Public Certificate Acquisition:
A Step-by-Step Guide for Beginners

Acquiring a public certificate for the first time can feel more intimidating than it needs to be, mostly because the terminology arrives faster than the explanations do. This guide walks through the process in order, from generating a key pair to keeping the resulting certificate alive long after installation, written for someone doing this for the first time rather than for someone who already knows the vocabulary.

01

Generate a Key Pair

Every certificate begins with a cryptographic key pair: a private key that must never leave your control, and a public key that will eventually be published inside the certificate itself. Most server software and command-line tools can generate this pair directly. The private key should be generated on the machine that will actually use the certificate whenever possible, and it should be stored with restricted file permissions and, for anything sensitive, backed by a hardware security module or a secure secrets manager rather than sitting as a plain file on disk.

02

Create the Certificate Signing Request

With the key pair generated, the next step is producing a Certificate Signing Request, which bundles the public key together with identifying details such as the domain name, and in some cases organizational information. The CSR is signed with the private key to prove possession, then submitted to the chosen Certificate Authority. Beginners commonly get this step wrong by requesting a certificate for the wrong domain variant, forgetting to include the www subdomain or additional Subject Alternative Names that the deployment will actually need, so it is worth listing every hostname the certificate needs to cover before generating the CSR.

03

Choose a Certificate Authority and Validation Level

Not every CA or validation level suits every situation. For a personal blog or an internal test environment, a free, automatically issued Domain Validated certificate is generally sufficient. For a business handling customer data or payment information, Organization Validation or a paid CA with stronger warranty and support terms is usually the better call. This decision should weigh the sensitivity of what the certificate protects against the cost and validation overhead involved.

04

Step Four: Complete Domain Validation

The CA needs proof that the requester actually controls the domain named in the CSR. This typically happens one of three ways: adding a specific DNS TXT record, placing a designated file at a known path on the web server, or responding to an email sent to an address associated with the domain's registration. Automated tools using the ACME protocol handle this step without manual intervention, which is strongly recommended even for beginners, since it removes an entire category of possible mistakes.

05

Install and Configure the Certificate

Once issued, the certificate needs to be installed on the server alongside its private key and, importantly, alongside the CA's intermediate certificate, which completes the chain of trust. A common beginner mistake is installing only the end-entity certificate and omitting the intermediate, which causes some browsers and clients to fail validation even though the certificate itself is perfectly valid. After installation, the configuration should be tested using an independent SSL testing tool to confirm the chain is complete and the configuration follows current best practices.

06

Monitor and Automate Renewal

Certificates expire, and an expired certificate breaks trust immediately and often without warning to the operations team. Setting up automated renewal, typically through the same ACME tooling used for issuance, removes the single most common cause of certificate-related outages: someone simply forgetting a renewal date. Monitoring tools should also be configured to send alerts well ahead of expiration as a safety net, even when automation is in place, since automation itself can silently fail.

A Note for Anyone Automating This With AI Tools

A growing number of beginners are now using AI coding assistants to help write the scripts that handle certificate generation, CSR creation, and renewal automation. This can genuinely speed up the process, particularly for someone unfamiliar with command-line tooling, but it comes with one important caution: private keys should never be pasted into an AI chat interface, included in a prompt, or logged anywhere an AI tool's context might be retained. Use AI assistance to help write and debug the automation scripts themselves, and let those scripts handle the actual key material locally, so the sensitive part of the process never leaves your own environment.

Putting It All Together

Acquiring a public certificate is a linear process once the terminology is demystified: generate a key pair, create a CSR, choose the right CA and validation level, complete domain validation, install the certificate correctly with its full chain, and automate renewal so the whole thing keeps working without ongoing manual attention. Get comfortable with these six steps and the rest of certificate management, including the more advanced topics covered elsewhere in this series, becomes far easier to build on.

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

The six-step process above is manageable by hand today; by the time certificates carry a 47-day lifespan, step six, automating renewal, stops being good advice for beginners and becomes the only workable option for anyone managing certificates at all.

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.