

CertBinder: TLS-Zertifikate erneuern und überall richtig binden

Ein TLS-Zertifikat zu erneuern ist leicht. Schwer ist, das neue Zertifikat überall einzubinden, wo das alte lief: in IIS, Exchange, ADFS, SharePoint, RDS, SQL Server und auf den Domänencontrollern. CertBinder findet heraus, welches Zertifikat wo steckt, erneuert es und bindet es atomar mit automatischem Rollback in jeden Dienst ein.

Das Problem: Das Erneuern ist nicht die Arbeit, das Binden schon

Eine Szene, die jeder Admin kennt: Das Wildcard-Zertifikat läuft in zwei Wochen ab. Das neue ist schnell besorgt, eingespielt auf dem Webserver, Haken dran. Drei Tage später ruft der Außendienst an, weil der Zugriff auf die Remote-Desktop-Umgebung eine Zertifikatswarnung wirft. Am Tag darauf meldet die Telefonzentrale, dass Smartphones keine Mails mehr abrufen. Und die Single-Sign-on-Anmeldung an Microsoft 365 stockt, weil im Federation-Dienst noch das alte Zertifikat hinterlegt ist. Das Zertifikat wurde erneuert. Nur eben nicht überall eingebunden.

Das Erneuern selbst ist heute mit wenigen Klicks erledigt. Was Zeit kostet und Fehler produziert, ist der zweite Teil: Ein Zertifikat steckt nicht an einer Stelle, sondern an vielen. In der HTTPS-Bindung einer IIS-Site, in den einzelnen Exchange-Diensten für IIS, SMTP, POP und IMAP, im SSL- und Service-Communication-Zertifikat von ADFS, im Gateway und im Connection Broker von RDS, in der Registry des SQL Servers. Jeder Dienst will es auf seine eigene Art haben. Und wer eine Stelle vergisst, hat einen Ausfall.

Renewal ohne vollständiges Binding-Update ist ein Ausfall

Ein erneuertes Zertifikat, das nicht in allen Diensten eingebunden ist, ist im Ergebnis nicht besser als ein abgelaufenes. Der Dienst liefert weiterhin das alte Zertifikat aus, bis es abläuft. Dann bricht die Verbindung ab, oft zu einem Zeitpunkt, an dem niemand mehr an den Wechsel denkt.

Und es wird häufiger: kürzere Laufzeiten

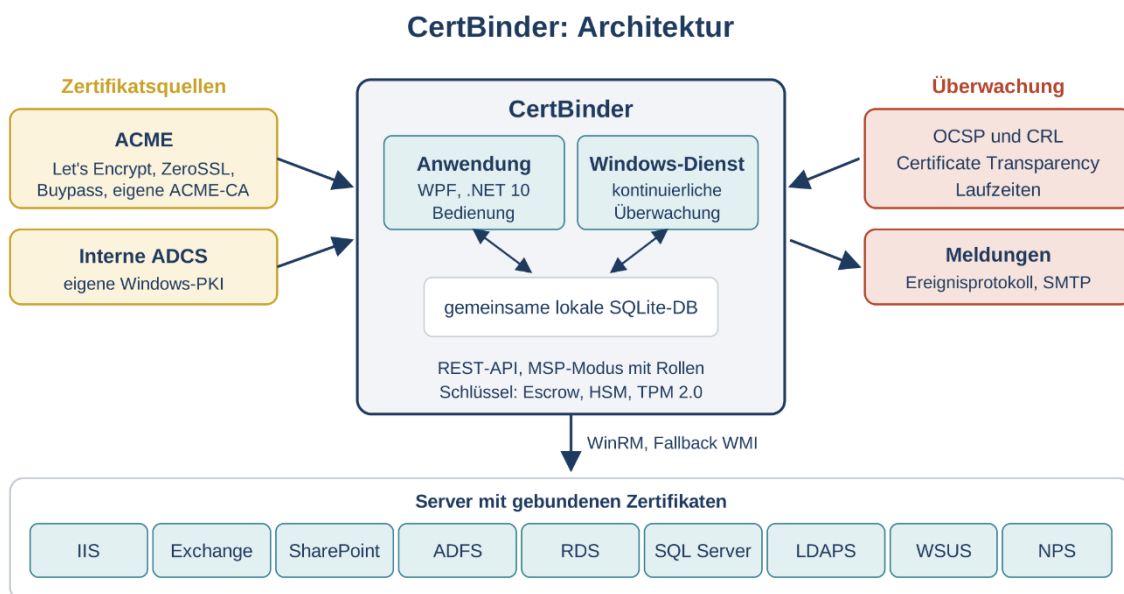
Bisher war der Zertifikatswechsel ein Ereignis, das einmal im Jahr stattfand und für das man sich einen Nachmittag freischaufelte. Das ändert sich gerade grundlegend. Öffentliche TLS-Zertifikate laufen künftig deutlich kürzer:

Ab	Maximale Laufzeit öffentlicher TLS-Zertifikate	Was das für dich heißt
März 2026	200 Tage	mindestens zwei Wechsel pro Jahr
März 2027	100 Tage	rund vier Wechsel pro Jahr
März 2029	47 Tage	Wechsel fast im Monatstakt

Wer heute für jeden Wechsel eine Checkliste abarbeitet und dabei zehn Server anfasst, macht das bei 47 Tagen Laufzeit nicht mehr von Hand. Oder anders gesagt: Was früher eine jährliche Inspektion war, wird zum Ölwechsel nach jeder Tankfüllung. Das funktioniert nur noch mit einer Werkstatt, die das automatisch erledigt.

Was CertBinder macht

CertBinder besteht aus zwei Teilen: einer Windows-Anwendung auf Basis von WPF und .NET 10 für die Bedienung und einem Windows-Dienst für die kontinuierliche Überwachung. Beide arbeiten mit einer gemeinsamen lokalen SQLite-Datenbank. Die Oberfläche ist deutsch und bietet einen Dunkelmodus für alle, die nachts Zertifikate wechseln müssen und dabei ihre Augen schonen wollen. Alle Daten bleiben bei dir im Haus, es gibt kein Cloud-Backend, über das deine Serverlandschaft nach außen gemeldet würde.

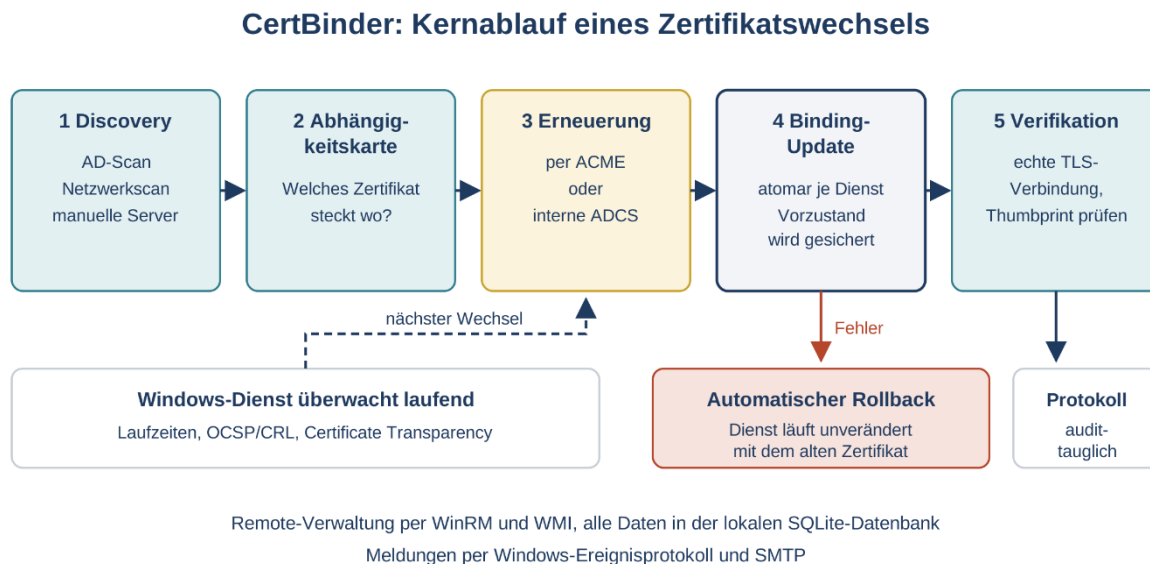


Architektur: Anwendung und Dienst teilen sich eine lokale SQLite-Datenbank und verwalten die Server per WinRM oder WMI.

Die Server verwaltet CertBinder aus der Ferne per WinRM und WMI. Ist WinRM auf einem Server nicht verfügbar, greift ein automatischer Fallback auf WMI. Du musst also nicht erst die halbe Infrastruktur umkonfigurieren, bevor das Werkzeug arbeiten kann.

Der Kernablauf in fünf Schritten

Was CertBinder von einem einfachen ACME-Client unterscheidet, ist der vollständige Ablauf: vom Aufspüren der Zertifikate über die Erneuerung bis zum Nachweis, dass das neue Zertifikat tatsächlich ausgeliefert wird.



Kernablauf: Discovery, Abhängigkeitskarte, Erneuerung, atomares Binding-Update mit Rollback und Verifikation per echter TLS-Verbindung.

1. Discovery: Was gibt es überhaupt?

Am Anfang steht die Bestandsaufnahme. CertBinder findet Server über einen Scan des Active Directory und des Netzwerks. Server, die dabei nicht auftauchen, trägst du manuell nach. Damit hast du zum ersten Mal einen Überblick, der nicht auf dem Gedächtnis einzelner Kollegen beruht.

2. Abhängigkeitskarte: Welches Zertifikat steckt wo?

Aus der Discovery entsteht eine Abhängigkeitskarte. Sie beantwortet die Frage, an der jeder manuelle Wechsel scheitert: In welchen Diensten auf welchen Servern ist dieses Zertifikat eingebunden? Das ist die Landkarte, ohne die du im Nebel fährst.

3. Erneuerung per ACME oder interner ADCS

Neue Zertifikate bezieht CertBinder per ACME bei Let's Encrypt, ZeroSSL, Buypass oder einer eigenen ACME-CA. Für interne Zertifikate nutzt er deine Windows-PKI auf Basis von Active Directory Certificate Services. Grundlagen zur eigenen PKI findest du im Kompetenzbereich [PKI mit ADCS](#).

4. Atomares Binding-Update mit automatischem Rollback

Jetzt kommt der Teil, der zählt. CertBinder bindet das neue Zertifikat je Dienst atomar ein. Vorher sichert er den Vorzustand. Geht beim Einbinden etwas schief, rollt er automatisch auf den gesicherten Stand zurück. Der Dienst läuft also entweder mit dem neuen Zertifikat oder unverändert mit dem alten weiter, aber nie in einem halbfertigen Zwischenzustand. Das

ist wie bei einer Banküberweisung: Entweder ist das Geld drüben, oder es ist noch hier. Irgendwo dazwischen gibt es nicht.

5. Verifikation per echter TLS-Verbindung

Ein Eintrag in der Konfiguration ist noch kein Beweis. CertBinder baut deshalb eine echte TLS-Verbindung zum Dienst auf und prüft den Thumbprint des Zertifikats, das tatsächlich ausgeliefert wird. Erst wenn der stimmt, gilt der Wechsel als erledigt. Alles landet in einem audit-tauglichen Protokoll.

Warum die Verifikation so wichtig ist

Manche Dienste übernehmen ein neues Zertifikat erst nach einem Neustart, andere halten das alte im Speicher, bis jemand nachhilft. Ein Blick in die Konfiguration zeigt dann das richtige Zertifikat, ausgeliefert wird aber das falsche. Die Prüfung über eine echte TLS-Verbindung schaut nicht in die Konfiguration, sondern auf das, was ein Client tatsächlich sieht.

Bindungen je Dienst

Jeder Microsoft-Dienst hat seine eigene Vorstellung davon, wo und wie ein Zertifikat hinterlegt wird. CertBinder kennt diese Eigenheiten und behandelt jeden Dienst so, wie er es braucht.

Dienst	Was CertBinder bindet
IIS	Sites, HTTPS-Bindungen, SNI
Exchange	IIS, SMTP, POP und IMAP einzeln
SharePoint	Webanwendungen, Zentraladministration
ADFS	SSL- und Service-Communication-Zertifikat, Primary/Secondary
RDS	Gateway, Web Access, Connection Broker
SQL Server	Registry, Reporting Services, Dienstneustart
Domänencontroller	LDAPS
WSUS	Port 8531
NPS	Zertifikat des Netzwerkrichtlinienservers

Gerade bei ADFS und SharePoint zeigt sich, wie viel Wissen in einem sauberen Wechsel steckt. Wer sich tiefer einlesen will, findet im Kompetenzbereich [ADFS](#) und im Kompetenzbereich [SharePoint](#) die Hintergründe.

Weitere Funktionen

Über den Kernablauf hinaus bringt CertBinder eine Reihe von Funktionen mit, die im Betrieb den Unterschied machen:

- **Schlüsselverwaltung und Key Escrow:** private Schlüssel geordnet verwalten und sichern.
- **HSM-Anbindung (PKCS#11) und TPM 2.0:** Schlüssel in Hardware schützen, wo es die Anforderungen verlangen.
- **OCSF- und CRL-Überwachung:** erkennen, wenn Sperrinformationen nicht erreichbar sind oder ein Zertifikat gesperrt wurde.
- **Certificate-Transparency-Überwachung:** Fehlausstellungen erkennen, also Zertifikate für deine Domänen, die du nicht selbst beantragt hast.
- **Benachrichtigung** per Windows-Ereignisprotokoll und SMTP, damit Meldungen in deinem vorhandenen Monitoring landen.
- **REST-API** für die Anbindung an eigene Abläufe und Werkzeuge.
- **Mandantenfähigkeit mit Rollen (MSP-Modus):** Dienstleister verwalten mehrere Kunden sauber getrennt.

CertBinder auf einen Blick

- **Produkttyp:** Werkzeug für den Lebenszyklus von TLS-Zertifikaten: Discovery, Erneuerung, Binding-Update, Verifikation, Überwachung
- **Plattform:** Windows-Anwendung (WPF, .NET 10) plus Windows-Dienst; deutsche Oberfläche, Dunkelmodus
- **Datenhaltung:** gemeinsame lokale SQLite-Datenbank, kein Cloud-Backend
- **Remote-Verwaltung:** WinRM und WMI mit automatischem Fallback
- **Zertifikatsbezug:** ACME (Let's Encrypt, ZeroSSL, Buypass, eigene ACME-CA) und interne ADCS
- **Zielgruppe:** Betreiber von Microsoft-Serverlandschaften on-premises, Systemhäuser und Dienstleister (MSP-Modus)
- **Lizenzmodell:** Einmallizenz on-premises, Preise auf Anfrage
- **Hersteller:** Eigenentwicklung von boddenberg.de

Typischer Einsatz

Der Einstieg beginnt mit der Discovery. Viele Organisationen erleben dabei den ersten Aha-Moment: Das Zertifikat, von dem alle dachten, es laufe nur auf dem Reverse Proxy, steckt zusätzlich auf drei weiteren Servern, von denen einer seit Jahren nicht mehr angefasst wurde. Die Abhängigkeitskarte macht solche Fundstücke sichtbar, bevor sie zum Ausfall werden.

Danach legst du fest, welche Zertifikate per ACME und welche über die interne ADCS erneuert werden. Der Windows-Dienst übernimmt die kontinuierliche Überwachung: Laufzeiten, Sperrinformationen, Certificate Transparency. Steht ein Wechsel an, läuft er nach dem immer gleichen Muster ab: erneuern, Vorzustand sichern, atomar binden, per TLS prüfen, protokollieren. Meldungen landen im Windows-Ereignisprotokoll und per Mail bei dir.

Mit der Landkarte anfangen

Auch wenn du die Erneuerung noch nicht sofort automatisieren willst: Allein die Discovery und die Abhängigkeitskarte lohnen sich. Sie beantworten die Frage, welches Zertifikat wo steckt, und machen aus dem nächsten Wechsel eine Aufgabe mit klarer Liste statt einer Schnitzeljagd.

Für wen CertBinder gedacht ist

- **IT-Abteilungen mit Microsoft-Serverlandschaft on-premises**, die IIS, Exchange, SharePoint, ADFS, RDS oder SQL Server betreiben und bei jedem Zertifikatswechsel mehrere Server anfassen.
- **IT-Leiter**, die die kürzeren Laufzeiten öffentlicher Zertifikate nicht mit mehr Personal, sondern mit einem sauberen Prozess abfangen wollen.
- **Organisationen mit Prüfpflichten**, die ein audit-taugliches Protokoll der Zertifikatswechsel brauchen.
- **Systemhäuser und Dienstleister**, die über den MSP-Modus mehrere Kunden mandantengetrennt betreuen; Partner und Multiplikatoren bekommen eigene Konditionen, bitte anfragen.

Für die Management-Ebene lässt sich der Nutzen in einem Satz zusammenfassen: Ein Zertifikatswechsel ist kein Risiko mehr, das an einzelnen Personen hängt, sondern ein dokumentierter, wiederholbarer Vorgang. Für den Admin heißt das: keine Nachtschichten mehr, weil irgendein Dienst das alte Zertifikat noch ausliefert.

Editionen und Lizenzierung

CertBinder wird als **Einmallizenz on-premises** angeboten. Kein Abo, keine Cloud-Abhängigkeit. Preise und Lizenzbedingungen nennt dir Uli auf Anfrage. Systemhäuser, Partner und Multiplikatoren bekommen eigene Konditionen, bitte anfragen.

Beratung, Einführung und Schulung

Ein Zertifikatswechsel berührt fast jede Ecke einer Microsoft-Infrastruktur. Bei der Einführung von CertBinder stellen sich deshalb auch Fragen jenseits des Werkzeugs: Wie ist die eigene PKI aufgestellt? Welche Dienste brauchen öffentliche, welche interne Zertifikate? Wie fügt sich die Überwachung in das bestehende Monitoring ein? Aus rund 30 Jahren

Microsoft-Enterprise-Projekten mit ADFS, Exchange, SharePoint und ADCS kennt Uli die typischen Stolperstellen. Auf Wunsch begleitet er die Einführung von der Discovery bis zur Schulung der Administratoren, damit der erste automatisierte Wechsel auch der erste ruhige ist.

Häufige Fragen

Ist CertBinder nur ein weiterer ACME-Client?

Nein. Die Erneuerung per ACME ist nur ein Schritt. Der Kern von CertBinder ist das, was danach kommt: herausfinden, wo das Zertifikat überall steckt, es je Dienst atomar mit Rollback einbinden und per echter TLS-Verbindung prüfen, ob es auch ausgeliefert wird.

Funktioniert CertBinder auch mit meiner eigenen Windows-PKI?

Ja. Neben ACME bei Let's Encrypt, ZeroSSL, Buypass oder einer eigenen ACME-CA unterstützt CertBinder die Erneuerung über die interne ADCS. Wenn du vorher wissen willst, wie gesund deine eigene CA ist, hilft dir der [PKI Diagnostiker](#) bei der Bestandsaufnahme.

Was passiert, wenn das Einbinden in einen Dienst fehlschlägt?

CertBinder sichert vor jedem Binding-Update den Vorzustand. Schlägt das Update fehl, rollt er automatisch auf diesen Stand zurück. Der Dienst läuft dann unverändert mit dem bisherigen Zertifikat weiter, und du bekommst eine Meldung.

Brauche ich WinRM auf allen Servern?

Nicht zwingend. CertBinder verwaltet die Server per WinRM und WMI. Ist WinRM nicht verfügbar, greift automatisch ein Fallback auf WMI.

Wo liegen meine Daten?

In einer lokalen SQLite-Datenbank, die Anwendung und Windows-Dienst gemeinsam nutzen. Es gibt kein Cloud-Backend. Deine Serverlandschaft und deine Zertifikatsdaten bleiben bei dir.

Kann ich CertBinder als Dienstleister für mehrere Kunden einsetzen?

Ja. CertBinder ist mandantenfähig und bietet Rollen für Dienstleister (MSP-Modus). Damit betreust du mehrere Kunden sauber getrennt. Für Systemhäuser und Partner gibt es eigene Konditionen auf Anfrage.

Kümmert sich CertBinder auch um S/MIME-Zertifikate?

Nein, CertBinder ist für TLS-Zertifikate in Serverdiensten gebaut. Für S/MIME-Zertifikate zur E-Mail-Signatur und -Verschlüsselung gibt es den [SMimeManager](#).

Weiter geht's

- Kompetenzbereich [PKI mit ADCS](#): Grundlagen und Praxis der Windows-PKI

- Kompetenzbereich [ADFS](#): Zertifikate im Federation-Dienst
- Kompetenzbereich [SharePoint](#)
- [PKI Diagnostiker](#): Bestandsaufnahme und Härungsprüfung der eigenen CA
- [SMimeManager](#): S/MIME-Zertifikate zentral verwalten