



Certiligent

# Automated TLS/SSL Certificate Renewal

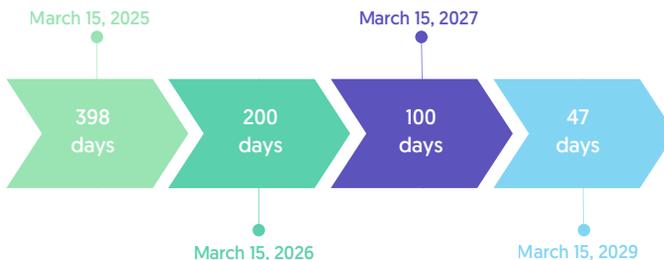
Security without manual work



## Why certificate automation is no longer optional?

Digital certificates are a core part of how all systems should establish trust. However, the way they are issued and maintained has fundamentally changed.

The **CA/Browser Forum (CA/B Forum)**, supported by all major browser vendors (Apple, Google, Microsoft, Mozilla), has defined a **mandatory, accelerated reduction of public TLS certificate lifetimes**. What was once an annual operational task is becoming a **monthly, continuous process**.



This increases renewal frequency by **8-12x** compared to traditional yearly cycles. Manual processes, scripts, and fragmented tools do not scale under these conditions and significantly increase the risk of outages, security gaps, and audit findings.

**Automation is no longer an optimization. It is a prerequisite for compliance, availability, and security.**

**Certiligent is designed specifically to address this new reality.**

## The Concept

Certiligent is a platform for **centralized, automated management of TLS digital certificates** across heterogeneous IT environments. It enables organizations to issue, renew, revoke, and monitor certificates through their entire lifecycle, while aligning operations with the accelerating requirements dictated by the CA/B Forum and modern security practices.

Certiligent is designed for **financial institutions, regulated enterprises, and large organizations** that require strict security controls, auditability, scalability, and flexibility in deployment.



## Key Benefits

### Outage Prevention

Automated renewal eliminates the risk of service disruptions caused by expired certificates.

### Operational Efficiency

Reduces manual work, ad-hoc scripting, and human error across certificate operations.

### Scalability

Supports short-lived certificate policies and high-frequency renewals without increasing operational overhead.

### Centralized Visibility

Provides a single, authoritative view of certificates across servers.

### CA Independence

Integrates with multiple public and private Certification Authorities (DigiCert, Sectigo, Keyfactor, GoDaddy, Let's Encrypt, ZeroSSL and more) and works with standard ACME clients such as Certbot, acme.sh, Kubernetes cert-manager, OpenShift Certman.

### Enterprise security & auditability

Built to meet the security, control, and traceability requirements of regulated environments.



## Key Features

### Flexible integration with CAs

Supports both ACME protocol and REST API integration, allowing seamless integration with major CAs based on your environments and compliance needs.

### Intuitive Management Dashboard

Provides real-time visibility into certificate status, issuance history, and renewal workflows, all from a user-friendly interface

### Automated Alerts and Notifications

Proactively informs administrators about failed renewals and upcoming expirations to prevent outages.

### Scalable and Secure Design

Designed to manage thousands of certificates with high security and reliability for enterprise needs.

### Multi-Tenant Architecture

Designed for all types of organizations, enabling secure segregation of tenants for their clients.



## Use Cases by Industry



### Banking & Financial Services

Automate certificate renewals for online banking portals, APIs, mobile apps, open banking / PSD2 interfaces, payment gateways (POS backend, card payment systems), and critical internal systems such as core banking platforms and VPNs to ensure regulatory compliance and prevent service outages.



### Regulated organizations

Manage certificates across highly controlled and audit-sensitive environments, including internal applications, external services, and secure network components, with full traceability, policy enforcement, and centralized governance.



### Large Enterprises

Manage thousands of certificates across web applications, email servers, VPNs, microservices with centralized visibility, automation, and operational control.