifecycle management approval flows that fit your organization's processes. This will allow personnel to request the TLS certificates they need without giving them direct access to your certificate management systems. Using an ITSM system to manage escalation processes will reduce human error and increase uptime.

Review alerts for needed actions

Leverage dashboards that surface items that need action or review.

Using a centralized dashboard to track the status of items identified during discovery is a key first step, but it's also important to ensure that the data presented is clear, refined and actionable.

Your reporting system should enable you to:

- View your total certificate landscape from a single console
- Build, download, schedule, or integrate detailed reports
- Easily identify specific problems and action items, like rogue certificates, pending expiration dates and compliance issues
- Display intuitive data visualizations and graphs that are easily communicable and present a clear course of action

Notify

Set up notification and escalation hierarchies

Define and automate notification methods and destinations.

Prevent lapsed certificate or security breaches by setting up notification and escalation hierarchies. These hierarchies should specify, for certificate groups:

- When notifications are triggered
- The method by which they are sent (e.g., email, alerts, slack, ITSM)
- The role to which they are delivered
- Escalation thresholds

Also, ensure that your email contacts are up to date with the issuing CA so that you get all necessary renewal or incident notifications.

Set up notification thresholds

Define and automate renewal alerts 90, 60 or 30 days prior to certificate expirations.

Stay ahead of certificate expiration dates by setting up notifications for defined thresholds (e.g., 90, 60, 30 days prior to expiration). We recommend renewing a certificate at least 15 days prior to the expiration date to ensure you have time for testing and verification. If you have a longer change control process, 32 days may be a more appropriate standard.

Alert thresholds should similarly be defined for expired certificates

90 60 30

Monitor certificate transparency logs

Any public certificate not logged in a public Certificate Transparency (CT) log will not be trusted by browsers.

Use a CT log monitoring tool to quickly detect rogue certificates and to identify and remediate misissued certificates. CT logs provide accountability to Certificate Authorities issuing trusted TLS/SSL certificates during the validation process. If someone else issues a certificate against your domain name—whether maliciously or in violation of policy—CT log monitoring detects the error and alerts you immediately.

Set up CAA alerts and prevent unauthorized certificate requests

Certificate Authority Authorization (CAA) is a DNS record used to specify which CAs are allowed to issue certificates for your domain.

In 2017, the CA/Browser Forum introduced Ballot 187 which requires all CAs to check the CAA DNS records and comply with any entries found for the domain in question. The purpose of this is to allow domain owners to declare which CAs are allowed to issue a certificate for their domain. CAA also provides a way to receive notifications in case someone requests a certificate from an unauthorized CA.

Automate

Automate the certificate lifecycle

Streamline certificate renewals, installations and CSR generation.

Certificate automation is an operational necessity for organizations wanting to efficiently manage high volumes of certificates. By automating various manual tasks involved in certificate management, you can minimize human error and reduce the time and resource burden of shorter certificate validity periods.

Automating certificates:

- Reduces overhead involved in certificate issuance, replacement, and renewal
- Prevents downtime and outages by removing human error and preventing misconfiguration
- Reduces IT support desk workload by automating user provisioning
- Speeds remediation by enabling efficient replacement of compromised certificates

Achieve cryptographic agility with certificate automation

Post-quantum cryptography is on the horizon

Organizations can future-proof their operations for any industry or cryptographic change by employing automation tools today. By automating certificate issuance and renewal, organizations will save time in the long run managing, tracking or replacing certificates during a security event. Automation can also aid in updating cryptographic algorithms for certificates more easily.

Automate business processes

Define access rules, workflows, templates, and integrations.

Automation of certificate management extends beyond hands-free TLS certificate provisioning, installation, and renewal. Business process automation streamlines PKI management across your entire certificate and cryptographic inventory. Predefined access rules, automated approval and notification workflows, automated enrollment, secured keys, and integration with enterprise systems are examples of ways in which business process automation can improve security posture and streamline operations.

Utilize APIs for custom integration

Get direct integration between your certificate management platform and your enterprise systems.

Streamline certificate management with your existing business systems, processes, or products by harnessing API integration. Examples of common integrations include integration with ITSM systems to direct certificate activity into change window processes, or with targeted discovery tools to contribute to a unified inventory view.

Unify

Implement root ubiquity across the network

Newer or smaller Certificate Authorities (CA) may not have their roots included in some browser root stores; this is especially an issue for older browsers.

As a best practice, make sure your organization is using certificates from a well-established CA with a high root ubiquity. That means that the certificates are present in the key stores of new and older browsers, which means they are compatible with 99.9% of client platforms and browsers.

In the past, some Certificate Authorities' roots have not been included when a new browser version was released, causing browser error messages for website visitors. This can have a serious effect on sales conversion and reputation for a website owner.

Use CA-Agnostic discovery and import services

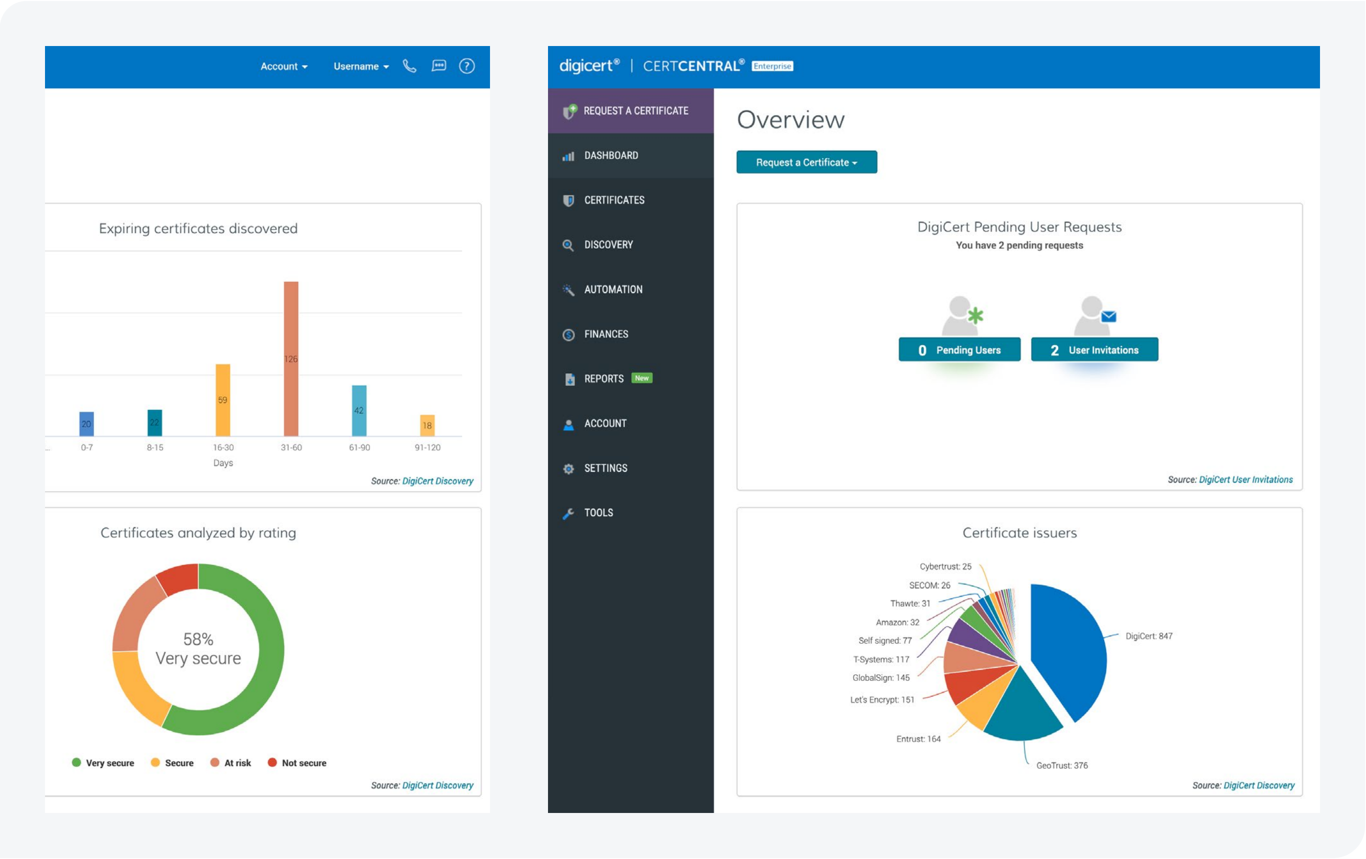
Use a certificate management platform that supports multiple certificates from multiple CAs.

A CA-agnostic platform enables you to track and manage every certificate—no matter the type or issuing CA—within one platform. Look for discovery tools that can import private root certificates for discovery off of those roots or other cryptographic assets. This approach can then deliver a comprehensive, unified view of your entire certificate inventory.

Conclusion

End the cycle of tedious certificate lifecycle management practices.

With DigiCert's certificate lifecycle management solutions, you have all the capabilities you need to implement the five pillars of TLS best practices, including how to discover, manage and report, notify, unify, and—even better—automate your certificate inventory. It's intuitive, comprehensive, and the best way to be proactive about your certificate management program.



Get proactive about your certificate management

Streamline certificate lifecycle management and focus your time on your business. Learn how DigiCert products help you implement every certificate best practice. Email contactus@digicert.com today to discuss your certificate management needs or visit digicert.com/tls-ssl/certcentral-tls-ssl-manager to sign up for a comprehensive demonstration.