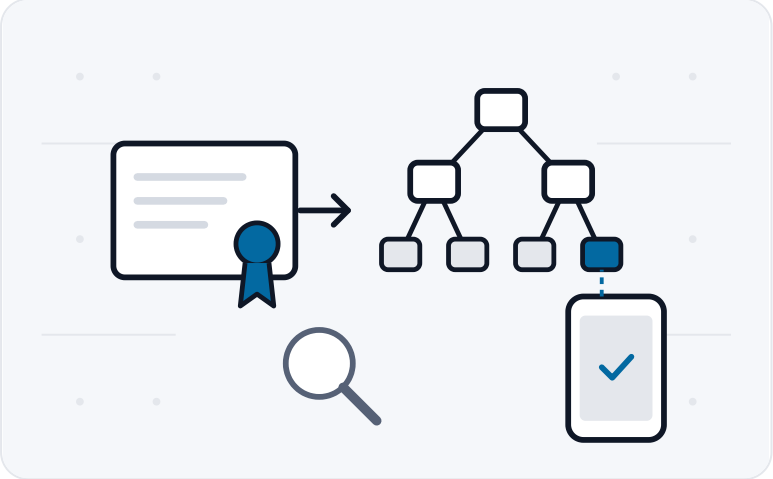
A-SIT Technology
E-Government Innovationszentrum

27. August 2025 Konzept · Digitale Identität · Transparenz · 32 Min. Lesezeit

Certificate Transparency für die Relying Party-Zertifikate

Edona Fasllija



Autorinnen und Autoren	Edona Fasllija
Quelle	https://technology2.a-sit.at/veroeffentlichungen/certificate-transparency-fuer-die-relying-party-zertifikate/
Hinweis	Maschinell übersetzt.

Kernaussagen

- Klassisches Certificate Transparency protokolliert keine Widerrufe und kann bei der SCT-Prüfung Nutzerverhalten preisgeben; es genügt nicht.
- RPCT protokolliert Ausstellung und Widerruf von RPAC und RPRC öffentlich; Monitore erkennen übermäßige Berechtigungen im Abgleich mit dem Register.
- Ein Merkle Patricia Trie mit Hash-Kette pro RP ermöglicht Abfragen in O(log N) statt Durchsuchen des ganzen Logs.
- Die Wallet prüft Inklusionsnachweise offline mit vom RP gelieferten Nachweisen, sodass keine Live-Abfrage ihr Nutzungsverhalten verrät.

- Ein Widerruf wird als neues Kettenglied protokolliert; eine separate Widerrufsliste ist nicht nötig und die Historie bleibt prüfbar.

Das europäische Rahmenwerk für digitale Identitäten eIDAS 2 überträgt die Kontrolle über eine sichere Wallet-App an den Nutzer. Relying Parties (RPs), die auf Wallet-Daten zugreifen möchten, müssen sich zuvor bei den Behörden registrieren und ihre beabsichtigte Datennutzung (Zweck) angeben. Während einer Transaktion authentifiziert ein RP Access Certificate (RPAC) die Service-Instanz der RP; ein RP Registration Certificate (RPRC) listet die spezifischen Attribute auf, die die RP anfordern darf. Zusammen sorgen diese Zertifikate für Privacy by Design: Die Wallet kann die Nutzer darüber informieren, welche RP welche Attribute anfordert, und Anfragen ablehnen, die über das hinausgehen, was offiziell als notwendig registriert wurde.

Die Verordnung weist jedoch eine kritische Lücke bei der Durchsetzung auf: Es gibt keinen integrierten Mechanismus, mit dem Nutzer oder Prüfer überprüfen können, was registriert wurde oder ob diese Grenzen eingehalten werden. Dieses Projekt schlägt RP Certificate Transparency (RPCT) vor, eine öffentliche, kryptografisch überprüfbare Log-Infrastruktur, die von Certificate Transparency inspiriert und an das RP-Authentifizierungs- und Autorisierungsmodell von eIDAS 2 angepasst ist. RPCT bietet Nachweise für Aufnahme, Konsistenz und Historie, die für nicht-verknüpfbare Wallet-Verifikation geeignet sind, und unterstützt Widerrufstransparenz sowie RP-spezifische Prüfpfade. Es ermöglicht Aufsichtsbehörden sowie externen Überwachungsstellen, Fehlvergaben und übermäßig weit gefasste Anfragen zu erkennen.

1. Einleitung und Motivation

Das Rahmenwerk der EU für digitale Identitäten (EUDI) unter eIDAS 2[1] führt digitale Identitäts-Wallets ein, die den Zugriff anhand von Zertifikaten kontrollieren, die an Relying Parties (RPs) ausgestellt werden. RPs müssen sich bei einer nationalen Behörde registrieren und ihre Datenzugriffsberechtigungen angeben. Jede RP muss von benannten Certificate Authorities (CAs) zwei Zertifikate beziehen: ein RP Access Certificate (RPAC), das die Service-Instanz der RP authentifiziert, und, soweit zutreffend, ein RP Registration Certificate (RPRC), das die Nutzerattribute auflistet, die die RP anfordern darf. Dieses Zwei-Zertifikate-Schema erzwingt Datenminimierung: Die Wallet überprüft die Identität der RP über das RPAC und stellt sicher, dass alle angeforderten Attribute im deklarierten Umfang des RPRC liegen. Nutzer werden darüber informiert, welche Attribute angefordert werden, und können gewarnt werden, wenn eine RP Daten außerhalb ihres registrierten Umfangs anfordert.

Ohne einen Prüfmechanismus könnte eine böswillige oder kompromittierte CA heimlich Zertifikate fehlerhaft ausstellen. Eine CA könnte beispielsweise ein RPRC mit übermäßigen Berechtigungen ausstellen, die von der RP gar nicht beantragt oder registriert wurden, sodass ein unseriöser Dienst mehr Nutzerdaten anfordern könnte, als erlaubt ist. Ebenso könnte ein gefälschtes RPAC an eine betrügerische RP ausgestellt werden, was Phishing oder Identitätsbetrug ermöglicht. Eine solche Fehlausstellung bliebe für Nutzer und andere Parteien unentdeckt, wenn es keine Transparenz oder öffentliches Log der ausgestellten Zertifikate gibt. Ein weiteres Risiko ist der stille Widerruf: Eine CA könnte das Zertifikat einer RP ohne Benachrichtigung widerrufen, was Dienste stören oder, schlimmer, einem Angreifer ermöglichen könnte, ein widerrufenes

Zertifikat vorzulegen, wenn Verifizierer keine aktuellen Widerrufsinformationen haben. Diese Bedrohungen untergraben die Privatsphäre der Nutzer, ihre Einwilligung und die Absicht der Verordnung. Das Web PKI hat die Fehlausstellung von TLS-Zertifikaten über Certificate-Transparency-Logs (CT)[2] adressiert. Standard-CT verlangt von CAs, jedes TLS-Zertifikat in einem öffentlichen, nur anfügbaren Log zu veröffentlichen, sodass jeder nach unautorisierten Zertifikaten suchen kann. Eine naive Anwendung von CT auf eIDAS-RP-Zertifikate reicht jedoch nicht aus. Klassische CT-Logs protokollieren nur die Ausstellung von Zertifikaten, nicht deren Widerruf. Es gibt keine eingebaute Möglichkeit festzustellen, ob ein Zertifikat später widerrufen wurde. Außerdem liefern CT-Logs Signed Certificate Timestamps (SCTs) als Nachweis der Protokollierung, doch Clients müssen zur Validierung von SCTs oft Log-Server oder Monitore kontaktieren. Dadurch wird der Browserverlauf der Nutzer preisgegeben (ein Privatsphäre-Problem), und das Vertrauen wird auf die Log-Server verlagert. Zudem verlässt sich CT darauf, dass externe Monitore Fehlausstellungen durch das Durchsuchen des gesamten Logs finden, ein Ansatz, der bandbreiten- und rechenintensiv ist. Zusammengefasst braucht das EUDI-Wallet-Ökosystem eine Transparenzlösung, die sowohl Ausstellung als auch Widerruf abdeckt, die Privatsphäre der Nutzer wahrt und für potenziell Millionen von RPs skaliert.

Um diese Lücken zu schließen, schlägt dieses Projekt Relying Party Certificate Transparency (RPCT) vor: eine Transparenz-Logging-Architektur, die auf den Kontext der eIDAS-2-RP-Zertifikate zugeschnitten ist. Das Ziel von RPCT ist es, die Fehlausstellung von RPACs/RPRCs abzuschrecken und aufzudecken sowie jeden Missbrauch des Autorisierungsumfangs zu erkennen. Jedes ausgestellte oder widerrufene RP-Zertifikat wird in einem öffentlichen, manipulationssicheren Log erfasst. So können RPs, Wallet-Anbieter, Aufsichtsbehörden und Nutzer prüfen, wer welche Daten anfordern darf. Im Kern fungiert RPCT als „Wächter des Registers“[3] und stellt sicher, dass die offiziellen RP-Registrierungsinformationen (das „Register“) mit den in der Praxis verwendeten Zertifikaten übereinstimmen.

2. Hintergrund

Dieser Abschnitt liefert Hintergrundinformationen zu den beiden zentralen RP-Zertifikatstypen in eIDAS 2: RPACs (zur RP-Authentifizierung) und RPRCs (für den Autorisierungsumfang). Der Abschnitt beschreibt, wie sie funktionieren und warum sie gesetzlich vorgeschrieben sind. Abschließend zeigen wir das Vertrauensproblem auf, das entsteht, wenn diese Zertifikate nicht öffentlich protokolliert werden, und wie ein böswilliger Akteur das System unterlaufen kann.

2.1. RPACs und RPRCs in eIDAS 2

RP Access Certificates (RPACs): Unter eIDAS 2 muss sich eine RP gegenüber der Wallet des Nutzers mit einem RP Access Certificate identifizieren. Ein RPAC ist im Wesentlichen ein Authentifizierungszertifikat für eine Service-Instanz (analog zu einem Webserver-Zertifikat in TLS), das von einer vertrauenswürdigen Stelle (CA) ausgestellt wird. Es bindet die Identität der RP (und gegebenenfalls Domain- oder Organisationsinformationen) an einen öffentlichen Schlüssel, sodass die Wallet die Identität der RP überprüfen kann, bevor Daten weitergegeben werden. Dies geht auf Artikel 5b(8) der Verordnung zurück, der die RP-Authentifizierung vorschreibt: „Wallets dürfen keine Daten an eine RP weitergeben, die kein gültiges RPAC besitzt.“ In der Praxis prüft die Wallet, wenn ein Nutzer eine Credential-Präsentation gegenüber einer RP versucht, ob die RP ein gültiges, aktuelles RPAC vorlegt. Das verhindert, dass sich unseriöse oder nicht registrierte Dienste als legitime RPs ausgeben.

RP Registration Certificates (RPRCs): Zusätzlich zum Identitätsnachweis erhalten RPs (optional) ein RPRC, das die beabsichtigte Datennutzung und die konkreten Attribute kodiert, die die RP anfordern darf. Im Wesentlichen ist ein RPRC ein maschinenlesbarer „Erlaubnisschein", der beispielsweise festhält, dass eine Banking-App nach Name und Kontostatus fragen darf, nicht aber nach dem vollständigen Identitätsprofil. Dieses Zertifikat wird üblicherweise von einer mit einer Registrierungsstelle (Registrar) verbundenen RP-CA ausgestellt, nachdem die RP ihren Datennutzungszweck bei einer Regierungsstelle oder zuständigen Behörde registriert hat (eine Anforderung nach Artikel 5b(1–2) von eIDAS). Abbildung 1 zeigt einen Überblick darüber, wie diese Zertifikate in eIDAS 2 verwendet werden.

Die Wallet erzwingt, dass jedes Attribut, das eine RP während einer Transaktion anfordert, von den im RPRC aufgeführten Attributen abgedeckt ist. Fordert die RP mehr Daten an als aufgelistet, kann die Wallet dies dem Nutzer melden oder die Anfrage sogar ablehnen und so den Grundsatz der Rechtmäßigkeit und der Nutzereinwilligung wahren (RPs „dürfen nicht mehr Daten anfordern, als im Register angegeben"). RPACs und RPRCs sind zentral für den Aufbau von Vertrauen im Ökosystem. Sie versichern den Nutzern, dass (a) die RP die ist, die sie vorgibt zu sein (RPAC), und (b) die RP nur Daten anfordert, die sie rechtmäßig registriert hat (RPRC). Diese Zusicherungen hängen jedoch von der Integrität der CAs/Behörden ab, die die Zertifikate ausstellen. Wenn eine CA betrügt (absichtlich oder aufgrund einer Kompromittierung) und ein Zertifikat außerhalb der Regeln ausstellt, würden Nutzer und andere RPs das nicht erfahren. Beispielsweise könnte eine böswillige RP mit einer korrupten CA zusammenarbeiten, um ein RPRC zu erhalten, das übermäßige Attribute auflistet (und damit gegen die im offiziellen Register hinterlegten Angaben verstößt). Oder ein Angreifer könnte eine CA dazu bringen, ein RPAC für einen gefälschten Dienstnamen auszustellen, der einem legitimen gleicht (ein Phishing-Versuch). In beiden Fällen kann die Wallet allein die Anomalie nicht erkennen, da sie die ihr vorgelegten Zertifikate gewissenhaft verifiziert, es sei denn, es gibt ein externes Prüf-Log.

CT-Grundlagen: Im Web-PKI-CT wird jede Ausstellung eines TLS-Zertifikats in einem öffentlichen Log als neuer Eintrag (Blatt) in einem Merkle-Hash-Baum erfasst. Logs veröffentlichen regelmäßig einen signierten Baumkopf (Signed Tree Head, STH) als globale Zusammenfassung aller Einträge. CAs stellen Clients ein SCT (Signed Certificate Timestamp) als Nachweis der Übermittlung an das Log aus. Monitore (jede interessierte Partei) laden Log-Einträge herunter und prüfen sie, um Zertifikate für Domains zu finden, die sie interessieren, und markieren fehlerhaft ausgestellte Zertifikate (z. B. ein gefälschtes Zertifikat für example.com). Auditoren (das können Browser oder Dritte sein) prüfen, dass Log-Aktualisierungen nur anfügend und über die Zeit konsistent sind. Dieses System ermöglicht es, Fehlverhalten von CAs in der Praxis aufzudecken.

Einschränkungen von CT für unseren Anwendungsfall: Das Design von CT funktioniert für Domain-Zertifikate, löst aber unser RP-Zertifikatsproblem nicht unmittelbar. Die wichtigsten Einschränkungen führen wir im Folgenden auf:

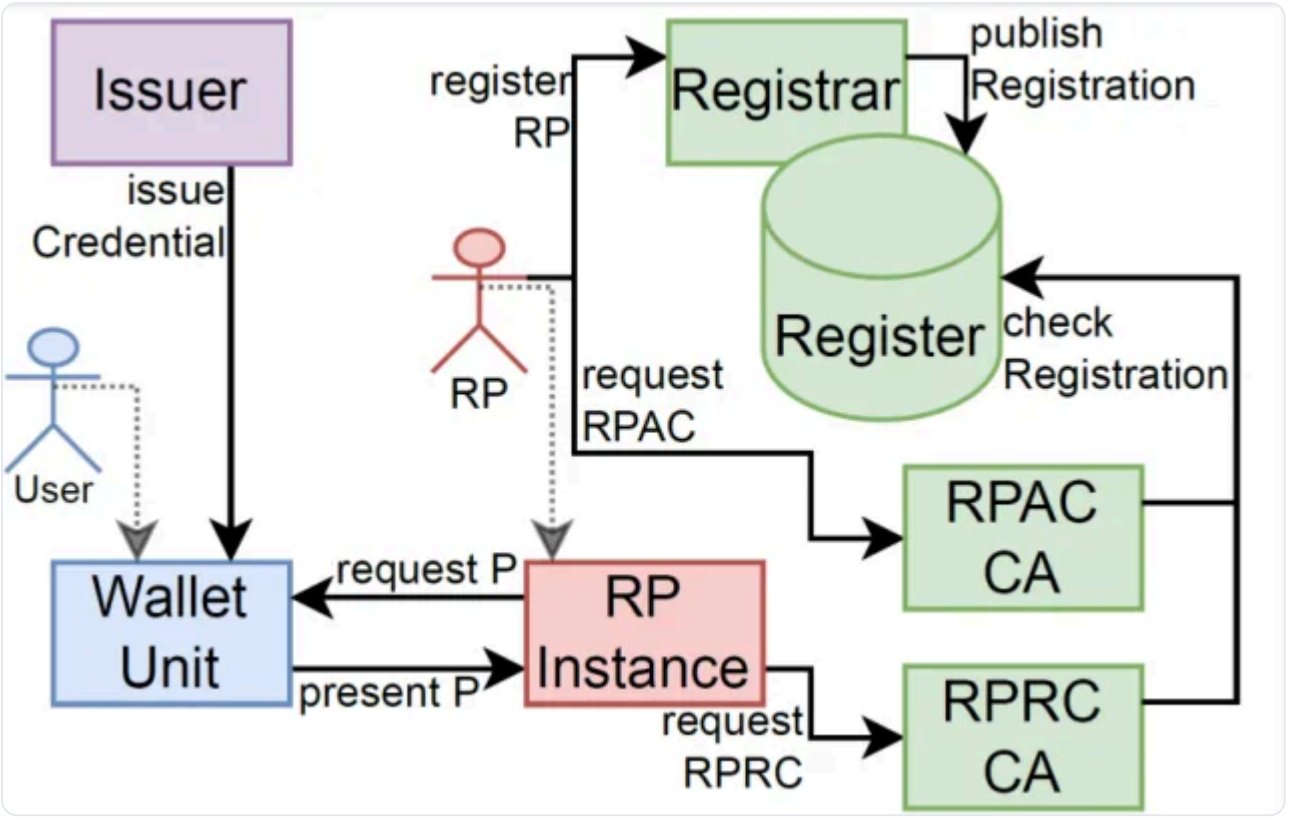


Abbildung 1: RPAC/RPRC-Überblick in eIDAS 2

2.2. Von CT zu RPCT

Certificate Transparency (CT) führte öffentliche Logs von Zertifikaten ein, um Fehlausstellungen zu erkennen. Im CT-Modell protokollieren CAs jedes Zertifikat, Log-Server pflegen einen nur anfügbaren Merkle-Baum der Einträge, und unabhängige Monitore und Auditoren überwachen die Logs. Die klassische CT-Ökosystemarchitektur sieht wie folgt aus: CAs übermitteln Zertifikate an Log-Server, Monitore beobachten die Logs auf verdächtige Einträge (z. B. ein gefälschtes Zertifikat für eine Domain), und Auditoren stellen die Integrität des Logs sicher. RPCT übernimmt eine ähnliche Architektur aus Logs, Monitoren und Auditoren, passt sie aber an den EUDI-Wallet-Kontext mit RPs und deren speziellen Zertifikaten an.

Keine Widerrufsprotokollierung: Wird in CT ein Zertifikat widerrufen, wird dieses Ereignis weder protokolliert noch im Merkle-Baum abgebildet (der Baum wächst nur durch neue Einträge). CT allein kann einem Client daher nicht beweisen, dass ein vorgelegtes Zertifikat noch gültig ist, sondern nur, dass es einmal protokolliert wurde.

Datenschutzbedenken: Die Verifikation von SCTs kann offenlegen, welche Websites ein Nutzer besucht. Ein Browser könnte beispielsweise ein CT-Log abfragen oder einen Monitor zu einem bestimmten SCT befragen und damit effektiv offenlegen, dass „Nutzer X die Website Y besucht“. In einem Wallet-Szenario könnte es, wenn die Wallet bei jeder Verwendung eines Credentials ein RP-Zertifikats-Log abfragen müsste, die Muster der Dienstnutzung offenlegen und die Datenschutzanforderungen von eIDAS verletzen (keine Nachverfolgung von Nutzern durch Dritte).

Effizienz: Klassisches CT erfordert das Durchsuchen der gesamten Logs, um relevante Einträge zu finden. Bei potenziell Tausenden von RPs pro Land wäre es sehr ineffizient, wenn Monitore ständig riesige Log-Bäume erneut herunterladen müssten, um problematische RP-Zertifikate zu finden.

Einfluss von Key Transparency (KT)[4], [5]: Ein verwandtes Konzept ist Key Transparency, das die Transparenzidee auf öffentliche Schlüssel von Nutzern ausdehnt (wie in CONIKS). KT-Systeme verwenden ein nur anfügbares Wörterbuch (häufig einen Merkle-Präfixbaum), das jeden Nutzer (oder Bezeichner) seinem aktuellen öffentlichen Schlüssel zuordnet. Statt alles zu durchsuchen, können Clients den Schlüssel eines bestimmten Nutzers direkt mit einem kryptografischen Korrektheitsnachweis nachschlagen. Das inspiriert einen Teil unserer Lösung: Wir wollen eine Abfrageeffizienz pro RP. Mit anderen Worten: Es soll möglich sein, alle Zertifikate für „RP X“ einfach aus dem Log abzurufen, ohne unzusammenhängende Einträge durchkämmen zu müssen.

3. RPCT-Design

RPCT verbindet die öffentliche Rechenschaftspflicht von CT mit der effizienten Abfrage von KT. Es behält die vertrauten Rollen bei: RP-CAs (für RPACs und RPRCs) müssen jede Ausstellung und jeden Widerruf an das Log übermitteln. Der RPCT-Log-Server nimmt diese Übermittlungen entgegen und pflegt eine nur anfügbare Datenstruktur (dazu später mehr), die sowohl chronologische Prüfungen als auch schlüsselbasierte Abfragen erlaubt. Wallets (Clients) verhalten sich ähnlich wie Browser in CT: Sie nutzen Nachweise aus dem Log, um die Zertifikate einer RP zu validieren, bevor Daten freigegeben werden. Monitore (die von Aufsichtsbehörden, Watchdog-Organisationen oder auch von RPs selbst betrieben werden können) rufen Log-Daten ab und achten auf Auffälligkeiten: z. B. ein RPRC, das Attribute enthält, die über das hinausgehen, was diese RP anfordern dürfte. Sie können den Inhalt des Logs mit den offiziellen RP-Registern abgleichen, die von den Mitgliedstaaten veröffentlicht werden (wie in eIDAS Artikel 5b vorgeschrieben)[6], um Inkonsistenzen zu erkennen. Auditoren stellen sicher, dass der Log-Server ehrlich ist: dass der Zustand des Logs nie einen Eintrag vergisst oder sich für verschiedene Personen in inkonsistente Versionen aufspaltet.

Erweiterung der CT-Prinzipien: Im Kern erweitert RPCT CT in drei entscheidenden Punkten: (1) Widerrufstransparenz: Nicht nur Hinzufügungen werden protokolliert, sondern auch, wann ein Zertifikat widerrufen oder entfernt wird, sodass Clients einen kryptografischen Nachweis erhalten, dass ein Zertifikat nicht widerrufen wurde (oder umgekehrt erkennen, dass es widerrufen wurde). (2) Organisation pro RP: Eine Datenstruktur erlaubt den direkten Zugriff auf die Zertifikatskette einer RP, vermeidet vollständige Log-Durchsuchungen und macht das Monitoring gezielter und skalierbarer. (3) Privatsphäre-wahrende Nachweise: Wallets können die Zertifikate einer RP offline validieren, mit Nachweisdaten, die vorab abgerufen oder von der RP bereitgestellt werden können, sodass die Wallet das Log nicht in Echtzeit abfragen muss (und Beobachter nicht nachverfolgen können, mit welcher RP ein Nutzer interagiert). Diese Verbesserungen zielen darauf ab, die spezifischen Anforderungen von eIDAS 2 an Rechenschaftspflicht (öffentliche Protokollierung von RP-Zertifikaten), Übereinstimmung mit den offiziellen Registrierungen und Privatsphäre zu erfüllen.

3.1 Entitäten und Rollen

Dieser Abschnitt beschreibt die Systemkomponenten und ihr Zusammenspiel, um CAs zur Rechenschaft zu ziehen und Aufsichtsbehörden die Werkzeuge zur Überwachung des gesamten Systems zu geben.

In der RPCT-Systemarchitektur (dargestellt in Abbildung 2) gibt es die folgenden zentralen Entitäten:

RP-CAs: Dies sind die Stellen (staatlich benannte oder akkreditierte private CAs), die RPACs und RPRCs ausstellen. Unter RPCT ist eine RP-CA verpflichtet, jedes neu ausgestellte oder widerrufene Zertifikat unverzüglich an das Transparenz-Log zu übermitteln. Man kann sie sich als „Anfüger" („Appender") des Logs vorstellen. Protokolliert eine CA ein Zertifikat nicht, gilt dieses Zertifikat für Wallets als nicht vertrauenswürdig (da es keinen Protokollierungsnachweis gibt). RPCT Logger (Log-Server): Ein Server (oder verteilter Dienst), der das nur anfügbare Log der RP-Zertifikate pflegt. Er stellt APIs bereit, über die CAs Einträge hinzufügen und Clients/Monitore Nachweise abfragen können. Der Log-Server erzeugt regelmäßig eine neue globale Zusammenfassung (einen Merkle-Baum-Wurzelhash), die alle bisherigen Einträge festschreibt. In RPCT basiert dieses Log auf einer spezialisierten authentifizierten Datenstruktur (siehe Abschnitt 4). Der Logger erzeugt außerdem kryptografische Nachweise: Wird ein Zertifikat hinzugefügt, kann er einen Inklusionsnachweis zurückgeben, der zeigt, dass das Zertifikat nun in der Zusammenfassung des Logs enthalten ist. Ebenso kann er Nachweise der Abwesenheit (Nicht-Inklusion) oder der Konsistenz zwischen Log-Zuständen über die Zeit liefern. Wallet: Die EUDI-Wallet-Anwendung des Nutzers (z. B. auf einem Telefon) fungiert als verifizierender Client. Wenn eine RP während einer Transaktion ihre Credentials (RPAC und RPRC) vorlegt, muss sie der Wallet außerdem einen Log-Nachweis (oder mehrere Nachweise) liefern. Die Wallet verifiziert die Zertifikate und ihre Log-Nachweise, bevor sie Nutzerdaten freigibt. Das heißt, die Wallet prüft, dass RPAC/RPRC tatsächlich protokolliert und nicht widerrufen wurden. Fehlt der Nachweis oder ist er ungültig, sollte die Wallet die Credential-Präsentation ablehnen, damit nicht protokollierte (und damit potenziell betrügerische) Zertifikate nicht akzeptiert werden. Monitor-Dienste: Diese können von jedem betrieben werden, wahrscheinlicher jedoch von zuständigen Behörden (z. B. Datenschutzbehörden, nationalen Aufsichtsstellen) oder datenschutzorientierten NGOs. Ein Monitor beobachtet das Log fortlaufend auf verdächtige Zertifikate. Eine Datenschutzbehörde könnte beispielsweise alle unter ihrem Ländercode ausgestellten RPRCs überwachen und prüfen, ob die angeforderten Attribute mit dem übereinstimmen, was die RP registriert hat. Wird ein unzulässiger Eintrag entdeckt (etwa wenn eine RP nur „Name und Geburtsdatum" registriert hat, ein RPRC im Log für diese RP aber auch die „Sozialversicherungsnummer" auflistet), kann der Monitor die betroffenen Parteien (die RP, Aufsichtsbehörden oder die Öffentlichkeit) alarmieren. Monitore könnten auch auf unautorisierte RPACs achten (z. B. ein Zertifikat für eine RP, die nicht im offiziellen Register steht). Da die Datenstruktur von RPCT Abfragen pro RP erlaubt, kann sich ein Monitor auf bestimmte, für seinen Aufsichtsbereich relevante Abschnitte des Logs konzentrieren (etwa nach Land oder Sektor), statt das gesamte globale Log zu verarbeiten. Auditoren: Ein Auditor stellt sicher, dass der Betrieb des Logs ehrlich und nur anfügend bleibt. Auditoren können unabhängige Dienste sein oder sogar in Clients integriert werden. Auditoren rufen regelmäßig Log-Zusammenfassungen ab und nutzen Konsistenznachweise, um sicherzustellen, dass das Log nicht manipuliert wurde (d. h., dass eine neuere Zusammenfassung tatsächlich alle älteren Einträge enthält und die Historie nicht umgeschrieben wurde). Auditoren prüfen außerdem, dass das Log nicht verschiedenen Personen unterschiedliche Sichten präsentiert (keine gespaltene Historie). In der Praxis lassen sich viele dieser Funktionen automatisieren: Browser übernehmen in CT beispielsweise die Rolle des Auditors, indem sie eine konsistente Sicht auf die Logs verlangen. In eIDAS könnten etwa die Europäische Kommission oder ein Zusammenschluss von Mitgliedstaaten Auditprozesse betreiben, um sicherzustellen, dass offizielle RPCT-Logs korrekt arbeiten.

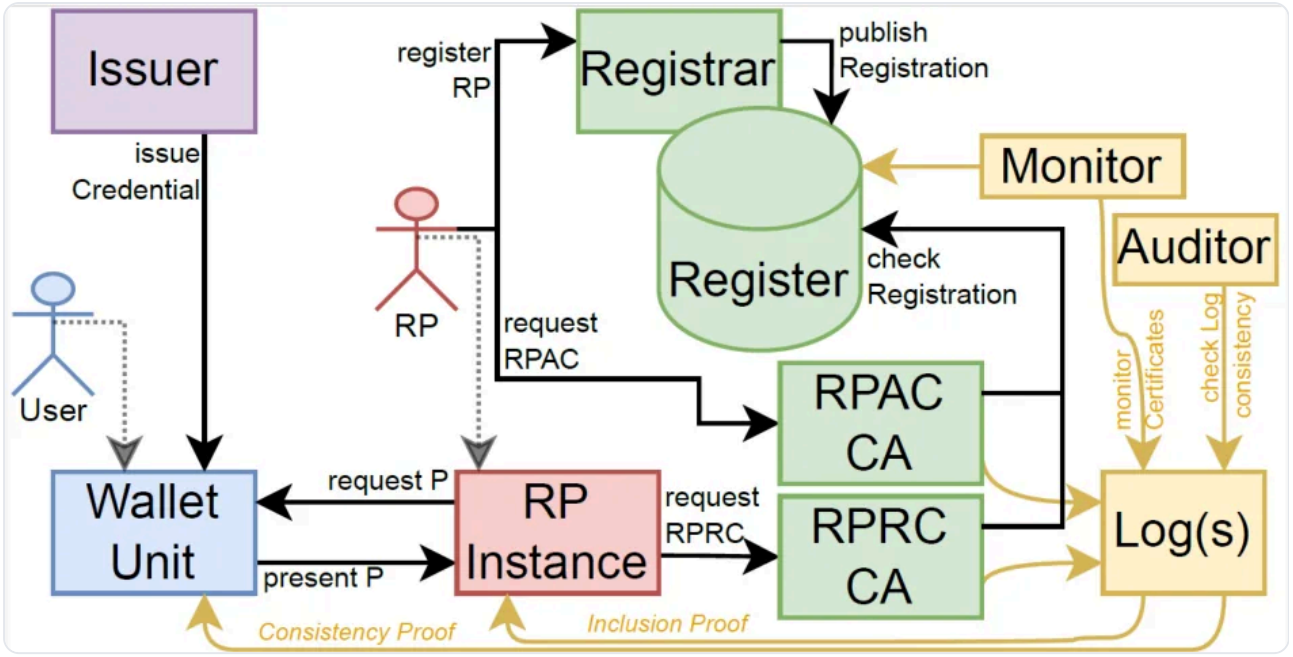


Abbildung 2: RPCT-Überblick

3.2 RPCT-Ablauf/Lebenszyklus:

1. RP-Registrierung: Eine RP registriert sich zunächst bei einem RP-Registrar (gemäß eIDAS-Durchführungsrechtsakt) und gibt dabei ihre Identität und beabsichtigte Datennutzung an. Der Registrar (eine Regierungsstelle) fügt die RP in das offizielle RP-Register ein (eine öffentliche Liste/Datenbank der RPs und ihrer genehmigten Attributnutzungen). Dieser Schritt liegt außerhalb von RPCT, ist aber wichtiger Kontext: Das RPRC muss später mit diesen Angaben übereinstimmen.

2. Zertifikatsausstellung: Die RP bezieht ein RPAC (und optional ein RPRC) von den jeweiligen CAs. Stellt die CA diese Zertifikate aus, übermittelt sie sie gleichzeitig an das RPCT-Log. Die Übermittlung könnte die Kennung der RP als Log-Schlüssel und die Zertifikatsdaten als Wert enthalten.

3. Protokollierung und Nachweis: Der RPCT Logger fügt diesen neuen Zertifikatseintrag an. Er aktualisiert seine interne Datenstruktur und berechnet einen neuen Wurzel-Digest. Das Log gibt einen Inklusionsnachweis an die CA zurück. Die CA kann diesen Nachweis zusammen mit dem Zertifikat an die RP weiterleiten. Alternativ könnte das Log die Nachweise auch direkt an Monitore senden, die die Kennung dieser RP abonniert haben. Ab diesem Zeitpunkt ist das Zertifikat im Log öffentlich sichtbar (für jeden prüfbar), und es gibt einen kryptografischen Nachweis seiner Aufnahme.

4. Credential-Präsentation: Möchte ein Nutzer auf den Dienst der RP zugreifen, legt die RP der Wallet des Nutzers ihr RPAC und ihr RPRC vor (wahrscheinlich über ein Protokoll wie OpenID for Verifiable Presentations). Mit RPCT liefert die RP nun zusätzlich den aktuellen Inklusionsnachweis für diese Zertifikate (und gegebenenfalls einen Konsistenznachweis, um die Entwicklung des Log-Zustands zu zeigen). Der Nachweis überzeugt die Wallet, dass die Zertifikate in der globalen Zusammenfassung protokolliert wurden, der sie vertraut. Die Wallet verwendet den öffentlichen Schlüssel des Logs, um

den Nachweis gegen eine vertrauenswürdige Zusammenfassung zu verifizieren (die sie zwischengespeichert haben oder aus einem Gossip-Netzwerk oder einer regelmäßigen Aktualisierung beziehen kann).

5. Wallet-Verifikation: Die Wallet prüft: (a) die Signatur des RPAC (wurde es von einer vertrauenswürdigen CA ausgestellt und ist es weder abgelaufen noch widerrufen?), (b) den Inhalt des RPRC (deckt es die angeforderten Attribute ab?) und nun auch (c) die RPCT-Nachweise: Sie stellen sicher, dass RPAC/RPRC im Transparenz-Log enthalten und nicht widerrufen wurden. Ist alles in Ordnung, fragt die Wallet den Nutzer nach der Einwilligung zur Freigabe der angeforderten Daten. Schlägt etwas fehl (z. B. ist der Inklusionsnachweis ungültig oder zeigt das Log, dass das RPRC widerrufen wurde), lehnt die Wallet ab oder warnt den Nutzer.

6. Monitoring und Warnungen: Während sich Log-Einträge ansammeln, können Monitore Richtlinienverstöße erkennen. Hat das RPRC einer RP beispielsweise einen Zweck, der nicht mit dem offiziellen Register übereinstimmt (ein Verstoß gegen die Konsistenzanforderung), könnte ein Monitor die Behörden benachrichtigen, damit diese die RP oder CA untersuchen. Wird ein Zertifikat widerrufen, zeigt das Log seine Entfernung bei der nächsten Aktualisierung; ein Monitor könnte Relying Parties (oder Wallets per Software-Update) über Massenwiderrufe oder verdächtige Muster informieren (etwa viele Widerrufe einer einzelnen CA, was auf eine Kompromittierung dieser CA hindeuten könnte).

7. Widerrufsbehandlung: Muss das Zertifikat einer RP widerrufen werden (etwa weil die RP ihre Akkreditierung verloren hat oder eine Sicherheitsverletzung vorlag), leitet die CA einen Widerruf ein. In RPCT übermittelt die CA ein „Entfernungs“-Update für den Eintrag dieser RP an das RPCT-Log. Das Log fügt einen neuen Datensatz für den Schlüssel dieser RP hinzu, der anzeigt, dass das Zertifikat nicht mehr aktiv ist (Abschnitt 4 erläutert, wie dies ohne Löschen der Historie geschieht). Wallets und Monitore nutzen das Log dann, um kryptografische Gewissheit über den Widerrufsstatus zu erhalten (das Fehlen des Zertifikats im neuesten Zustand dient als Nachweis des Widerrufs).

Bindung von RPACs und RPRCs: Ein wichtiges Merkmal von RPCT ist, dass es erzwingen kann, dass die beiden Zertifikate einer RP im Log kryptografisch verknüpft sind. Da RPAC und RPRC einer RP denselben Kennungsschlüssel im Log (die eindeutige ID der RP) teilen, kann jedes Update des Eintrags dieser RP beide Zertifikate gemeinsam als Menge enthalten. Das bedeutet, dass das Log sicherstellen kann, dass ein gültiges RPAC und ein dazugehöriges RPRC zusammen gesehen werden. Versucht ein Betrüger, ein RPRC für ein nicht existierendes RPAC zu protokollieren (oder umgekehrt), würde das auffallen. Die Wallet kann außerdem die aktuelle Zertifikatsmenge der RP aus dem Log abrufen, um die Konsistenz zu prüfen (z. B. sicherstellen, dass die RP, die ein bestimmtes RPRC vorlegt, auch ein passendes RPAC protokolliert hat).

4. Authentifizierte Datenstruktur: Merkle Patricia Trie und Hash pro RP

Das Herzstück von RPCT ist eine authentifizierte Datenstruktur (dargestellt in Abbildung 3), die sich vom herkömmlichen CT-Merkle-Baum unterscheidet. Das Entwurfsziel war, chronologisches, nur anfügendes Logging und effiziente Abfragen pro RP zu unterstützen. Die Lösung ist ein Merkle Patricia Trie (MPT) [7], der nach RP-Kennungen indiziert ist, kombiniert mit individuellen Hash-Ketten pro RP.

Schlüsselindizierter Merkle Trie: Statt eines großen Zeitleisten-Baums aller Zertifikate organisiert RPCT das Log als Wörterbuch (Map) von `RP ID → certificates`. Jeder RP wird ein eindeutiger Schlüssel zugewiesen (das könnte etwa eine gehashte Kennung wie eine EORI-Nummer, ein Domainname oder eine länderbezogene ID sein), der ihre Position im Trie bestimmt. Alle Zertifikate dieser RP (sowohl RPAC als auch RPRC sowie historische Versionen) liegen unter dem Knoten dieses Schlüssels. Die Verwendung eines Patricia Trie (eines komprimierten Präfixbaums) stellt sicher, dass dieser Schlüsselraum effizient abgebildet wird: Gemeinsame Schlüsselpräfixe (z. B. derselbe Ländercode) teilen sich Zweige, und die Blätter liegen je nach Schlüssel in unterschiedlicher Tiefe. Diese Struktur ähnelt dem Ansatz von Key Transparency (Zuordnung von Nutzer-IDs zu Schlüsseln), hier ist der „Nutzer" jedoch eine RP.

Hash-Kette pro RP (chronologische Historie): Jedes Blatt im Trie entspricht einer RP und speichert eine kryptografische Hash-Kette, die die Abfolge der Zertifikatsereignisse dieser RP festhält. Im Einzelnen: Erhält eine RP erstmals ein Zertifikat, wird ein Hash `h1` gebildet, indem (i) ein Zeitstempel oder eine Epoche `T1`, (ii) die Menge der aktiven Zertifikatswerte `v1` (etwa das RPAC oder RPRC oder beides) und (iii) ein initialer vorheriger Hash `h0` (der ein Nullwert sein kann) zusammen gehasht werden.

Dieses `h1` wird zum „Kopf" der Kette dieser RP. Aktualisiert die RP später ihr Zertifikat oder erhält sie ein neues (oder wird eines widerrufen), wird ein neuer Hash `h2 = Hash(T2 ‖ v2 ‖ h1)` berechnet, der auf den vorherigen Kopf verweist. Mit der Zeit entsteht eine Kette `h0 → h1 → h2 → ... → ht`, in der jedes Glied den Zertifikatszustand dieser RP in der jeweiligen Epoche unveränderlich festhält. Der Wert `vi` in jedem Schritt lässt sich als die aktuelle Zertifikatsmenge dieser RP nach dem jeweiligen Update verstehen (zum Beispiel könnte `v1` zunächst das erste RPAC enthalten; wird später ein RPRC ausgestellt, enthält `v2` {RPAC, RPRC}; wird das RPAC erneuert, aktualisiert `v3` dieses usw.). Da jeder neue Hash den vorherigen Hash einschließt, entsteht für diese RP eine nur anfügbare Historie.

Globales Log als Trie der Köpfe: Der jeweils aktuelle Kettenkopf `ht` jeder RP wird im Merkle Patricia Trie an der Position für den Schlüssel dieser RP abgelegt. Anschließend wird die Merkle-Wurzel dieses gesamten Tries berechnet. Diese Wurzel ist die globale Log-Zusammenfassung (`R`), die den Zustand aller RP-Einträge in der aktuellen Epoche widerspiegelt. Ändert sich der Eintrag einer RP (ein neues Zertifikat oder ein Widerruf), ändert sich ihr Blatt-Hash und damit der Wurzel-Hash. Die Wurzel dient als Festschreibung der aktuellen Zertifikatszustände aller RPs. Das Log signiert diese Wurzel regelmäßig (wie einen Signed Tree Head in CT).

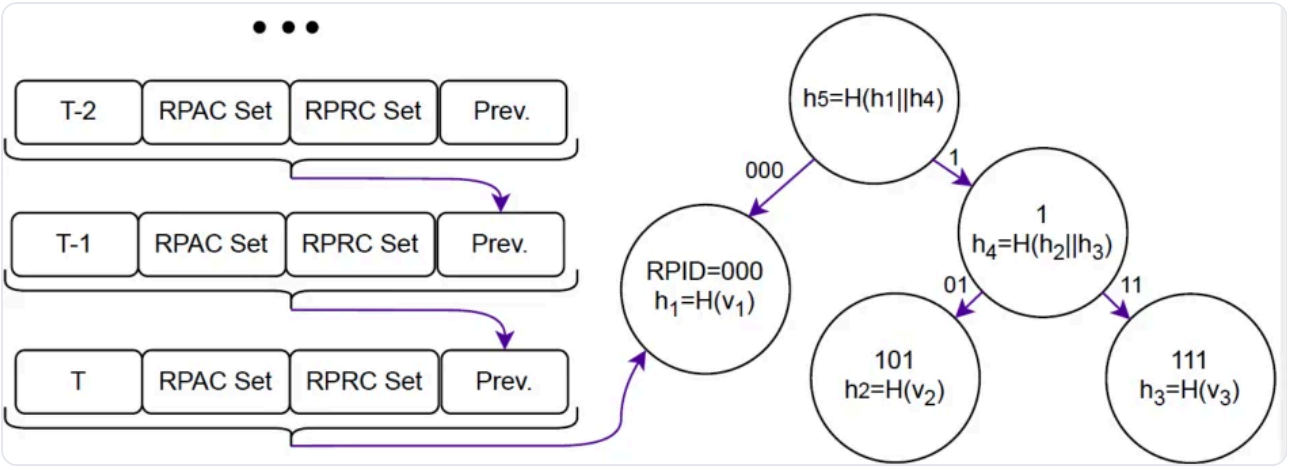


Abbildung 3: Beispiel für die authentifizierte RPCT-Datenstruktur: Angenommen, die RP „Alice's Service" registriert sich zunächst und erhält ein RPAC. Der Trie des Logs erhält ein Blatt für Alices RPID mit dem Hash-Kettenkopf `h1 = Hash(T1` ‖ `{RPAC}` ‖ ⊥ `)`. Die globale Wurzel `R1` deckt nun Alices Eintrag ab. Später erhält Alice ein RPRC, das die Attribute „E-Mail, Name" auflistet. Die CA protokolliert dies als Update: Berechnet wird `h2 = Hash(T2` ‖ `{RPAC, RPRC}` ‖ `h1)`. Das Blatt für Alice wird auf `h2` aktualisiert. Die globale Wurzel ändert sich zu `R2`. Fragt jemand das Log zu diesem Zeitpunkt nach Alices RPID ab, findet er `h2` und kann (mit einem Nachweis) verifizieren, dass `h2` in `R2` enthalten ist. Widerruft die CA anschließend Alices RPRC (etwa weil Alice ihre Datennutzung geändert hat und das alte RPRC ungültig wird), protokolliert die CA ein neues Update: `h3 = Hash(T3` ‖ `{RPAC}` ‖ `h2)`, wodurch das RPRC effektiv aus der aktuellen Menge entfernt wird. Der neueste Kopf spiegelt nun nur noch das RPAC wider, und `R3` ist die neue Wurzel. Wichtig: Das alte RPRC ist nicht aus der Historie verschwunden, es ist nur nicht mehr in der aktuellen Menge. Jeder kann die Kette `[h1,` `h2, h3]` einsehen und sehen, dass das RPRC zum Zeitpunkt `T2` hinzugefügt und zum Zeitpunkt `T3` entfernt wurde.

Der MPT ermöglicht eine effiziente Erzeugung von Nachweisen für Abfragen. Muss eine Wallet die Zertifikate einer RP verifizieren, kann sie einen Abfragenachweis für den Schlüssel dieser RP anfordern. Das Log gibt den aktuellen Wert (die Menge der aktiven Zertifikate oder den Kopf-Hash) und einen Merkle-Nachweispfad im Trie bis zur Wurzel zurück. Dieser Nachweis ist logarithmisch in der Anzahl der Schlüssel, nicht in der Anzahl aller Einträge, was in einem System kontinentalen Maßstabs entscheidend ist. Gibt es beispielsweise 1 Million RPs, könnte der Nachweis das Durchlaufen eines Baums mit etwa 20 Ebenen umfassen, statt potenziell Millionen unzusammenhängender Einträge einzuschließen. Der Patricia Trie ist besonders nützlich, wenn die Schlüssel strukturell zusammenhängen (wie Domainnamen oder Ländercodes). Unterschiedliche Kodierungsstrategien für Schlüssel (Hashing gegenüber semantischen Segmenten) lassen sich einsetzen, um Baumtiefe und Nachweisgröße auszubalancieren.

Bereitgestellte Nachweisarten: Das RPCT-Log unterstützt mehrere Abfragen:

Inklusionsnachweis: Für ein bestimmtes Zertifikat (genauer: eine bestimmte RP und einen Zertifikats-Fingerabdruck) kann das Log nachweisen, dass es in einer bestimmten Epochen-Zusammenfassung im Log enthalten ist. Da pro RP Mengen gespeichert werden, könnte ein Inklusionsnachweis in der Praxis zeigen, dass „für den RP-Schlüssel `k` der Wert `v` (der das Zertifikat enthält) in der Menge beim Blatt-Hash `ht`" enthalten ist", und den Pfad im Trie liefern.

Konsistenznachweis: Für zwei Wurzel-Digests `Ri` und `Rj` (`i<j`) beweist ein Konsistenznachweis, dass der spätere Zustand alle Einträge des früheren Zustands enthält (d. h., dass das Log nur gewachsen ist). Im Design Trie-von-Ketten stellt ein Konsistenznachweis sicher, dass kein RP-Blatt, das bei `Ri` existierte, bis `Rj`

entfernt wurde – formal, dass jeder Blatt-Hash im Zustand `i` auch im Zustand `j` noch vorhanden ist, möglicherweise mit aktualisierten Werten, und kein Blatt gelöscht wurde. Das entspricht den Konsistenznachweisen von CT zwischen STHs.

Abfrage-/Abwesenheitsnachweis: Fragt eine Wallet „Existiert RP X im Log (ist sie registriert)?“, kann das Log mit der Trie-Struktur einen Abwesenheitsnachweis führen. Der Nachweis besteht aus dem Pfad, an dem der Schlüssel gestanden hätte, und zeigt eine Lücke, in der kein solcher Schlüssel existiert. Damit ist bewiesen, dass die RP (noch) nicht im Log steht (was „nicht registriert“ bedeuten könnte).

Historiennachweis: Man kann auch die gesamte Historie einer RP anfordern. Das Log kann die Folge aller Werte `{v0, v1, ..., vt}` zurückgeben, die dieser RP jemals zugeordnet waren, zusammen mit einem Nachweis, dass diese Folge vollständig ist. Da jedes Kettenglied den vorherigen Hash einbezieht, kann ein Verifizierer die Integrität der Historie prüfen. Das ist nützlich für Auditoren oder Monitore, um zu untersuchen, wie sich die Berechtigungen einer RP im Lauf der Zeit verändert haben.

Vergleich der Datenstrukturen (CT vs. RPCT): Im Standard-CT ist jedes neue Zertifikat ein neues Blatt in einem Merkle-Baum, und Monitore müssen Blätter verfolgen, indem sie den ganzen Baum durchsuchen. Im MPT+Ketten-Ansatz von RPCT ist das „Blatt“ pro RP angelegt, und Updates gehen in die Kette dieses Blatts. Das ergibt kleinere Nachweisgrößen und lokalisiertes Monitoring: Um eine bestimmte RP oder eine Gruppe von RPs zu überwachen, kann man nur diese Schlüssel abfragen. Tabelle 1 zeigt, dass diese Struktur den Aufwand sowohl für Inklusionsnachweise als auch für Konsistenzprüfungen im EU-Maßstab deutlich reduziert. Sie behandelt außerdem den Widerruf von Natur aus: Das Entfernen eines Zertifikats erzeugt lediglich ein neues Kettenelement, statt ein Blatt als widerrufen markieren oder eine separate Widerrufsliste führen zu müssen.

Tabelle 1 Nachweiskomplexitäten CT vs. RPCT

Operation	**CT**	**RPCT**
Abfrage nach Schlüssel	Nicht nativ unterstützt, Durchsuchen O(N)	Unterstützt O(logN)
Inklusionsnachweis	O(logN) Hashes	O(logN) Hashes + 1 Kettenglied
Konsistenznachweis	O(logN) Hashes	O(logN) pro Update
Historiennachweis	Nicht nativ (mehrere Inklusionsnachweise nötig)	Unterstützt über die Kette (O(L) Glieder bei L Updates)

Widerrufstransparenz und Konsistenz: Eine wesentliche Verbesserung von RPCT gegenüber reinem CT ist die Unterstützung der Widerrufstransparenz. Mit anderen Worten: Jeder kann über das Transparenz-Log nicht nur prüfen, dass ein Zertifikat protokolliert wurde, sondern auch, ob es noch gültig ist oder widerrufen wurde.

Protokollierung von Widerrufen: Widerruft eine CA in RPCT ein RPAC oder RPRC, übermittelt sie ein Update an das Log, genau wie bei einem Ausstellungsereignis. Das Update für einen Widerruf bedeutet im Wesentlichen „Entferne Zertifikat X aus der aktiven Menge von RP Y“. Technisch berechnet die CA einen neuen Wert für das Blatt dieser RP (war der vorherige Wert z. B. `{Cert1, Cert2}` und wird `Cert2`

widerrufen, ist der neue Wert `vnew = {Cert1},` ohne `Cert2`). Dieser neue Wert wird mit dem vorherigen Kettenkopf gehasht, wodurch ein neuer Kopf für die Kette dieser RP entsteht, und die globale Wurzel wird aktualisiert. Da der vorherige Kopf in der Kette verknüpft bleibt, ist die Tatsache, dass Cert2 früher vorhanden war, in der Historie weiterhin prüfbar, eine aktuelle Abfrage zeigt es jedoch als entfernt. Ein spezieller „widerrufen"-Marker oder eine separate Widerrufsliste ist nicht nötig : Der Zustand des Logs enthält das Zertifikat schlicht nicht mehr.

Nachweis von Widerruf (oder Nicht-Widerruf): Für eine Wallet läuft der Nachweis, dass ein Zertifikat nicht widerrufen wurde, darauf hinaus, einen Inklusionsnachweis im neuesten Zustand zu erhalten. Ist das Zertifikat noch enthalten, ist es nicht widerrufen; ist es nicht enthalten, wurde es widerrufen (oder hat nie existiert). Das Log kann für ein Zertifikat in der neuesten Epoche einen Abwesenheitsnachweis erzeugen, der zusammen mit einem historischen Inklusionsnachweis einen Client überzeugen kann, dass „Zertifikat X zum Zeitpunkt T protokolliert wurde und jetzt nicht mehr erscheint, also zwischen T und jetzt widerrufen wurde". Umgekehrt ist ein Inklusionsnachweis im neuesten Zustand ein Beleg dafür, dass das Zertifikat noch gültig ist. Das entspricht einem OCSP-Status „good", der aber durch die Integrität des Transparenz-Logs gestützt wird. Wichtig: Monitore könnten auch Widerrufsmuster beobachten: Widerruft etwa eine bestimmte RP oder CA plötzlich viele RPRCs, könnte das auf eine Häufung von Fehlverhalten hindeuten, die eine Untersuchung rechtfertigt.

Privatsphäre-wahrende Nachweise und Offline-Verifikation: Eines der herausragenden Merkmale von RPCT ist, dass es die Offline-Verifikation von Nachweisen ermöglicht, was für die Privatsphäre der Nutzer entscheidend ist. Im herkömmlichen CT musste ein Browser oft ein Log oder einen Monitor kontaktieren, um sicherzustellen, dass der SCT eines Zertifikats im Log enthalten ist. In RPCT kann die Wallet dagegen den Zertifikatsstatus einer RP lokal verifizieren, ohne zum Zeitpunkt der Präsentation das Log abzufragen.

Auslieferung von Nachweispaketen: Erhält eine RP ihre Zertifikate und werden diese protokolliert, kann die RP (oder die CA) auch die benötigten Nachweise aus dem Log abrufen. Nach der Protokollierung könnte die CA der RP beispielsweise einen Inklusionsnachweis für RPRC und RPAC gegen die neueste (oder eine kürzlich aktuelle) Wurzel bereitstellen. Dieser Nachweis könnte Folgendes enthalten: den Merkle-Pfad im Trie für den Schlüssel der RP und den neuesten, vom Log signierten Wurzel-Hash. Die CA könnte außerdem einen Konsistenznachweis liefern, der diese Wurzel mit einem früheren Wurzel-Hash verknüpft, dem die Wallet vertraut (falls die Wallet nur eine ältere Referenz hat). Alternativ könnten die Wallet-Software oder das Betriebssystem regelmäßig aktualisierte Log-Zusammenfassungen abrufen (ähnlich wie Browser eine Liste vertrauenswürdiger CT-Log-Wurzeln aktualisieren oder Gossip-Protokolle nutzen, um STHs zu erfahren). Entscheidend ist, dass zu dem Zeitpunkt, an dem ein Nutzer einer RP Credentials präsentiert, die nötige Log-Zusammenfassung und die Nachweise ohne Live-Abfrage bereitgestellt werden können.

Offline-Verifikation: Die Wallet nutzt den Inklusionsnachweis und die bekannte Log-Zusammenfassung, um zu verifizieren, dass das Zertifikat der RP tatsächlich protokolliert wurde. Da die Log-Zusammenfassung global bezeugbar ist (und idealerweise das Gerät des Nutzers sie besitzt oder auf privatsphäre-wahrende Weise erhalten kann, z. B. per Broadcast oder in behördlichen Bekanntmachungen), muss die Wallet das Log für diese eine RP nicht kontaktieren. Sie berechnet die Hashes einfach aus dem Nachweis neu: Ausgehend vom Blatt der RP (das den Hash des Zertifikats in der Wertemenge enthält) hasht sie durch die bereitgestellten Geschwister-Hashes aufwärts und prüft, ob das Ergebnis mit der vertrauenswürdigen Wurzel übereinstimmt. Stimmt es überein, ist bestätigt, dass das Zertifikat im Log vorhanden ist. Wäre das

Zertifikat widerrufen worden, würde der Inklusionsnachweis für die aktuelle Wurzel fehlschlagen (das Zertifikat würde nicht mehr in der Wertemenge dieses Blatts erscheinen). In diesem Fall könnte die RP keinen gültigen Inklusionsnachweis für den neuesten Zustand vorlegen.

Das ist wichtig, weil eIDAS 2 ausdrücklich verlangt, dass die Nutzung der Wallet keine Nachverfolgung oder Profilbildung von Nutzern durch externe Parteien erlauben darf. Müsste die Wallet bei jeder Nutzung ein staatliches Log fragen „Ist das Zertifikat von RP X gültig?“, könnte ein Beobachter des Log-Verkehrs oder das Log selbst daraus schließen, dass man mit RP X interagiert. RPCT vermeidet dies durch ein Modell, in dem entweder: die Wallet eine zwischengespeicherte Kopie des gesamten Log-Zustands besitzt (unwahrscheinlich, zu groß) oder, realistischer, die Wallet regelmäßig (oder über einen vertrauenswürdigen Kanal) die neueste Log-Wurzel erhält und die RP den spezifischen Nachweis liefert. Da die RP ohnehin die Gegenstelle der Verbindung ist, entsteht durch den Bezug des Nachweises von der RP kein neues Privatsphäre-Leck. Der von der RP bereitgestellte Nachweis kann zudem geprüft werden, ohne jemand anderen zu kontaktieren.

5. Diskussion

5.1. Einhaltung der eIDAS-2-Anforderungen

Die eIDAS-2-Verordnung und ihre Entwürfe der Durchführungsrechtsakte legen konkrete Anforderungen an die Aufsicht und Sicherheit von Relying Parties fest. RPCT wurde unter Berücksichtigung dieser Anforderungen entworfen und erfüllt sie. Zu den adressierten zentralen regulatorischen Anforderungen gehören:

Artikel 5b(1–2) – RP-Registrierung und Zweckangabe: RPs müssen sich registrieren und ihre beabsichtigte Wallet-Nutzung und ihren Datenbedarf gegenüber Behörden angeben.

Konformität: Die Verwendung von RPRCs in RPCT unterstützt dies unmittelbar. Jedes RPRC ist im Wesentlichen ein signiertes Artefakt dieser Angabe, und durch die öffentliche Protokollierung macht RPCT die Registrierungsinformationen transparent (dies entspricht der Anforderung, dass Mitgliedstaaten eine Liste oder ein Register der RPs und ihrer zulässigen Daten veröffentlichen). RPCT kann sogar ein stets aktuelles Log aller autorisierten RPs und ihrer Umfänge bereitstellen.

Artikel 5b(8) – RP-Authentifizierung (Access Certificates): Nutzer sollen Daten nur an RPs senden, die sich ordnungsgemäß identifizieren (daher werden RPACs verlangt).

Konformität: RPCT protokolliert alle RPACs und stellt so sicher, dass es eine öffentliche Aufzeichnung darüber gibt, welche RPs gültige Authentifizierungszertifikate haben. Ein unseriöser Dienst ohne protokolliertes RPAC wäre erkennbar. Wallets setzen dies durch, indem sie nur fortfahren, wenn ein RPAC einen gültigen Log-Nachweis hat, und erfüllen damit die Anforderung, dass „Wallets einer RP ohne gültiges Access Certificate keine Daten präsentieren dürfen“.

Artikel 5b(3) – Rechtmäßigkeit von Datenanfragen: RPs „dürfen keine anderen als die im Register angegebenen Daten anfordern“.

Konformität: Genau das erzwingen RPRCs und RPCT. Das RPRC ist eine konkrete Liste der erlaubten Attribute (der „angegebenen“ Daten), und die Wallet kann sie mit der tatsächlichen Anfrage vergleichen. Da das Log öffentlich ist, können Behörden außerdem verifizieren, dass RPRCs mit den registrierten Zwecken

übereinstimmen. Fordert eine RP jemals mehr Daten an (Over-Asking), würde die Abweichung entweder von der Wallet erkannt (wenn Wallets diese Prüfung mit Hilfe des Logs umsetzen) oder im Nachhinein von Monitoren.

Artikel 5a(14) und Erwägungsgrund 32 – Nutzerkontrolle und Einwilligung: Nutzer müssen die volle Kontrolle haben und der Datenweitergabe ausdrücklich zustimmen, und es dürfen keine versteckten Datenübermittlungen stattfinden.

Konformität: RPCT unterstützt dies indirekt, indem es Vertrauen in das aufbaut, was die Wallet dem Nutzer mitteilt. Zeigt eine Wallet „RP X fordert Ihr Geburtsdatum und Ihre E-Mail an (wie registriert)“, kann der Nutzer sicher sein, dass diese Anfrage in einem offiziellen Verfahren geprüft wurde (wegen der Transparenz der RPRCs). Versuchte eine RP, Daten ohne ordnungsgemäße Registrierung oder über ein fehlerhaft ausgestelltes Zertifikat zu sammeln, würde das Transparenzsystem dies wahrscheinlich erkennen und Nutzer so vor Missbrauch hinter den Kulissen schützen.

Artikel 5a(5)(b) und 5a(16)(a) – Privatsphäre (kein Tracking oder Verknüpfen): Keine Stelle darf die Credential-Nutzung von Nutzern ohne Erlaubnis nachverfolgen.

Konformität: Wie besprochen stellen die Offline-Nachweise von RPCT sicher, dass der Verifikationsvorgang der Wallet keine Nutzeraktivität an Logs oder Dritte verrät. Anders als beim naiven CT, bei dem jede Zertifikatsprüfung ein Log anpingen könnte, kann die Wallet bei RPCT während der Nutzung im Netzwerk stumm bleiben.

5.2. Überlegungen zum Einsatz

Dieser Abschnitt listet einige zentrale Überlegungen zur Implementierung und zum Einsatz von RP Certificate Transparency im EU-weiten Maßstab in Bezug auf Infrastruktur, Leistung und Governance auf.

Log-Infrastruktur: Wer betreibt die RPCT-Logs? Zu den Optionen gehören ein zentraler, von der EU betriebener Log-Dienst oder mehrere nationale Logs (jeder Mitgliedstaat betreibt eines für seine RPs). Ein dezentrales Modell mit mehreren unabhängigen Logs ist wahrscheinlich ratsam (so wie CT mehrere Logs nutzt). Das bietet Redundanz und verhindert Single Points of Failure oder des Vertrauens. Werden mehrere Logs verwendet, müssen Standards festlegen, wie CAs Logs auswählen und wie Wallets Nachweisen aus ihnen vertrauen (z. B. durch Führen einer Liste vertrauenswürdiger RPCT-Logs).

Skalierbarkeit und Leistung: Die Zahl der RPs in Europa könnte groß sein (auch weil mittlere und kleine Unternehmen EUDI-Wallets anbinden). Die gewählte Datenstruktur (MPT) wurde auf Skalierbarkeit geprüft: Nachweise bleiben logarithmisch in der Anzahl der RPs, und Update-Operationen sind effizient. Dennoch ist der Betrieb eines Log-Servers mit hohem Durchsatz eine anspruchsvolle Aufgabe. Das Log sollte Spitzen bei Zertifikatsregistrierungen bewältigen (z. B. wenn eine neue regulatorische Frist ansteht und viele RPs sich gleichzeitig registrieren).

Integration mit Wallets und RPs: Wallet-Software (die voraussichtlich von verschiedenen Anbietern im Rahmen des EUDI-Rahmenwerks bereitgestellt wird) muss die Logik zur Verarbeitung von RPCT-Nachweisen implementieren. Das heißt, Wallet-Entwicklern müssen Bibliotheken oder Standards zur Verfügung gestellt werden, um Merkle-Nachweise zu verifizieren, die RPCT-Datenstruktur zu verstehen und die vertrauenswürdige Log-Wurzel abzurufen. Außerdem ist ein UX-Design nötig, um Warnungen anzuzeigen, wenn etwas nicht stimmt (etwa ein nicht protokolliertes Zertifikat oder eine RP, die zu viele

Daten anfordert). RPs wiederum müssen Nachweise beziehen und bereitstellen. Das ließe sich durch Middleware abstrahieren: Eine RP könnte etwa ein SDK verwenden, das automatisch den neuesten Inklusionsnachweis von einem Transparenz-Log-Dienst abruft. Denkbar wäre ein „Transparency Proof Token“, das die RP in ihre OIDC- oder Präsentationsanfrage an die Wallet aufnimmt. All dies sollte standardisiert werden, um Interoperabilität sicherzustellen (voraussichtlich über das EUDI Architecture Reference Framework).

Vertrauensmodell und Governance: RPCT führt eine neue kritische Komponente ein: den Log-Betreiber. Er muss zwar nicht für die Korrektheit vertrauenswürdig sein (da Fehlverhalten kryptografisch erkennbar ist), Verfügbarkeit und zeitnaher Betrieb sind jedoch wichtig. Die öffentlichen Schlüssel der Logs müssen an alle Wallets verteilt werden, damit diese Signaturen auf Wurzeln und Nachweisen verifizieren können.

6. Schlussfolgerungen

eIDAS 2 führt einen strukturierten Ansatz zur Authentifizierung und Autorisierung von Relying Parties (RPs) ein, und zwar durch RP Registration Certificates (RPRCs) und RP Access Certificates (RPACs). Das Fehlen öffentlicher Überprüfbarkeit im Lebenszyklus der Zertifikate birgt jedoch Risiken durch Fehlausstellung, stille Unterdrückung von Widerrufen und das Anfordern übermäßig vieler personenbezogener Daten (Over-Asking).

RP Certificate Transparency (RPCT) bietet eine kryptografisch überprüfbare Infrastruktur, um diesen Risiken zu begegnen. Es protokolliert RPACs und RPRCs in einer öffentlichen, nur anfügbaren Merkle-basierten Struktur, die eine effiziente Erzeugung und Verifikation von Inklusions-, Widerrufs- und historischen Konsistenznachweisen ermöglicht. Jede RP wird in einem Merkle Patricia Trie abgebildet, wobei Hash-Ketten pro RP die Abfolge der Zertifikatsereignisse festhalten. Das erlaubt skalierbare Audits und Offline-Verifizierbarkeit durch Wallet-Anwendungen.

Das System erfüllt zentrale regulatorische Ziele: Es stellt sicher, dass RPs ihre deklarierten Berechtigungen nicht überschreiten können (Artikel 5b), unterstützt die Durchsetzung zur Laufzeit durch Wallets und ermöglicht unabhängiges Monitoring. RPCT bietet eine grundlegende Vertrauensschicht, indem es RP-Berechtigungen überprüfbar und das Over-Asking von RPs im großen Maßstab prüfbar macht.

Diese Arbeit führte zu einem Paper, das bei ARES 2025 im Rahmen des International Workshop on Emerging Digital Identities (EdID) angenommen wurde.

Referenzen

[1] “EU Digital Identity Wallet Home - EU Digital Identity Wallet -.” Accessed: Nov. 11, 2024. [Online]. Available: https://ec.europa.eu/digital-buildingblocks/sites/display/EUDIGITALIDENTITYWALLET/EU+Digital+Identity+Wallet+Home

[2] “Certificate Transparency : Certificate Transparency.” Accessed: Nov. 11, 2024. [Online]. Available: https://certificate.transparency.dev/

[3] E. Fasllija and S. More, “Guardians of the Registry: Certificate Transparency for Relying Party Authorization in eIDAS 2,” Aug. 11, 2025, *Springer* . Accessed: Jul. 30, 2025. [Online]. Available: https://tugraz.elsevierpure.com/en/publications/guardians-of-the-registry-certificatetransparency-for-

relying-pa

[4] “Deploying key transparency at WhatsApp - Engineering at Meta.” Accessed: Nov. 11, 2024. [Online]. Available: https://engineering.fb.com/2023/04/13/security/whatsapp-keytransparency/

[5] H. Malvai *et al.*, “Parakeet: Practical Key Transparency for End-to-End Encrypted Messaging,” *Cryptology ePrint Archive*, 2023, doi: 10.14722/NDSS.2023.24545.

[6] “Regulation - EU - 2024/1183 - EN - EUR-Lex.” Accessed: Jul. 30, 2025. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2024/1183

[7] “What are Patricia Merkle Tries? | Alchemy Docs.” Accessed: Jul. 30, 2025. [Online]. Available: https://www.alchemy.com/docs/patricia-merkle-tries