

sayTRUST

WHITEPAPER

Kommunikation und Verschlüsselung

Sicherer Zugriff vom Token bis zur Anwendung

Token Client und geschützte Kommunikation

- Hardwaregestützte Identität
- Verifizierte Verbindung
- Verschlüsselte Kommunikation

Inhalt

Executive Summary

1. Schutzschichten und Zertifikatsverarbeitung

1.1 Speicher, Zertifikat und Prozessspeicher

1.2 Native Transportverschlüsselung und Perfect Forward Secrecy

2. Mikroprozessor und Token-Hardware

2.1 Aufbau und Sicherheitsfunktionen

2.2 Kenndaten des Herstellers

2.3 Bedeutung der Features im UZE-Kontext

3. sayDEMON und Zugriffsentscheidung

4. Persönliche Aktivierung und Zugriffsprüfung

5. Single-Sign-On (SSO) und Passwortdatenbank

Fazit

Executive Summary

sayTRUST UZE verbindet persönliche Aktivierung, geschützte Zertifikatsverarbeitung, gegenseitige TLS-Authentisierung und anwendungsbezogene Zugriffsregeln zu einer mehrstufigen Sicherheitskette. Dieses Whitepaper erläutert, wie die einzelnen Schutzschichten zusammenwirken und welche Aufgaben der Mikroprozessor-Token, der Kommunikationsclient, sayDEMON und die Konfigurationsverwaltung übernehmen.

Der biometrisch aktivierte Hardware-Token stellt einen AES-verschlüsselten Speicherbereich für portable Programme, persönliche Daten und die verschlüsselte SSO-Datenbank bereit. Die Zertifikatsdatei besitzt eine zusätzliche Verschlüsselung und wird über die Benutzer-PIN entsperrt. Hardwarebindung und OTP ergänzen die Identitätsprüfung. Diese Mechanismen erfüllen unterschiedliche Aufgaben in den jeweiligen Sicherheitsstufen und bilden gemeinsam die Grundlage für die Freigabe der zugewiesenen Anwendungen.

1. Persönliche Identität	2. Geschützte Kommunikation	3. Nachvollziehbarer Betrieb
Biometrie, PIN, Hardware-bindung und OTP ergänzen sich bei der Aktivierung und Zugriffsprüfung.	TLS 1.3 und gegenseitige Zertifikatsprüfung sichern die Verbindung; Sitzungsregeln begrenzen die Ziele.	Syslog unterstützt die Auswertung. Policy-Aktualisierung und Schlüsselverwaltung sind gezielt zu prüfen.

sayTRUST Universal Post Zero Trust Exchange (UZE)

Sichere Kommunikations- und Verschlüsselungsverfahren

Dieses Whitepaper beschreibt den Mikroprozessor-Token, die Zertifikatsdatei und die native Kommunikation über TLS 1.3 mit gegenseitiger Authentifizierung (Mutual TLS) im neuen sayTRUST UZE. Es erläutert die Prüfungen beim Verbindungsaufbau sowie die Grenzen der laufenden Policy-Kontrolle. Sicherheitsbefunde zur aktuellen Schlüsselablage sind als intern zu klärende Punkte dokumentiert.

1. Schutzschichten und Zertifikatsverarbeitung

1.1 Speicher, Zertifikat und Prozessspeicher

Das biometrische Mikroprozessor-Token verfügt über einen AES 256 verschlüsselten Speicherbereich. Je nach Token Variante werden Kapazitäten von 16 bis 128 GB angeboten. AES geschützt liegen portable Programme, persönliche Daten und die verschlüsselte Single-Sign-On-Datenbank im Speicherbereich.



Abb.: Persönlicher Token und geschützter Anwendungszugriff.

Die Zertifikatsdatei auf dem AES verschlüsselten Token ist selbst AES-256 symmetrisch verschlüsselt; zur Entschlüsselung wird die Benutzer-PIN mit einem Salt verwendet. sayTRUST.exe lädt die Datei beim Start und entschlüsselt sie nach Eingabe des PIN's vor dem Verbindungsaufbau.

Die Größenangaben von 16 bis 128 GB bezeichnen den nutzbaren Speicher der Token varianten. Der Mikrocontroller selbst hat getrennte, wesentlich kleinere interne Flash- und SRAM-Bereiche.

Die Verarbeitung des entschlüsselten Zertifikats und die Lebensdauer des privaten Schlüssels im Speicher erfolgen im Clientprozess. Die Hardware-ID wird aus einem Salt, der Software-Volume-ID, der Hardware-Seriennummer und einer auf dem Medium gespeicherten GUID gebildet. Die Werte werden zusammengeführt und gehasht. Beim Kommunikationsaufbau meldet der Client diesen Hashwert als HWID an den Server. Der Server vergleicht die gemeldete HWID mit dem gespeicherten Wert.

Bei Abweichung wird der Benutzer automatisch gesperrt. Die Wiederfreigabe erfolgt durch Support über die Serveroberfläche. Maßgeblich ist die Benutzerkennung, zu der die HWID hinterlegt ist.

1.2 Native Transportverschlüsselung und Perfect Forward Secrecy

sayDEMON 2.0 erzwingt im nativen Self-TLS-Modus TLS 1.3; ein TLS-1.2-Fallback ist nicht vorgesehen. Die TLS-1.3-Cipher-Suite wird ausgehandelt und ist nicht fest im Code vorgegeben. Verfügbar sind:

- TLS_AES_256_GCM_SHA384,
- TLS_CHACHA20_POLY1305_SHA256 und
- TLS_AES_128_GCM_SHA256.

Cipher-Suite und die Schlüsselaustauschgruppe werden zwischen Client und installierter OpenSSL-Version ausgehandelt. Das untersuchte System bietet unter anderem folgende Gruppen an:

- X25519,
- X448,
- NIST-Kurven P-256/P-384/P-521,
- Brainpool-Kurven,
- ffdhe2048 bis ffdhe8192,
- ML-KEM-Gruppen und
- hybride Gruppen wie X25519MLKEM768

TLS 1.3 bezeichnet die Schlüsselaustauschgruppe getrennt von der Cipher-Suite; aus dem Suite-Namen lässt sich keine feste Gruppe oder ein bestimmtes PFS-Verfahren ableiten.

Die Zertifikate der Vertrauenskette sind mit SHA-256 oder einem stärkeren Verfahren signiert. Die Stärke der TLS-1.3-Verschlüsselung und die Vertrauenswürdigkeit der Zertifikatskette müssen getrennt bewertet werden. Im nativen Standardmodus terminiert sayDEMON TLS selbst über Python/OpenSSL und sieht die tatsächliche Client-IP sowie die Seriennummer im Clientzertifikat.

2. Mikroprozessor und Token-Hardware



Abb. 1: sayTRUST Mikroprozessor-Token als persönlicher Hardware-Client.

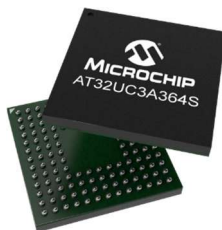


Abb. 2: Abbildung des AT32UC3A364S.

2.1 Aufbau und Sicherheitsfunktionen

Hier dargestellte Produktfamilie AT32UC3A3/A4 ist mit einem 32-Bit-AVR-Mikrocontroller und integriertem AES-Hardwaremodul in den mit „S“ gekennzeichneten Ausführungen. Zu den für das Token Konzept relevanten Bausteinen gehören die Memory Protection Unit, interne Flash- und SRAM-Speicher, Speicher- und USB-Schnittstellen sowie ein Watchdog. Das AES-Modul unterstützt Schlüssel mit 128, 192 und 256 Bit. Der Mikrocontroller stellt damit technische Funktionen bereit.

2.2 Kenndaten des Herstellers

Kenngroße AT32UC3A3/A4 und Varianten

<i>Architektur und Takt</i>	32-Bit AVR32 UC; bis 84 MHz
<i>Interner Flash</i>	64 / 128 / 256 KB je nach Chipvariante
<i>Internes SRAM</i>	64 KB direkt angebunden + 2 × 32 KB am Bus (insgesamt 128 KB)
<i>AES-Hardware</i>	AES-128, AES-192 und AES-256 bei den S-Varianten
<i>USB</i>	USB 2.0 High Speed, Device und Embedded Host
<i>Betriebstemperatur</i>	–40 °C bis +85 °C (Chip-Spezifikation)
<i>Versorgung</i>	3,0–3,6 V laut parametrisierter Produktübersicht
<i>Token-Nutzspeicher</i>	16–128 GB je nach sayTRUST-Token Variante; kein interner MCU-Speicher

Die Betriebstemperatur ist eine Angabe für den Mikrocontroller.

2.3 Bedeutung der Features im UZE-Kontext

Hardware-AES

Der Chip kann kryptografische AES-Operationen auf Hardware ausführen. Die tatsächliche Token-Speicherverschlüsselung, Betriebsart und Schlüsselverwaltung müssen produktspezifisch belegt werden.

Memory Protection Unit und Speicher

Die MPU unterstützt die Trennung von Speicherbereichen. Der interne Flash speichert Chipcodes, das SRAM dient der laufenden Verarbeitung; beides ist vom 16–128-GB-Token-Speicher zu unterscheiden.

USB und externe Speichereinbindung

Die USB-Schnittstelle verbindet Token und Arbeitsplatz. Schnittstellen für externe Speicher erlauben zusätzliche Komponenten, belegen aber noch nicht deren konkrete Verschlüsselungsarchitektur.

Watchdog und Energieverwaltung

Der Watchdog kann auf blockierte Abläufe reagieren. Welche Schutz- und Fehlerbehandlung die Firmware bei Ausfall oder Entfernen des Tokens ausführt, bestimmt die Produktimplementierung.

3. sayDEMON und Zugriffsentscheidung

Der sayDEMON 2.0 ist die serverseitige Komponente. Er begrenzt die Anzahl der Verbindungen je Quell-IP, führt die gegenseitige TLS-Authentisierung sowie die CA-Prüfung durch und liest die Zertifikatsseriennummer aus. Diese prüft er per HTTP gegen das Preconfiguration-Backend.

Wird die Seriennummer akzeptiert, verarbeitet der sayDEMON die Anfrage mit den Parametern void und votyp. Anschließend ruft er die clientspezifische Konfiguration vom Konfigurations-Backend ab, speichert die Regeln lokal und übermittelt die Konfiguration an den Client.

Nachfolgende TCP- und UDP-Zielanfragen werden anhand der lokal gespeicherten Regeln geprüft. Ohne passende Freigabe wird der Zugriff auf das Ziel nicht erteilt.

Die Backendprüfung der Zertifikatsseriennummer ist aus Sicht des serverseitigen sayDEMON die Berechtigungsprüfung. Die fachliche Bedeutung einer „aktiven Buchung“ liegt im Backend. Token-Aktivierung, PIN, SSO, Hardwarebindung und OTP werden in der Client- beziehungsweise Konfigurationsverwaltung verarbeitet; diese Vorgänge sind nicht Teil des serverseitigen sayDEMON.

Der sayDEMON prüft jede neue TCP- und UDP-Zielanfrage gegen die in der Sitzung geladenen Regeln. Weitere operative Prüfungen betreffen Protokollvalidierung, Verbindungslimits und Idle-Timeouts. Änderungen werden beim Konfigurationsabruf einer neuen Sitzung wirksam. Nach Wiederverbindung werden TLS-Handshake, Zertifikats- und Seriennummernprüfung sowie Policy-Abruf wiederholt.

Der sayDEMON schreibt strukturierte JSON-Ereignisse über den lokalen Syslog-Socket /dev/log mit Facility LOCAL0. Eine separate Logdatei oder direkte Verbindung zu einem zentralen Syslog-/SIEM-System legt sayDEMON nicht an. Die Weiterleitung, Aufbewahrung und Zugriffsrechte steuert die Systemkonfiguration. Ereignisse enthalten je nach Typ Zeit, Schweregrad, Host und Prozess, Session-ID, Quell- und Zieladresse samt Port, Clientname, TLS. Zertifikatsseriennummern werden aus Logfeldern entfernt und Zugangsdaten sowie URL-Parameter vor der Protokollierung redigiert.

4. Persönliche Aktivierung und Zugriffsprüfung

Die Sicherheitskette einer UZE-Verbindung

Vom persönlichen Token bis zur freigegebenen Anwendung



Abb.: Sicherheitskette vom Token bis zur freigegebenen Anwendung.

1. Biometrische Aktivierung

Der berechtigte Benutzer aktiviert das Mikroprozessor-Token per Fingerabdruck. Erst nach erfolgreicher Autorisierung und Aktivierung wird der geschützte Tokenbereich zugänglich und das Token vom Arbeitsplatzrechner erkannt.

2. SSO-Aktivierung

Der Benutzer entsperrt die verschlüsselte SSO-Datenbank mit seinem Datenbankpasswort. Nach erfolgreicher Eingabe startet sein zugewiesenes Anwendungsmenü.

3. Zertifikats-PIN und Bindungsprüfung

Die Benutzer-PIN entschlüsselt die geschützte Zertifikatsdatei vor dem Verbindungsaufbau. Ab dem dritten falschen PIN-Versuch beginnt eine Wartezeit, die mit jedem weiteren Fehlversuch um zehn Sekunden steigt.

4. sayDEMON und Policy

Nach der Identifizierung erhält der Client seine Konfiguration. Der serverseitige sayDEMON speichert die Regeln lokal und prüft spätere TCP- und UDP-Zielanfragen. Die Aufgaben des clientseitigen Kommunikationsclients sind davon zu unterscheiden.

5. TLS-Verbindung und Zertifikatsprüfung

Nach erfolgreicher gegenseitiger TLS-Authentisierung und den Prüfungen der Zertifikatsidentität, Hardwarebindung und OTP-Nummer lädt der Client seine Konfiguration. Ohne gültige Konfiguration kann der Tunnel nicht für Anwendungen verwendet werden.

6. OTP und Konfigurationsfreigabe

Die vom Nutzer eingegebene OTP-Nummer wird vom Client beim Verbindungsaufbau an die serverseitige Konfigurationsverwaltung übermittelt und dort validiert. Bei falschem OTP wird die Verbindung abgelehnt. Der Client versucht den Verbindungsaufbau derzeit dreimal und bricht danach ab; weitere Eingaben innerhalb desselben Aufbaus sind nicht möglich.

5. Single-Sign-On (SSO) und Passwortdatenbank

Die SSO-Funktion verwendet eine verschlüsselte Datenbank auf Basis der KeePass-Bibliothek. In der Hardwarevariante liegt die Datenbank zusätzlich im AES-verschlüsselten Token Bereich. Das Datenbankpasswort entsperrt die Einträge; der Zertifikats-PIN kann daraus für den Verbindungsaufbau bereitgestellt werden.

- Die Datenbankverschlüsselung und die Verschlüsselung des Token-Speichers sind getrennte Schutzschichten.
- Der SSO-Prozess ersetzt weder die biometrische Aktivierung noch die Zertifikats- und OTP-Prüfung.

Die sayTRUST UZE - Menüversionen verwenden eine KDBX-4.0-Datenbank. KeePass verarbeitet das Datenbankpasswort mit AES-KDF, 60.000 Runden und zufälligem 32-Byte-Seed. Die Datenbank ist mit AES-256-CBC und 16-Byte-IV verschlüsselt; KDBX 4.0 verwendet HMAC-SHA-256 für die Integritätsprüfung.

Fazit

Die Schutzwirkung von sayTRUST UZE beruht auf dem Zusammenspiel von Benutzerautorisierung, persönlicher Aktivierung, geschützter Zertifikatsdatei, Hardwarebindung, gegenseitiger TLS-Authentisierung und serverseitiger Freigabe definierter Anwendungsziele. Das Mikroprozessor-Token und die verschlüsselte SSO-Datenbank ergänzen diese Zugriffskette um Schutzschichten für lokale Daten und Zugangsinformationen. Für den Betrieb müssen die Aufgaben der Client-, Konfigurations- und Serverkomponenten im Zusammenhang betrachtet werden.

Eine belastbare Sicherheitsbewertung muss die tatsächlich eingesetzten Versionen von Client, Firmware und OpenSSL berücksichtigen. Neue Zielfragen werden anhand der geladenen Sitzungsregeln geprüft; geänderte Policies greifen erst beim Konfigurationsabruf für eine neue Sitzung. Die zentrale Logauswertung setzt eine entsprechend eingerichtete Systemkonfiguration voraus. Die konkrete Schlüsselverwaltung, die Lebensdauer privater Schlüssel im Clientprozess und die Nutzung der Hardware-Sicherheitsfunktionen müssen produktspezifisch nachgewiesen werden.

Mit dokumentierter Konfiguration, geklärter Schlüsselverarbeitung und abgestimmten Betriebsabläufen bietet UZE eine Grundlage für identitätsgebundene, verschlüsselte und gezielt begrenzte Anwendungszugriffe. Die beschriebenen Verfahren und Grenzen ermöglichen es, Schutzmaßnahmen nachvollziehbar zu planen und ihre Wirksamkeit in der jeweiligen Installation zu überprüfen.