When and Why You Should Revoke a Certificate

Revocation is the emergency brake of the certificate world, and like any emergency brake, its value depends entirely on people knowing when to actually pull it. Too hesitant, and a compromised certificate stays trusted far longer than it should. Too trigger-happy, and legitimate services get disrupted unnecessarily. This article covers the situations that genuinely call for revocation, and the reasoning behind each one.

Private Key Compromise: The Clearest Case

If a certificate's private key has been exposed, whether through a server breach, an accidentally committed code repository, a misconfigured storage bucket, or any other route, that certificate needs to be revoked immediately, without exception. A compromised private key means anyone who obtained it can impersonate the certificate's legitimate owner for as long as the certificate remains trusted. There is no meaningful gray area here; suspicion of key compromise alone is generally sufficient grounds to revoke, since the cost of an unnecessary revocation is far lower than the cost of leaving a genuinely compromised key trusted.

Certificate Issued in Error

Sometimes a CA issues a certificate incorrectly, perhaps to the wrong organization due to a validation failure, with an incorrect Subject Alternative Name, or in violation of the CA's own issuance policy. When a misissuance is discovered, whether by the CA itself, the domain owner, or a third party monitoring Certificate Transparency logs, revocation is the standard remedy, and CA/Browser Forum baseline requirements typically mandate revocation within a defined window once misissuance is confirmed.

Change in Organizational Status

Certificates tied to an organization's identity, particularly Organization Validated and Extended Validation certificates, may need revocation if the underlying facts that justified issuance change: a company is acquired or dissolves, a domain changes ownership, or a business relationship that the certificate implicitly represented comes to an end. Continuing to present a certificate reflecting outdated organizational facts can mislead relying parties who trust that information.

Departing Employees With Client Certificates

Organizations that issue client certificates for VPN access, email signing, or internal authentication need a reliable process for revoking those certificates when an employee leaves, changes roles into a position that no longer requires that access, or is terminated under circumstances warranting immediate access removal. This is one of the more commonly overlooked revocation triggers, since it lives at the intersection of HR processes and IT security, and gaps here are a genuine, recurring source of lingering unauthorized access.

CA Compromise or Policy Violation

In rarer, more severe cases, an entire intermediate or root CA can be compromised or found to have violated its issuance policies seriously enough that browser and operating system vendors distrust it. When this happens, every certificate that CA issued may need to be revoked or reissued under a different, trusted CA, a large-scale event that underscores why limiting an intermediate's scope and keeping the root offline, as covered elsewhere in this series, matters so much for containing damage when something does go wrong.

Revoking Certificates Issued to AI Agents and Automated Services

A newer but increasingly common revocation trigger involves AI agents and automated service identities. If an agent's credentials are found to have been misused, if the automated pipeline that issued a machine identity is found to be compromised, or if an AI service is being decommissioned or has its permissions substantially reduced, its certificate should be revoked rather than simply left to expire naturally. Because these identities are often issued and retired at high velocity, effective revocation for this category depends heavily on automation, tying certificate revocation directly into the same orchestration systems responsible for provisioning and retiring the underlying workloads.

Revoking Without Overreacting

Not every anomaly warrants revocation. A certificate nearing its natural expiration in a few days, a minor configuration mismatch unrelated to key security, or a routine organizational rebranding that does not affect the underlying legal entity generally do not require emergency revocation and can typically be handled through normal renewal or reissuance instead. Having clear, documented criteria for when revocation is warranted helps organizations respond decisively to genuine incidents without treating every certificate question as a five-alarm fire.

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

As certificates move onto the shorter lifespans described below, the natural expiration window itself becomes a form of built-in risk reduction, but it does not replace the need for prompt, deliberate revocation whenever key compromise or misuse is discovered, since even 47 days is more than enough time for a compromised credential to cause real damage.

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.