Quick Start Guide

Certificate Lifecycle Management and Automation

appviewX

Contents

Executive Summary

Forming the core of PKI, SSL/TLS certificates have served as the Internet's primary means of authentication, encryption, and security for years – they're used universally across business, vertical, and function to this very day. PKI has never been a set-and-forget solution. Without diligent and constant management, certificates are vulnerable to a host of issues like expiry and invalidity which can cripple network security and leave users open to attacks, breaches, and system downtime.

The average certificate count of a business runs into the high thousands. What's more, there are several variables associated with managing one certificate (multiple CAs, expiration dates, levels of encryption, locations in the network, etc.). Both these factors considered, the fact remains that an automated management process (of the certificate lifecycle) has numerous advantages over its manual counterpart.

This contents of this whitepaper are intended for teams responsible for *NetOps, SecOps, PKI,* and *IAM,* as well as *Directors/VPs of* Security *and CISOs/CIOs*. It outlines the primary elements of certificate lifecycle management, the advantages it provides, and the inevitable consequences of not implementing it in a growing certificate ecosystem. We've also defined five concrete steps you can take to quickly implement certificate lifecycle automation in your organization.

Key learning outcomes include:

- Actionable knowledge about the workings of a certificate lifecycle management system
- Increased visibility and control over your network security
- Heightened threat deterrence, response, and recovery capabilities

Note: Click here to skip to the Quick Launch Strategy

Why Automate the Certificate Lifecycle?

The answer is simple. Not automating the lifecycle management process can be disastrous in the long haul. Managing tens of thousands of certificates, keys, and ciphers is tiresome and inefficient when manual methods are leveraged (read: spreadsheets). And the number of endpoints that require PKI-backed protection are increasing by the day, with trends like the Internet-of-Things only snowballing the urgent need for PKI management systems.

Certificate management, by itself, is a complex process. This degree of complexity is elevated when the certificates are spread across multiple geographies, networks, and cloud ecosystems. What's more, when there's no concrete management system in place, lifecycle management becomes an ad-hoc routine that's siloed by departments, ultimately leading to a lack of visibility. In the long haul, this translates to a lack of agility – for instance, organization-wide changes in CA vendors or quick response to depreciation of standards like SHA-1 could take months to safely implement.

The lack of a defined system to manage PKI is simply a synonym for mismanagement. And the effects of mismanagement are often manifold. After all, just one expired certificate paints a target on the entire network for hackers to infiltrate and abuse. Poorly managed certificates often lead one or more of the following repercussions:

- Application downtime or network outages due to unprecedented expiration
- Data breaches caused by poorly secured certificates and compromised keys
- Lawsuits and federal penalties for customer data mishandling
- Compliance or audit failure, and accompanying fines
- Drop in employee productivity due to internal service downtime
- Loss of business and customer trust due to external-facing service downtime

We've covered these devastating consequences in detail in an associated whitepaper, [The Business Impacts of PKI Mismanagement.]

Components of the Certificate Lifecycle

The certificate lifecycle may be defined as the duration of a certificate's existence, from the time it is issued by a CA, to when a client chooses to retire it. There are a handful of well-defined components in this cycle, and each stage has to be carefully vetted and monitored – either by an engineer or by software – to ensure that it fully plays its role in securing an endpoint on a network.

Fig 1.1 : The Certificate Lifecycle

Issue - Certificates are purchased from a vendor (Certificate Authority or CA). There are several classes of certificates available, based on application, location, and lifetime.

Inventory - Details about the certificates are documented – information like validity period, certificate type, position in the chain, and location on the network become critical as it nears expiration.

Provision - The certificate is pushed to the endpoint it was meant to secure. This could be a server, an application, a hardware device, or a network tool like a firewall.

Secure - The private keys (the public key's twin) is documented and stored in an encrypted database or a HSM to prevent theft and misuse.

Monitor - The certificate's status is continually monitored to ensure that it is valid, secure, has no vulnerabilities, and meets compliance standards.

Renew - A certificate that is not renewed past its expiration date becomes invalid, and is not accepted as a valid network object by communicating entities.

Revoke - Once a certificate is no longer necessary, it has to be formally retired from the system to prevent misuse.

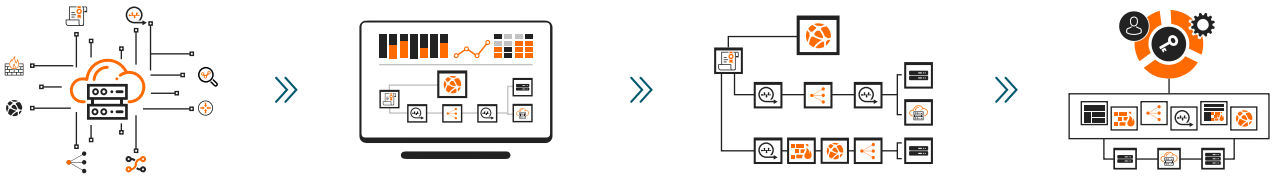
The Core Principle of Certificate Lifecycle Management

Ideally, a certificate lifecycle management system should allow for the administration of all 7 steps in the lifecycle. It should act as a central hub via which an administrator can:

- Gain complete visibility into the network's PKI
- Manipulate every aspect of the PKI from a single interface

The best tools enable the implementation of a structured governance process within the organization. It formalizes and documents every activity in the system – from requesting certificates and CSRs to zero-touch renewals (the ability to renew a certificate without having to use the CA's interface), enforcing the presence of an audit trail. It also allows for dynamic, real-time monitoring of the PKI ecosystem, enabling rapid troubleshooting and issue remediation. The presence of user-definable automation capabilities and custom workflow definition would be a bonus.

In short, fundamental principles of lifecycle management are centralization, minimization of human contact, and bird's-eye visibility into the network infrastructure.



Visibility into PKI infrastructure

Management of certificates and keys

Monitoring via reporting and audit logs

Automation workflows for certificate operations

Quick Launch Strategy : Certificate Lifecycle Management in 5 Steps

Follow these steps for a straightforward implementation. The merits of adopting lifecycle management are vast and instantaneous – decreased expirations and outages, potent threat deterrence capabilities, and vastly improved cryptographic agility are some of the immediate advantages you'll notice and appreciate.

Note: It's alright if you can't locate every tool we talk about in this segment – we'll tell you what to do in the next section.

Step 1: Scan your network

Employ a sweeping tool that runs comprehensive, top-down scans across your entire network to discover every certificate that is present on it, and where each one is located. Ensure that you run these scans at periodic intervals (for instance, once every three weeks) to maintain a healthy inventory devoid of undocumented certificates.

This is a critical first step – often, large organizations have several departments across multiple geographies, with each team requesting certificates based on individual requirements. This leads to many certificates going undocumented, and being present in remote locations on the network, with many of them being labelled 'rogue' (unapproved).

By detecting every certificate and determining where it's located (on a firewall, attached to a browser, and so on), you'll ensure that you're not missing anything when you attempt bulk renewals or revocations. You can also ensure that each one is well-managed, since even one vulnerable certificate serves as a weak link which can be exploited by malicious actors.

However, some certificate detection tools and CA-provided software are limited to only detecting the certificates issued by that particular CA, or certificates of certain types. **This is important:** Ensure that you're using a **universal scanning tool** that discovers both CA-issued and in-house certificates across all networks, cloud environments, and hardware locations.

Step 2: Build an inventory

The obvious next step is to consolidate the results of the scan in a database, in order to draw meaningful insights from it. Ideally, the scanning tool should automatically inventorize its findings in a structured database with complete details of every certificate discovered – name, expiration date, encryption strength, key strength, the endpoint's network, location, and so on. Once this is done, group certificates based on CA or location to ease the process of renewals and revocations when the time comes.

Your inventory should also give you a lateral view into your infrastructure – in other words, the chain of trust should be easily verifiable. Ensure that you store certificates in the format: **Root Certificate - Intermediate A - Intermediate B ... Intermediate n - End User Certificate**. This method of documentation greatly simplifies the implementation of business workflows to each certificate in the chain – either all at once, or individually.

Step 3: Ensure monitoring and reporting

A single round of discovery is simply not sufficient in an ever-changing certificate environment. Update your inventory details every time you run a network scan. Ensure that you keep tabs on the status of all your PKI components at all times. While this might seem impossible with the sheer number of certificates and devices involved, creating a simple dashboard makes the process a whole lot simpler.

Create a business-intelligence style control panel with graphical reports on statistics like expiry dates of certificate groups, CA-based classification of lifetimes, and organization-wide compliance statuses. Make sure that these reports are dynamic, and reflect changes in real time – this can only be achieved if the system you are using for report generation is linked with your certificate environment. You should also strive for customizability of your reporting capabilities, to cater to the needs of every member in the PKI management value chain.

Step 4: Enforce an ownership hierarchy and audit trail

The underlying principle of this step is to structure the certificate enrolment process and ensure that the only people who are able to make permanent changes to the certificate infrastructure are those with the clearance to do so. This way, there will be no undocumented or unapproved certificates in circulation. You need to ensure that the following boxes are checked:

1. Network-level changes **must** require approval from authorized personnel
2. Every change made to the system **must** be documented

To achieve this, you must first establish well-defined roles within your PKI management team (or across your entire organization). Each level in the hierarchy should be a part of an approval chain that allows the delegation of ownership. There are several ways to implement role-based access, either by using an internal tool, or by integrating your management system with external directory systems like LDAP, RADIUS, and ActiveDirectory.

Next, create a management process which provides varying levels of privileges according to individual roles. For example, a Super Administrator will be able to request, enrol, and push certificates to their endpoints, while an Administrator will only be able to request certificates – he/she would have to require the Super Admin's approval before they can take any further actions on it.

Finally, create an audit trail system, which logs actions taken by every person in the hierarchy, along with a timestamp. Audit logs are immensely useful for determining the cause of certificate-related issues, and for detecting policy violations.

Step 5: Automate certificate operations

Renewals, Revocations, and Deployment : these are three primary tenets of certificate lifecycle management. With thousands of certificates on file, how can you ensure the diligent execution of all three activities without interruptions or delays? The answer is automation.

First, purchase a certificate lifecycle automation tool which ties into your network. The best automation tools allow users to define entire automation workflows that can run without human intervention.

Automation tools simplify certificate operations by allowing administrators to carry out the following activities from a single interface (i.e without having to use each CA's interface to renew or revoke the certificates they've issued) . First, The tool will refer from the PKI inventory to determine when expiry dates are approaching, and will automatically notify you via email a certain number of days prior. You can also automate renewals and simply set up an approval process to keep tabs on the cycle. A tool that integrates with all major Certificate Authorities would accelerate and simplify the entire process.

If the certificate has reached the end of its life, or if your organization requires a PKI overhaul, you can also decommission a certificate permanently at the click of a button.

Deployment is a bit more complex that revocation and renewal, since there are a myriad of endpoint types that certificates are deployed to. Ensure that the tool supports native and API-based integrations with various service vendors so that you can set up a zero-touch push-to-device policy right from within the tool's interface.

A Unified Solution : AppViewX CERT+

In the previous section, we mentioned the need for tools to scan your network, create an inventory, automatically perform renewals, and more. Of course, you'll need every one of those tools to successfully implement certificate lifecycle automation. But using separate tools for each step in the process defeats the purpose of lifecycle management – it introduces another level of clutter and complexity, which in turn generates more possibilities of risk and human error.

What you need is a tool that can help you follow all the above steps: from discovery, renewal, revocation, and enrolment, to reporting. We would recommend the use of AppViewX CERT+, our flagship certificate lifecycle management tool. It's an end-to-end management tool which features seamless integrations with not only CAs, but also network device providers, mobile device management platforms, security vendors, and more. It enables you to execute every step of the quick launch strategy for certificate lifecycle automation – and will also empower you to build upon your configuration as your requirements inevitably change.

Visit our website for in-depth information about the product.

We also provide assistance to organizations that need help in implementing certificate lifecycle management. Sign up for a demo by clicking the link below, and one of our Solution Experts will personally analyze your PKI infrastructure, give you personalized action items on the steps to be taken, and even help you set it up.

30-Minute Live Solution Demo REGISTER NOW

About AppViewX

AppViewX is revolutionizing the way NetOps and SecOps teams deliver services to Enterprise IT. The AppViewX Platform is a modular, low-code software application that enables the automation and orchestration of network infrastructure using an intuitive, context-aware, visual workflow. It quickly and easily translates business requirements into automation workflows that improve agility, enforce compliance, eliminate errors, and reduce cost. AppViewX is headquartered in Seattle with offices in the U.S., U.K., and India. To know more, visit **www.appviewx.com**.

AppViewX Inc.,
500 Yale Avenue North, Suite 100, Seattle, WA 98109

info@appviewx.com
www.appviewx.com

+1 (206) 207-7541
+44 (0) 203-514-2226