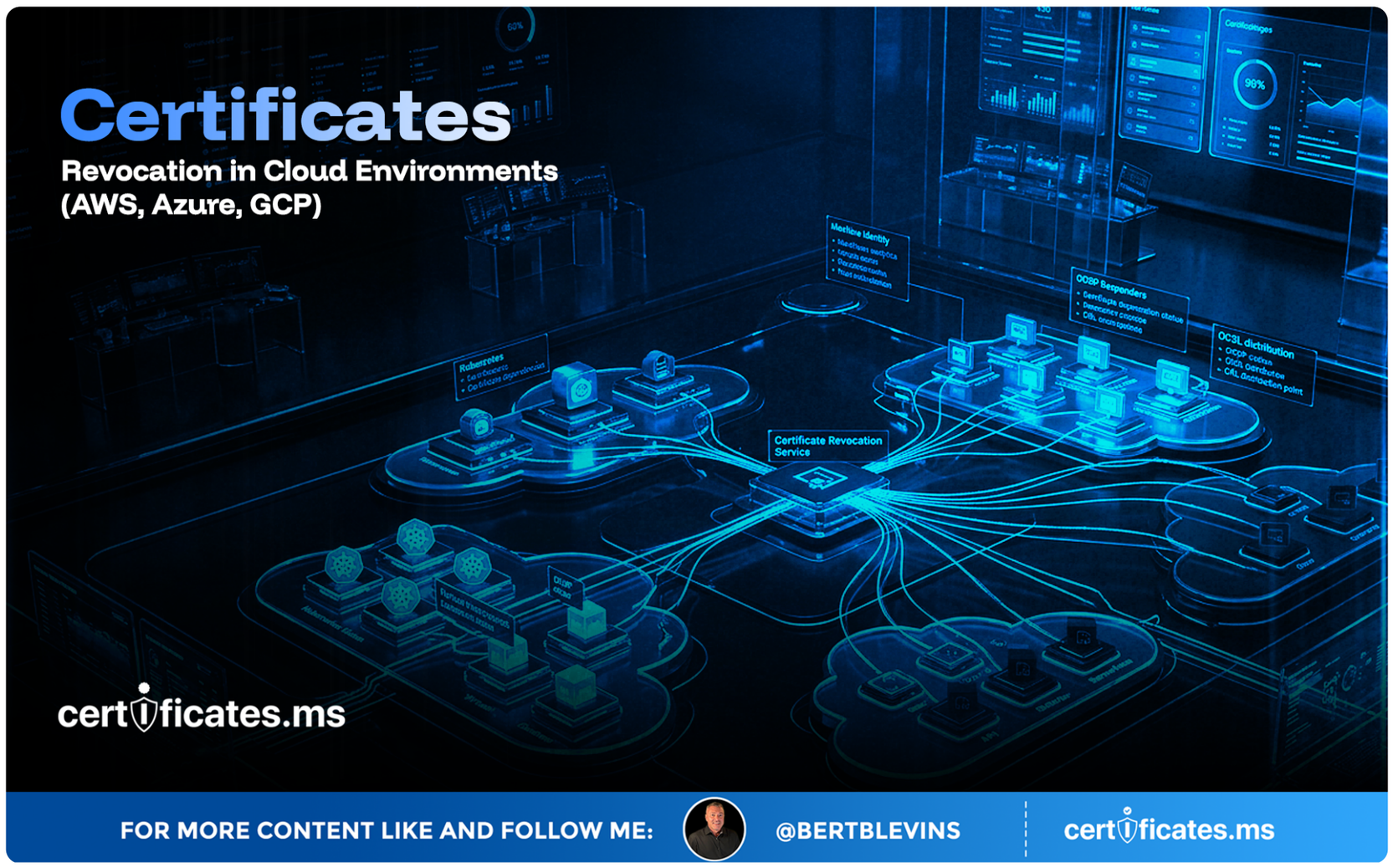
Certificate Revocation in Cloud Environments
(AWS, Azure, GCP)



Revocation, covered conceptually elsewhere in this series, takes on distinct practical characteristics inside each major cloud provider's ecosystem, since AWS, Azure, and Google Cloud each handle certificate issuance and revocation through their own managed services with their own specific mechanics and limitations. This article covers what revocation actually looks like across these three platforms.

AWS Certificate Manager: Revocation Through Reissuance

AWS Certificate Manager, the service most commonly used for certificates attached to AWS resources like load balancers and CloudFront distributions, does not expose a traditional manual revocation function the way managing your own CA would; instead, if a certificate needs to be invalidated, the standard approach is deleting the certificate resource and issuing a new one, relying on ACM's automated renewal and the underlying certificate's own validity period rather than a separate explicit revocation request to the issuing CA. This model works well within ACM's managed context but means organizations need to think about incident response for a compromised ACM-issued certificate slightly differently than they would for a self-managed certificate with direct revocation control.

Azure Key Vault and Certificate Revocation

Azure Key Vault manages certificates alongside its broader secrets and key management functionality, and revocation handling depends on which CA actually issued the certificate, since Key Vault can integrate with several partner CAs alongside supporting self-signed and Azure-internal certificates. For certificates issued through an integrated public CA partner, revocation typically follows that CA's standard process, initiated through Key Vault's management interface or API, while internal or self-signed certificates managed purely within Key Vault can be directly revoked or deleted as part of the vault's own access and lifecycle controls.

Google Cloud Certificate Authority Service

Google Cloud's Certificate Authority Service allows organizations to run a fully managed private CA within Google Cloud, and it provides direct, explicit revocation functionality consistent with standard CRL and OCSP mechanisms discussed elsewhere in this series, since organizations using this service are effectively running their own CA rather than relying purely on automated reissuance the way ACM's model works. This gives organizations using Google's CA Service more granular, traditional revocation control compared to the reissuance-centric model AWS Certificate Manager uses for its own issued certificates.

Multi-Cloud Revocation Complexity

Organizations running infrastructure across more than one of these cloud providers face a genuine coordination challenge: a security incident requiring rapid certificate revocation across a multi-cloud environment needs a response plan that accounts for each provider's specific revocation or reissuance mechanics, rather than assuming a single unified revocation process works identically everywhere. Building and testing a documented, cloud-specific incident response runbook in advance, rather than discovering these platform differences for the first time during an actual active incident, is considerably safer given how much

Automating Response Across Cloud Providers

Organizations serious about incident readiness increasingly build automation that can trigger a coordinated response across multiple cloud providers simultaneously, using each provider's respective API to revoke or reissue certificates as part of a single, orchestrated incident response workflow rather than requiring manual intervention separately within each cloud console during a time-sensitive security event. This kind of cross-cloud automation considerably reduces response time during an actual compromise, when speed matters most.

Cloud Certificate Revocation for AI Workloads

AI workloads running across multi-cloud environments, increasingly common as organizations distribute AI infrastructure across providers for redundancy or specific service availability, inherit all of the multi-cloud revocation complexity described above, with the added consideration that a compromised AI agent's credentials may need to be revoked considerably faster than a traditional service's certificate, given how quickly an autonomous agent could take unauthorized action if its compromised identity remains valid even briefly after detection.

The Countdown Is Already Running: 200 Days, 100 Days, 47 Days

Every certificate conversation in 2026 eventually arrives at the same clock, and it is worth closing on it here. The CA/Browser Forum's Ballot SC-081v3 is not a proposal under discussion; it is an approved, already-in-motion schedule. Maximum public TLS certificate lifetimes fall from 398 days to 200 days on March 15, 2026. They fall again to 100 days on March 15, 2027. By March 15, 2029, they drop to just 47 days, with domain validation itself needing to be re-proven roughly every 10 days.

01

Translate that into operational terms and the picture gets stark quickly. An organization currently renewing certificates a few times a year will be handling renewal events on the order of every couple of weeks by the end of this countdown, across every endpoint it operates. Manual tracking, calendar reminders, and a spreadsheet somebody checks once a month will not survive contact with that cadence. What has always been an occasional chore is becoming a continuous, automated operation, whether an organization plans for it or not.

02

Regardless of which cloud provider issues a given certificate, that certificate is still subject to the shrinking public lifetime schedule below wherever it interoperates with public trust, making each provider's specific automation and revocation mechanics something every cloud-based certificate strategy needs to account for well before renewal cycles compress toward every 47 days.

03

The 200-day, 100-day, and 47-day milestones are not distant hypotheticals; the first has already arrived. Organizations that build the automation loop now, generating keys, vaulting them securely, brokering issuance across Certificate Authorities through APIs, and rebinding certificates to live endpoints without manual intervention, will meet each deadline without disruption. Organizations that wait will be rebuilding their certificate operations under deadline pressure, with far less room for error and far less time to get it right. The countdown is the call to action. The only real decision left is whether to automate on your own schedule, or on the CA/Browser Forum's.