Quantensichere Verschlüsselung in Gruppen mit dem MLS-Protokoll

Stefan Schumacher

cryptomancer.de

Quantenkrypto-Symposium ITZBund Ilmenau

$ Id: MLS-Protokoll-Input.tex,v 1.24 2026/04/29 09:14:17 stefan Exp $

cryptomancer.de

Inhaltsverzeichnis

1. Einführung
2. Matrix - 3DH-KEX und DoubleRatchet
3. Messaging Layer Security Protocol
   - Fähigkeiten
   - Architektur
   - Ratschenbaum
4. PQC

cryptomancer.de



- Robotron KC85/3 Mitte der 80er Jah
- Geek, Nerd, Hacker
- Kryptographie seit 30 Jahren
- einige Jahre NetBSD-Entwickler
- ehem. Sicherheitsforscher u.a. Fach-Didaktik der Kryptographie
- jetzt Sicherheitsarchitekt bei einem öffentlichen IT-Dienstleister

cryptomancer.de

Definition (Outrage as a Svc @OaaSvc)

Science is awesome. You aren't doing science in infosec. Why not? Seems to be the overriding message of @0xKaishakunin #AusCERT2014

Glossar

- CLT2023: Das Matrix-Protokoll näher betrachtet – Die Kryptographie-Architektur
- https://cryptomancer.de/posts/202303matrix/
-
- Glossar samt Formelzeichen und Literatur im Handout
- https://www.cryptomancer.de/messaginglayer/

Inhaltsverzeichnis

1 Einführung

2 Matrix - 3DH-KEX und DoubleRatchet

3 Messaging Layer Security Protocol
- Fähigkeiten
- Architektur
- Ratschenbaum

4 PQC

cryptomancer.de

Matrix

- Application-Layer-Kommunikationsprotokoll für föderierte Echtzeitkommunikation
- sicheres Verbreiten und Speichern von JSON-Nachrichten
- geteilter Zustand *nicht* direkte Nachrichten zwischen Partnern
- HTTP und WebRTC
- VoIP, IoT, IM
- Ziel: generisches Messaging- und Datensynchronisationsprotkokoll

Glossar I

(Perfect) Forward Secrecy (PFS) Sitzungsschlüssel werden so vereinbart, dass nach einem Einbruch *vergangene* Sitzungsschlüssel nicht gebrochen werden können (break-backwards protection)

Break-in recovery Sitzungsschlüssel werden so vereinbart, dass nach einem Einbruch auch *zukünftige* Sitzungsschlüssel nicht gebrochen werden können (break-forward protection)

Zurechenbarkeit Nachrichten sind dem zuzuordnen, der sie abgeschickt hat

Verbindlichkeit Sender kann Urheberschaft gesendeter Nachrichten nicht abstreiten

cryptomancer.de

Glossar II

plausible Deniability Gegenteil zu Zurechenbarkeit && Verbindlichkeit, Alice und Bob können sich sicher sein, dass sie mit Bob bzw. Alice kommunizieren, aber Mallory kann nicht beweisen, dass Alice und Bob kommuniziert haben, im Englischen auch Off-the-Record genannt

Malleability Formbarkeit kryptographischer Nachrichten, d.h. Zurechenbarkeit && Verbindlichkeit können ausgehebelt werden

Device im Matrix-Kontext *alle* End-Geräte eines Nutzers

M: Message/Nachricht; K: Key pub: public $priv$: private sig: signature enc: decryption

cryptomancer.de

Triple Diffie-Hellman Key Exchange

- 3DH-KEX: Alice und Bob leiten ephemerale Schlüssel (a,b) ab und führen 3 Handshakes zwischen den ID-Schlüsseln und ephemeralen Schlüsseln
- Danach werden 3 geteilte Geheimnisse abgeleitet: S^{aB}, S^{Ab}, S^{ab}
- Algorithmische und Protokoll-Komplexität reduziert, Formbarkeit erhöht, Nutzlast verringert, Forward Secrecy beibehalten
- 3DH-KEX: Nur zum Aufbau einer Session, danach nicht mehr verwendet

Matrix und große Räume

- für jede Nachricht wird ein eigener neuer ephemeraler Schlüssel aus dem Session-Schlüsselpaar von Alice und Bob abgeleitet
- private Schlüssel nach der Nachricht vernichtet
- Jeder Sender hat eine eigene Ratsche (*outbound session* um Nachrichten an einen Raum zu verschlüsseln
- Raum-Mitglieder können die Ratschen-Historie lokal speichern, um die Nachrichten-Historie lokal zu speichern
- Kann vorwärts gespult werden, damit neue Mitglieder alte Nachrichten nicht lesen können
- Problem: n:n-Kommunikation mit $n \gg 1$ benötigt $\frac{n*n-1}{2}$ Verbindungen
- vollständiger Graph: n=25$\rightsquigarrow$ 300 ; n=50.000$\rightsquigarrow$ 1.249.975.000

cryptomancer.de

Matrix und MLS

- https://opencode.de/de/software/synapse-mit-mls-4201
- https://matrix.org/blog/2023/07/a-giant-leap-with-mls/
- https://arewemlsyet.com/

Inhaltsverzeichnis

1 Einführung

2 Matrix - 3DH-KEX und DoubleRatchet

3 Messaging Layer Security Protocol
- Fähigkeiten
- Architektur
- Ratschenbaum

4 PQC

cryptomancer.de

Messaging Layer Security

- Sicherheitsschicht für Ende-zu-Ende-Verschlüsselung
- sicheres und effizientes Protokoll
- Application Layer
- für große Gruppen mit bis zur 50 000 Teilnehmern
- Idee dazu 2016 auf der IETF96 in Berlin geboren
- 2017: Asynchronous Ratcheting Trees
- 29.3.2023: MLS von der IETF als neuer Standard verabschiedet
- Draft für PQC (ML-KEM) läuft

Messaging Layer Security

- Schlüssel-Einigungs-Protokoll für Gruppen
- für z.B. Instant Messaging
- Schutz gegen Abhören, Manipulation, gefälschte Nachrichten,
- Forward Secrecy und Post Compromise Security
- Geteilter, geschützter Gruppen-Zustand (Mitgliedschaft, krypt. Eigenschaften ...)
- Nachrichten-Konsistenz in Gruppe: jedes Mitglied erhält die gleiche Nachricht
- Architektur *und* Protokoll
- *kein* Nachrichtenprotokoll
- MLS soll in XMPP oder Matrix eingebettet werden
- MLS kann auch ohne kryptographische Absicherung eingesetzt werden

Logischer Aufbau

- *Clients* bilden *Gruppen*
- Gruppen bilden *Epochen*, z.B. wenn sich Eigenschaften der Gruppe ändern
- Jede Gruppe wird als Baum repräsentiert
- Gruppenmitglieder sind Blätter des Baumes
- Untergruppen des Baumes können effizient verschlüsseln
- jeder Client hat ein `LeafNode` mit Identität, Credentials und Fähigkeiten
- Fähigkeiten: des eingesetzten MLS-Protokolls und/oder Erweiterungen

Abstrakte Dienste

Authentication Service (AS)

- attestiert die Beziehung von öffentlichen MLS-Schlüsseln und der Identität
- generiert Credentials die diese Beziehung codieren
- validiert diese Credentials

Delivery Service (DS)

- verteilt Nachrichten an Gruppenmitglieder
- speichert und verteilt öffentliche Schlüssel (`KeyPackage`), die benötigt werden um geheime Gruppenschlüssel auszuhandeln

Dienste können im Netz verteilt sein

cryptomancer.de

Synchronität

- jede MLS-Operation kann asynchron ausgeführt werden
- Gruppen-Schlüssel können aktualisiert werden
- Mitglieder entfernen oder reinlassen
- Nachrichten verschicken
- Transportschicht des Messengers muss dafür sorgen dass MLS-Nachrichten asynchron und verlässlich zugestellt werden

Access Control

- Alle Gruppenmitglieder mit Zugriff auf den Gruppenschlüssel können Gruppen-Operationen starten
- jedes Gruppenmitglied kann Mitglieder entfernen/reinlassen
- Messenger muss Access Control implementieren
- z.B. Gruppenadministratoren ernennen oder offene Foren zulassen
- Handshake-Nachrichten können unverschlüsselt gesendet werden z.B. für ACL
- Transportverschlüsselung wichtig!

TreeSync Protokoll

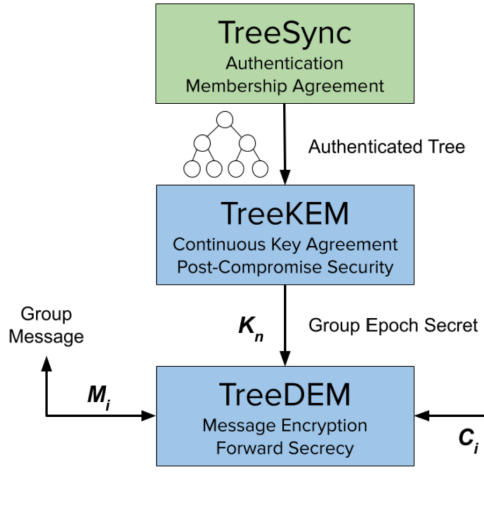
- synchronisiert den geteilten Gruppen-Zustand zwischen den Gruppenmitgliedern
- als Binär-Baum
- jedes Blatt repräsentiert ein Mitglied
- Gruppenzustand wird über Signaturen in Merkle-Bäumen authentifiziert
- Datenstruktur des Baumes stellt eine interne Integritäts-Invariante sicher
- authentifizierter, synchronisierter Zustand wird an TreeKEM übergeben

TreeKEM Protokoll

- ermöglicht es jedem Gruppenmitglied seine geheimen Schlüssel zu nutzen
- auf der Sequenz von authentifizierten Zuständen aus TreeSync
- um eine Sequenz von Gruppen-Schlüsseln (sog. Epochen-Geheimnissen) abzuleiten
- nutzt Baumstruktur für effiziente Updates: logarithmische Anzahl an Verschlüsselungen und 1 Entschlüsselungen
- Post Compromise Security und Schutz gegen ehemalige Gruppenmitglieder

TreeDEM Protokoll

- nimmt das Epochen-Geheimnis von TreeKEM
- leitet davon Nachrichten-Verschlüsselungs-Schlüssel für jedes Gruppenmitglied ab
- nach jeder Nachricht wird der Nachrichten-Verschlüsselungs-Schlüssel weiter geratscht
- um Forward Secrecy zu bekommen

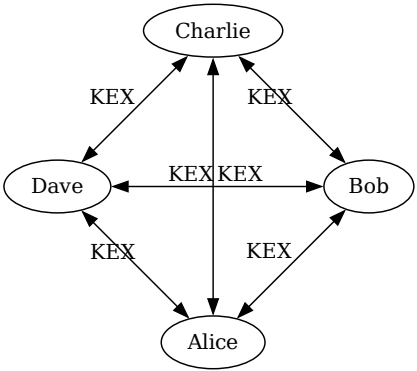


aus [1]

cryptomancer.de

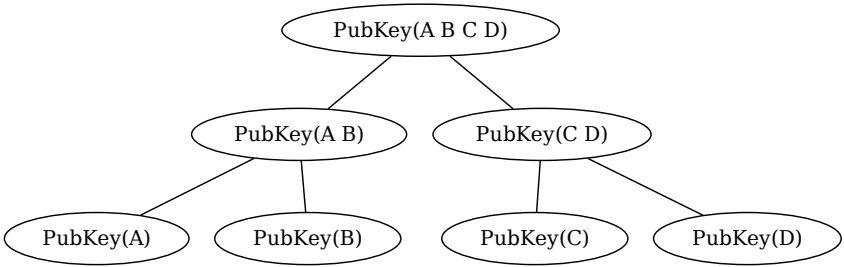
Ratschenbäume

- MLS nutzt ebenfalls Ratschen-Bäume
- als perfekt ausbalancierte Binär-Bäume
- Jedes Gruppenmitglied ist einem Blatt zugeordnet
- Pfad zur Wurzel enthält alle geheime Schlüssel, die das Mitglied kennt
- Der geheime Schlüssel in einem Knoten ist einem Mitglied nur bekannt, wenn der Unterbaum des betreffenden Knoten das Blatt des Mitglied beinhaltet



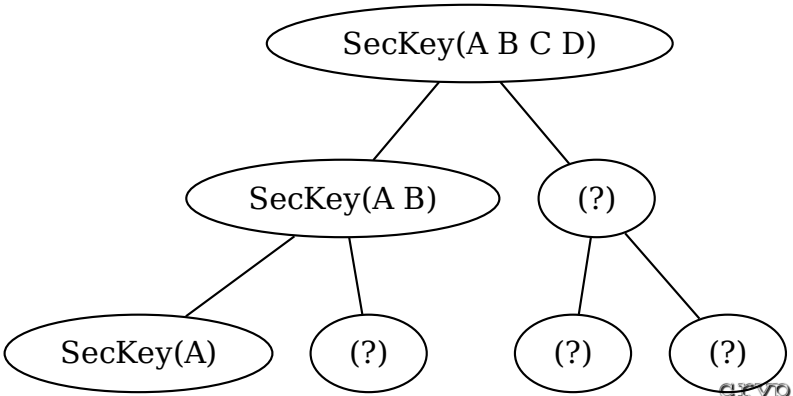
N to N (5 to 5 = 10)

cryptomancer.de



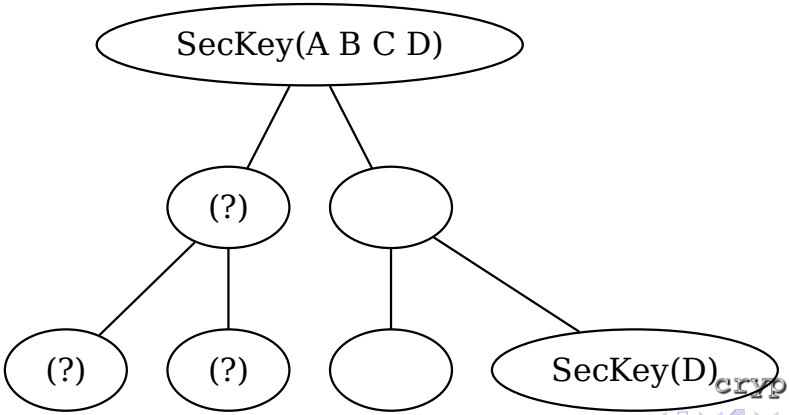
Public Tree

cryptomancer.de



Full Secret Tree for Alice

cryptomancer.de



cryptomancer.de

Inhaltsverzeichnis

1. Einführung
2. Matrix - 3DH-KEX und DoubleRatchet
3. Messaging Layer Security Protocol
   - Fähigkeiten
   - Architektur
   - Ratschenbaum
4. PQC

cryptomancer.de

Probleme durch PQC

- massiver Kommunikations-Overhead: Pre-Key Bundles und Messages wachsen (32Byte $\rightsquigarrow$ 1-2 kByte)
- CPU-Last $\uparrow$ $\rightsquigarrow$ Latenz steigt $\uparrow$ bei jeder PQC-Operation
- Pre-Key-Packages wachsen $\rightsquigarrow$ Speicher- und Bandreitenbedarf steigt, Skalierbarkeit?
- Abstreitbarkeit erschwert mit PQC-Ciphersuites: bisher Signatur durch symmetrische Schlüssel; PQC stärker formalisiert für härtere Beweisbarkeit nur ID-Key und PreKey mit PQC signieren, dann klassische Double-Ratchet, PCQ-Ciphertext nicht signieren
- PQC ist kein Drop-In-Replacement
- Protokoll noch komplexer!
- Implementierungsrisiko steigt

- hybride Ratchet - ECC-Double-Ratchet plus PQC-Ratchet
- X3DH ⇝X3PQC: DH durch ML-KEM ersetzen
- Layeransatz: zusätzlicher PQC-Layer
- neue, gemeinsame KDF-Pipeline ⇝ Sicherheit und Kompatibilität ↑
- Begrenzung von PQC-Operationen z.B. in Epochen-Ereignissen Last und Bandbreite ↓
- MLS: Konsistenz und Auditierbarkeit statt Abstreitbarkeit ⇝ Architekturänderungen in MLS: ID-Key und PreKeyBundle signiert, Nachrichten nur symm. verschlüsseln und MAC

- hybride Ratchet - ECC-Double-Ratchet plus PQC-Ratchet
- X3DH $\rightsquigarrow$X3PQC: DH durch ML-KEM ersetzen
- Layeransatz: zusätzlicher PQC-Layer
- neue, gemeinsame KDF-Pipeline $\rightsquigarrow$ Sicherheit und Kompatibilität $\uparrow$
- Begrenzung von PQC-Operationen z.B. in Epochen-EreignissenLast und Bandbreite $\downarrow$
- MLS: Konsistenz und Auditierbarkeit statt Abstreitbarkeit $\rightsquigarrow$ Architekturänderungen in MLS: ID-Key und PreKeyBundle signiert, Nachrichten nur symm. verschlüsseln und MAC

Fragen?

- https://cryptomancer.de/messaginglayer/
- https://fediverse.cryptomancer.de/cryptomancer
- https://mastodon.social/@0xKaishakunin *deprecating*
- 7B2B 45B1 330E 579E 3C3E 9B34 089D 8068 8050 ECCF
- public [at] cryptomancer [dot] de

cryptomancer.de