

# UNIVERSITÀ DEGLI STUDI DI SALERNO

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE ED  
ELETTRICA E MATEMATICA APPLICATA



Corso di Laurea Magistrale in Ingegneria Informatica

**eCRED**

Nome	Cognome	Matricola	E-Mail
Michele	Martino	0622702424	m.martino48@studenti.unisa.it
Francesco	Quagliuolo	0622702412	f.quagliuolo@studenti.unisa.it
Emanuele	Relmi	0622702368	e.relmi@studenti.unisa.it
Benito	Senese	0622702425	b.senese1@studenti.unisa.it

ANNO ACCADEMICO 2023/2024

---

# INDICE

<b>1</b>	<b>WORK PACKAGE 1</b>	<b>3</b>
1.1	Attori del Sistema . . . . .	3
1.1.1	Utente . . . . .	3
1.1.2	Server . . . . .	4
1.1.3	Autorità di Rilascio delle Credenziali . . . . .	4
1.1.4	Autorità Rilascio CIE . . . . .	4
1.2	Attaccanti del Sistema . . . . .	5
1.3	Proprietà . . . . .	9
1.3.1	Confidenzialità . . . . .	9
1.3.2	Integrità . . . . .	9
1.3.3	Trasparenza . . . . .	10
1.3.4	Efficienza . . . . .	10
1.4	Completeness . . . . .	11
<b>2</b>	<b>WORK PACKAGE 2</b>	<b>12</b>
2.1	Panoramica Generale di Funzionamento . . . . .	12
2.2	Generazione e Utilizzo delle Credenziali . . . . .	13
2.2.1	Supposizioni . . . . .	13
2.2.2	Processo di Richiesta e Rilascio delle Credenziali . . . . .	13
2.2.3	Processo di Identificazione e Accesso ai Servizi . . . . .	14
2.2.4	Configurazione certificati X.509v3 . . . . .	15
2.2.5	Politiche di Revoca e Aggiornamento della Certificate Revocation List (CRL) . . . . .	16
2.2.6	PoW: Puzzle Parametrizzato . . . . .	17

---

## INDICE

---

2.3	Specifiche degli algoritmi utilizzati . . . . .	17
2.3.1	Gen . . . . .	17
2.3.2	Sign <sub>sk</sub> . . . . .	17
2.3.3	Verify <sub>pk</sub> . . . . .	17
2.4	Protezione della Privacy e Integrità delle Credenziali . . . . .	18
2.5	Conclusione . . . . .	18
<b>3</b>	<b>WORK PACKAGE 3</b>	<b>19</b>
3.1	Confidenzialità . . . . .	19
3.2	Integrità . . . . .	20
3.3	Trasparenza . . . . .	22
3.4	Efficienza . . . . .	23
<b>4</b>	<b>WORK PACKAGE 4</b>	<b>26</b>
4.1	Introduzione al Work Package 4 . . . . .	26
4.2	Inizializzazione e Configurazione Iniziale . . . . .	26
4.2.1	Variabili e Cartelle . . . . .	26
4.2.2	Definizione del PIN CIE . . . . .	27
4.3	Generazione di Chiavi RSA . . . . .	27
4.3.1	Generazione chiave ECDSA . . . . .	27
4.4	Funzioni di Gestione di Certificati . . . . .	27
4.4.1	Creazione e Firma di Richieste di Certificato . . . . .	27
4.4.2	Creazione di Certificati con Credenziali . . . . .	28
4.5	Gestione di Server TLS . . . . .	28
4.5.1	Avvio e Connessione al Server TLS . . . . .	28
4.6	Configurazione di Apache con HTTPS (TLS) . . . . .	28
4.6.1	Configurazione del Server Apache . . . . .	28
4.7	Interazioni Utente-CA e Utente-Server . . . . .	28
4.8	Configurazione della CA . . . . .	28
4.8.1	Definizione delle Politiche di Certificazione . . . . .	28
4.9	Flusso Principale dello Script . . . . .	29
4.9.1	Sequenza Operativa . . . . .	29
4.10	Revoche e Pulizia . . . . .	29
4.11	Output Script . . . . .	29

---

---

# CAPITOLO 1

---

## WORK PACKAGE 1

In questo primo capitolo ci concentreremo sulla definizione degli attori onesti coinvolti nel sistema, analizzando i loro obiettivi e le funzionalità che si intendono realizzare. Gli attori onesti sono quegli individui o entità che agiscono in conformità con le regole e le politiche stabilite, cercando di raggiungere i loro obiettivi senza compromettere l'integrità del sistema.

Successivamente, esamineremo i possibili avversari (o threat models) che potrebbero essere interessati a compromettere il sistema, esaminando le loro risorse e le motivazioni che li spingono ad agire. Questa analisi ci permetterà di identificare gli attacchi che il sistema potrebbe subire e di comprendere quali misure di sicurezza dovranno essere adottate per contrastarli.

Una volta compreso il contesto in cui il sistema opera, identificheremo le proprietà di sicurezza che si vorrebbero preservare in presenza di attacchi. Queste proprietà sono fondamentali per garantire il corretto funzionamento del sistema e la protezione delle informazioni sensibili.

Di seguito un elenco dettagliato di tali attori e delle loro responsabilità.

### 1.1 Attori del Sistema

#### 1.1.1 Utente

Gli utenti sono le persone che necessitano di accedere ai servizi qualificati offerti dai server. Ogni utente possiede una Carta d'Identità Elettronica (CIE), contenente un certificato digitale di tipo X.509v3, e il PIN ad essa associato. La CIE ha una chiave segreta di firma, utilizzabile dall'utente durante la fase di richiesta delle credenziali, garantendo così

l'integrità e l'autenticità dei dati forniti alle CA. Il loro obiettivo è richiedere e ottenere le credenziali necessarie dalle autorità, per poi utilizzarle per accedere a servizi specifici in modo sicuro.

### 1.1.2 Server

I server sono entità che offrono servizi qualificati, accessibili solo ai possessori di specifiche credenziali. Essi devono essere in grado di verificare le credenziali presentate dagli utenti e stabilire se sono rilasciate da autorità fidate. Il loro obiettivo è offrire servizi in modo sicuro e limitato solo agli utenti autorizzati. Sono, inoltre, responsabili di gestire le richieste di accesso, garantire la sicurezza e la privacy delle operazioni e minimizzare il coinvolgimento di terze parti fidate così da ridurre i rischi legati ad un singolo punto di fallimento.

### 1.1.3 Autorità di Rilascio delle Credenziali

Le autorità di rilascio delle credenziali (CA) sono enti fidati che emettono credenziali agli utenti, garantendo che queste siano valide e autentiche. Queste autorità verificano l'identità degli utenti e le informazioni correlate prima di rilasciare le credenziali. Esse possono essere enti governativi, istituzioni pubbliche o altre organizzazioni autorizzate a rilasciare certificati digitali. Sono, quindi, responsabili di:

- verificare e validare le informazioni degli utenti;
- emettere credenziali autentiche;
- garantirne la sicurezza e l'integrità;
- mantenere un registro delle credenziali emesse.

### 1.1.4 Autorità Rilascio CIE

Gli enti che si occupano dell'emissione della CIE con il PIN associato sono fondamentalmente due: l'Ufficio Anagrafe e l'Istituto Poligrafico e Zecca dello Stato (IPZS), che vengono accorpati in un'unica entità per convenienza e semplicità. Questa autorità si occupa della gestione delle pratiche amministrative relative alla registrazione delle informazioni anagrafiche e della generazione e gestione delle chiavi crittografiche e dei certificati digitali necessari per la firma elettronica.

È responsabile di:

- produrre e consegnare la CIE, insieme al suo PIN, all'utente;
- generare e memorizzare in modo sicuro la chiave segreta e il certificato digitale all'interno della CIE;

- garantire che i certificati digitali siano validi e riconosciuti dalle autorità competenti;
- implementare misure di sicurezza avanzate per proteggere la CIE da contraffazioni e accessi non autorizzati.

### 1.2 Attaccanti del Sistema

Nel contesto del sistema descritto, è fondamentale analizzare e comprendere i potenziali attaccanti che potrebbero cercare di comprometterlo, considerando le motivazioni e le risorse a loro disposizione. Questi attori e gruppi di attaccanti rappresentano diverse minacce al sistema e richiedono misure di sicurezza specifiche per essere contrastati efficacemente.

- **Anti-Tech Theorist**
  - **Tipologia:** passivo/attivo
  - **Descrizione:** avversario che si oppone all'innovazione e all'adozione di nuove tecnologie, sostenendo che qualunque dispositivo può essere compromesso, a discapito della privacy degli utenti. Il suo scopo è quello di amplificare notizie di violazioni di dati o problemi di privacy dei sistemi per influenzare l'opinione pubblica e screditare le nuove proposte nell'ambito della sicurezza informatica.
  - **Risorse**
    - \* Abilità comunicative elevate
    - \* Rete di contatti con altri oppositori delle nuove tecnologie
    - \* Accesso ai media e ai social network per diffondere messaggi allarmistici
- **The Eavesdropper**
  - **Tipologia:** passivo
  - **Descrizione:** avversario interessato a intercettare tutte le operazioni svolte tramite il sistema (autenticazione, dati della CIE, etc.). Questa tipologia di avversario può essere sia interna alle CA o ai server che esterna. L'obiettivo principale è quello di collezionare le informazioni inerenti agli utenti, senza però compromettere i protocolli, i quali sono eseguiti onestamente.
  - **Risorse**
    - \* Conoscenza approfondita delle tecniche di intercettazione e analisi delle comunicazioni, nonché della crittografia utilizzata nel sistema per comprendere e interpretare correttamente i dati intercettati

- \* Capacità di monitorare e intercettare le comunicazioni tra gli utenti e il sistema
- \* Disponibilità di risorse computazionali significative per analizzare e elaborare grandi quantità di dati intercettati

- **Malevolent Server**

- **Tipologia:** attivo
- **Descrizione:** server con comportamento malevolo. I suoi compiti includono: la corretta gestione delle connessioni, la verifica delle credenziali e l'erogazione del servizio richiesto. Tuttavia, questa tipologia di avversario sfrutta le informazioni ricevute dagli utenti per scopi illeciti, come il riutilizzo di informazioni personali o la vendita dei dati carpiri a terzi, compromettendo la privacy degli utenti.
- **Risorse**
  - \* Controllo completo sui dati e sulle risorse del server, inclusi database utenti, registri di accesso e altri dati sensibili
  - \* Possibilità di accedere alle credenziali ricevute per l'accesso al servizio, consentendo di utilizzare queste informazioni per scopi illeciti.

- **Identity Thief**

- **Tipologia:** attivo
- **Descrizione:** avversario che possiede o mira a rubare identità e/o credenziali per impersonare gli utenti legittimi e ottenere accesso non autorizzato ai servizi. Potrebbe utilizzare tecniche di phishing, furto di credenziali o exploit delle vulnerabilità per ottenere informazioni sensibili e assumere l'identità di altri utenti.
- **Risorse**
  - \* Disponibilità di risorse computazionali sufficienti per eseguire attacchi brute-force, decrittazione o altri metodi per ottenere informazioni sensibili o compromettere la sicurezza del sistema
  - \* Possesso di un vasto database di informazioni personali ottenute illegalmente, che consente la creazione di profili dettagliati delle vittime e di utilizzare le informazioni per scopi malevoli
  - \* Competenze nel phishing e nell'ingegneria sociale, accesso a database di credenziali rubate, capacità di creare siti web e messaggi convincenti

- **The Denier**

- **Tipologia:** attivo
- **Descrizione:** avversario il cui scopo è compromettere il sistema, colpendo gli utenti o i servizi contemporaneamente. Potrebbe utilizzare botnets, malware o attacchi DoS/DDoS per sovraccaricare o infiltrare il sistema, causando danni diffusi e gravi interruzioni del servizio.
- **Risorse**
  - \* Capacità di coordinare attacchi su vasta scala
  - \* Accesso a risorse informatiche distribuite (come server cloud, dispositivi IoT compromessi o altre infrastrutture che possono essere sfruttate per eseguire questo tipo di attacchi)
  - \* Possibilità di avere motivazioni finanziarie o ideologiche significative per condurre gli attacchi, come il lucro finanziario, il furto di dati sensibili o il danneggiamento delle infrastrutture critiche per fini politici o di estorsione

- **Evil Insider**

- **Tipologia:** passivo/attivo
- **Descrizione:** avversario interno al sistema, che agisce in modo malevolo per ottenere benefici personali o danneggiare l'organizzazione. Potrebbe rubare credenziali o informazioni sensibili, manipolare il sistema o sabotare le operazioni per raggiungere i suoi obiettivi.
- **Risorse**
  - \* Autorizzazioni elevate all'interno del sistema, compreso l'accesso a dati sensibili e risorse critiche
  - \* Conoscenza interna dei processi e delle vulnerabilità

- **The Cryptographer**

- **Tipologia:** attivo
- **Descrizione:** avversario che mira a compromettere la sicurezza del sistema, attaccando le autorità di certificazione o le infrastrutture chiave del sistema di identificazione, sfruttando vulnerabilità nei protocolli crittografici utilizzati per proteggere le comunicazioni e le credenziali. Potrebbe condurre attacchi di tipo man-in-the-middle, falsificare certificati o compromettere le chiavi crittografiche per ottenere accessi non autorizzati.



- **Risorse**

- \* Comprensione approfondita dei protocolli crittografici, dei meccanismi di autenticazione e delle vulnerabilità delle autorità di certificazione
- \* Accesso a risorse computazionali per analizzare e rompere algoritmi crittografici

- **Data Miner**

- **Tipologia:** passivo/attivo

- **Descrizione:** questo avversario mira a raccogliere e sfruttare i dati degli utenti per fini di analisi, profilazione o vendita a terzi

- **Risorse**

- \* Competenze nell'analisi dei dati
- \* Accesso a tecnologie per l'estrazione e l'elaborazione dei dati
- \* Comprensione delle leggi sulla privacy e delle normative sul trattamento dei dati

- **The Data Harvesters**

- **Tipologia:** attivo

- **Descrizione:** questo gruppo mira a raccogliere e sfruttare i dati degli utenti per scopi di profitto o di spionaggio. Utilizzano una combinazione di tecniche di intercettazione, furto di identità, spyware e analisi dei dati per ottenere informazioni sensibili

- **Risorse**

- \* Competenze nell'intercettazione delle comunicazioni e nell'analisi dei dati (The Eavesdropper)
- \* Accesso a database di informazioni personali rubate (Identity Thief)
- \* Capacità di elaborare grandi quantità di dati raccolti per identificare opportunità di profitto o informazioni sensibili (Data Miner)

- **Crypto Conspirators**

- **Tipologia:** attivo

- **Descrizione:** si concentra sull'attacco alla crittografia e alla sicurezza del sistema. Utilizzano competenze avanzate in crittografia, furto di identità e infiltrazione interna per compromettere la sicurezza

– **Risorse**

- \* Competenze avanzate in crittografia e conoscenza dei protocolli di sicurezza (The Cryptographer)
- \* Accesso a informazioni personali rubate per ottenere accesso non autorizzato (Identity Thief)
- \* Autorizzazioni elevate e accesso interno per facilitare gli attacchi (Evil Insider)

### 1.3 Proprietà

L'obiettivo è sviluppare un sistema che consenta l'accesso remoto a servizi riservati agli utenti in possesso delle credenziali richieste. In questo contesto, diverse autorità rilasciano credenziali agli utenti, i quali le utilizzano per accedere a servizi qualificati, ossia servizi limitati ai possessori di specifiche credenziali.

Per garantire l'efficacia e la sicurezza del sistema, è necessario considerare diverse proprietà fondamentali: confidenzialità, integrità, trasparenza ed efficienza. Queste proprietà non solo assicurano che il sistema funzioni correttamente, ma proteggono anche contro potenziali minacce e abusi.

#### 1.3.1 Confidenzialità

- **C1:** I dati sensibili degli utenti devono essere protetti, ovvero accessibili solo dalle parti autorizzate in modo controllato
- **C2:** Garantire la privacy delle comunicazioni tra gli utenti e i server che offrono i servizi, in modo che non sia possibile intercettare o rubare le credenziali trasmesse
- **C3:** Garantire la privacy delle comunicazioni tra gli utenti e le autorità di rilascio delle credenziali, in modo che non sia possibile carpire dati personali relativi agli utenti
- **C4:** Tutelare gli utenti contro i malintenzionati, di modo che questi ultimi non possano utilizzare l'identità di altri utenti, firmando certificati falsi o accedendo ai servizi sfruttando le loro credenziali

#### 1.3.2 Integrità

- **I1:** Preservare l'integrità delle credenziali rilasciate, garantendo che non possano essere alterate o falsificate

- **I2:** I server devono garantire che le credenziali esibite dagli utenti siano verificabili e autentiche, ovvero rilasciate da autorità riconosciute come affidabili
- **I3:** Verificare che la CIE sia riconducibile all'utente che la utilizza
- **I4:** Se un servizio è rivolto ad un individuo in possesso di due credenziali allora non deve essere possibile per due utenti, ciascuno avente una sola credenziale, combinare le loro informazioni per accedere al servizio

### 1.3.3 Trasparenza

- **T1:** Il sistema di autenticazione dovrebbe basarsi su algoritmi noti e verificabili, di modo da permettere una verifica di essi da parte degli utenti, rendendo arduo il processo di manomissione da parte di un malintenzionato, senza che quest'ultimo sia scoperto
- **T2:** Il processo di rilascio delle credenziali deve essere trasparente e verificabile da tutte le parti coinvolte, riducendo la dipendenza da terze parti fidate così da ridurre il rischio di abuso
- **T3:** Assicurarsi che le politiche di accesso ai servizi siano chiare e comprensibili, consentendo agli utenti di sapere quali credenziali siano necessarie per accedere agli stessi, limitando così la possibilità di accessi non autorizzati
- **T4:** La fiducia in specifiche parti deve essere giustificata da motivazioni concrete, come la possibilità di rilevare e punire abusi o il preservare la buona reputazione

### 1.3.4 Efficienza

- **E1:** Il sistema deve consentire ai server di controllare l'accesso utilizzando politiche avanzate e flessibili, permettendo di definire criteri di accesso complessi e specifici
- **E2:** Garantire che il processo di rilascio e verifica delle credenziali sia eseguito in modo rapido ed efficiente, riducendo al minimo i tempi di attesa e l'impatto sulle prestazioni del sistema
- **E3:** Assicurare che l'ottenimento delle credenziali tramite CIE avvenga in modo semplice e veloce, facilitando l'accesso remoto ai servizi digitali
- **E4:** Progettare il sistema in modo che possa gestire un grande numero di utenti e richieste contemporaneamente senza degradare le performance

### 1.4 Completeness

In questo paragrafo verrà definita la completeness, la quale illustra il comportamento del sistema nel caso in cui tutte le parti coinvolte si comportino onestamente.

Nello specifico:

- l'utente possessore della CIE richiede le credenziali necessarie alle autorità di rilascio
- le CA verificano la firma della CIE e l'identità dell'utente e, in caso positivo, emettono le credenziali
- l'utente detentore delle credenziali può quindi richiedere l'accesso ad un servizio specifico, istanziato su un server
- il server sottopone l'utente ad un'autenticazione prima di permettere l'accesso al servizio
- l'utente onesto accede al servizio

---

---

# CAPITOLO 2

---

## WORK PACKAGE 2

In questo capitolo verrà posta attenzione sul presentare una soluzione che risponde al modello identificato nel *Work Package 1 (WP1)*. L'obiettivo è di proporre un sistema che riesca a raggiungere un ragionevole compromesso tra efficienza, trasparenza, confidenzialità e integrità.

Concentreremo la nostra attenzione sui seguenti problemi chiave:

- Richiesta e ottenimento, dalle CA, delle credenziali necessarie per l'accesso ai servizi specifici
- Autenticazione e accesso ai servizi qualificati
- Protezione della privacy e dell'integrità delle credenziali

### 2.1 Panoramica Generale di Funzionamento

Innanzitutto ogni utente possiede una CIE, con al suo interno un certificato X.509v3, ed il PIN ad essa associato. Ciascuno di essi, tramite l'utilizzo della CIE, può richiedere delle credenziali ad un'autorità di rilascio delle stesse. Esistono varie autorità che rilasciano credenziali agli utenti e, una volta entrate in possesso, è possibile utilizzarle per identificarsi e accedere ai servizi qualificati. È consentito, all'utente, l'accesso ai servizi qualificati solo se le credenziali soddisfano i requisiti di accesso imposti dal servizio stesso.

## **2.2 Generazione e Utilizzo delle Credenziali**

In questa sezione verrà sviluppato un protocollo per la generazione e l'utilizzo di un sistema di identificazione tramite credenziali per l'accesso a servizi digitali qualificati. Questo processo coinvolge i seguenti attori chiave: l'utente, l'autorità di rilascio delle credenziali e il server del servizio. Gli utenti interagiscono con la CA per la richiesta e l'ottenimento delle credenziali, e con il server per usufruire dei servizi offerti. Si inizierà descrivendo come un utente può richiedere una credenziale all'autorità, chiarendone anche il processo di rilascio. Successivamente, verrà delineato il contenuto e la struttura delle credenziali, affinché possano essere utilizzate come strumento digitale per l'identificazione e l'accesso ai servizi, mantenendo un'alta riservatezza delle informazioni contenute. Infine, analizzeremo il processo di identificazione e accesso a un servizio digitale.

### **2.2.1 Supposizioni**

Il funzionamento del protocollo proposto si basa sulle seguenti supposizioni:

- Le istituzioni che si occupano dell'emissione della CIE, con annesso il suo PIN, sono assunte come parte fidata
- Ogni utente dispone inizialmente della propria CIE e del PIN associato
- Il certificato digitale presente nella CIE è di tipo X509v3
- All'interno del certificato sono presenti i dati che sono leggibili su una carta di identità
- Le autorità di rilascio sono fidate ed emettono credenziali corrette
- Gli algoritmi e i protocolli utilizzati sono pubblici, così da avere maggiore trasparenza e dare modo agli utenti di verificarne gli aspetti critici

Si procede adesso con l'analisi dettagliata di quanto delineato fino a questo momento.

### **2.2.2 Processo di Richiesta e Rilascio delle Credenziali**

#### **Generazione di una coppia di chiavi**

- Ogni utente genera una coppia di chiavi pubblica/privata
- La chiave pubblica  $pk_{utente}$  è generata in accordo all'algoritmo  $Gen(1^n)$

#### **Richiesta ed Emissione delle Credenziali**

- Le comunicazioni tra le CA e gli utenti avvengono tramite canali cifrati con TLS

- L'utente invia il certificato X.509v3 relativo alla sua CIE alla CA, insieme alla richiesta della specifica credenziale
- La CA invia una *challenge*  $b$  di bit casuali inviandola all'utente
- L'utente inserisce il PIN della sua CIE e calcola l'hash  $h = H(b)$  dei bit ricevuti  
**NOTA:**  $H : \{0,1\}^n \rightarrow \{0,1\}^{256}$  è una funzione hash di tipo SHA-256, con dominio illimitato e codominio di dimensioni pari a 256 bit
- Viene inviata una query  $Sign_{sk_{CIE}}(PIN, b)$  alla CIE
- La CIE verifica il PIN e, se corretto, rilascia una firma  $\sigma_{CIE}$  ECDSA del messaggio (la *challenge*)  $b$ .
- L'utente, infine, invia la firma  $\sigma_{CIE}$  all'autorità
- La CA verifica l'identità dell'utente e, se valida, emette una credenziale firmata digitalmente, sotto forma di certificato X.509v3, contenente le informazioni specifiche richieste dal soggetto

### 2.2.3 Processo di Identificazione e Accesso ai Servizi

Quando un utente presenta una credenziale per accedere a un servizio, il servizio deve verificare che l'utente possieda effettivamente la chiave privata utilizzata per firmare il documento X.509, corrispondente alla chiave pubblica inclusa nello stesso. Anche in questo caso, le comunicazioni tra utenti e server sono protette da canali cifrati tramite TLS.

#### Autenticazione dell'Utente

Quando l'utente desidera accedere a un servizio, utilizza lo schema di firma di Schnorr per autenticarsi al server. Nello specifico, l'utente rappresenta il *prover*  $P$ , mentre il server il *verifier*  $V$ . In questo modo, l'utente dovrà dimostrare al server di essere a conoscenza del segreto (la  $sk_{utente}$ , indicata con  $x$ ) a partire dalla  $pk_{utente}$  (indicata con  $y = g^x$ ).

Il protocollo si articola come segue:

- l'utente seleziona un valore casuale  $r \in \mathbb{Z}_q$ , calcola  $a = g^r$  e invia quest'ultimo al server
- il server sceglie una stringa casuale  $c \in \mathbb{Z}_q$  e lo invia all'utente
- l'utente invia al server il valore  $z = r + cx$ , con  $z \in \mathbb{Z}_q$

Il server, poiché  $g^z = ay^c$ , riesce a capire che, a richiedere il servizio, è la persona realmente in possesso delle credenziali.

### Verifica delle Credenziali

- Il servizio riceve, dall'utente, la credenziale tramite certificato X.509 e ne verifica la firma digitale
- Verifica, inoltre, che il numero di serie della credenziale non sia incluso nell'elenco delle revocazioni più aggiornato (CRL)
- Controlla che le informazioni contenute nella credenziale soddisfino i requisiti di accesso

### Accesso ai Servizi

- Per garantire un accesso sicuro e affidabile ai servizi, è stata implementata una strategia anti-spam basata su puzzle parametrizzati, la quale funge da PoW (Proof of Work)
- Durante il processo di accesso, il server genera un puzzle matematico complesso che richiede l'elaborazione da parte dell'utente per essere risolto. La corretta risoluzione del puzzle costituisce quindi un requisito fondamentale per l'accesso ai servizi qualificati.
- Superato il puzzle parametrico, il client procede con l'autenticazione basata su Schnorr
- Una volta fatto ciò, il server verifica la firma dell'emittente del certificato e se le credenziali sono ancora valide
- Se le credenziali soddisfano i requisiti di accesso imposti dal servizio e non sono state revocate, l'utente ottiene l'accesso al servizio richiesto
- Il server registra l'accesso dell'utente in modo sicuro e trasparente, mantenendo un log degli accessi per future verifiche

#### 2.2.4 Configurazione certificati X.509v3

I certificati adottano la struttura standard, la quale prevede:

- |  |  |
|--|--|
| • la specifica della versione;   | • il richiedente;  |
| • il numero seriale;   | • informazioni sulla chiave pubblica del richiedente, ovvero l'algoritmo utilizzato e la chiave pubblica stessa; |
| • il tipo di algoritmo di firma e i suoi parametri;                    | • ID univoco dell'autorità   |
| • la CA che ha emesso il certificato;                                  | • ID univoco soggetto  |
| • il periodo di validità, indicato da una data d'inizio e una di fine; |  |



### Certificato digitale interno alla CIE

Per questo tipo di certificato sono presenti anche le seguenti estensioni:

- Key Usage: Digital Signature
- Extended Key Usage: Client Authentication

Inoltre, anche i dati anagrafici, presenti sulla CIE, sono incorporati nel documento:

- |   |                          |
|---|--------------------------|
| • Comune o Ufficio Consolare emittitore | • Sesso                  |
| • Nome                                  | • Statura                |
| • Cognome                               | • Cittadinanza           |
| • Luogo di nascita                      | • Codice fiscale         |
| • Data di nascita                       | • Indirizzo di residenza |

### Certificato digitale per le credenziali

In questo X.509v3, sono presenti le seguenti estensioni:

- Custom Extension: Credenziale Richiesta dall'Utente
- CRL Distribution Point: URL della lista di revoca dei certificati (CRL)

### 2.2.5 Politiche di Revoca e Aggiornamento della Certificate Revocation List (CRL)

- Ogni certificato X.509 emesso contiene un numero di serie univoco
- Se un utente sospetta che la sua chiave privata sia stata compromessa o persa, può richiedere alla CA la revoca del certificato corrispondente
- La CA mantiene un database dei numeri di serie dei certificati emessi e revocati
- Periodicamente, la CA genera una Certificate Revocation List (CRL) contenente i numeri di serie dei certificati revocati e la firma digitale della CA
- La CRL include anche un campo CRL Distribution Point che indica l'URL da cui è possibile ottenere l'ultima versione della CRL
- I server dei servizi verificano la validità delle credenziali confrontando il numero di serie della credenziale con la CRL più aggiornata

### 2.2.6 PoW: Puzzle Parametrizzato

Quando l'utente cerca di connettersi al server erogatore di un servizio, deve innanzitutto risolvere un puzzle parametrizzato. Per la sua costruzione si è scelto SHA-256, ottenendo  $Z$ , un sottoinsieme del codominio della CRHF, composto quindi da stringhe di 256 bit. L'obiettivo è quello di chiedere all'utente di trovare un valore  $x$  tale che  $H(rand||x) \in Z$ .

## 2.3 Specifiche degli algoritmi utilizzati

### 2.3.1 Gen

L'algoritmo  $Gen(1^n)$  è utilizzato per generare la coppia di chiavi pubblica e privata. Tramite questo algoritmo si va ad istanziare il problema del logaritmo discreto ( $DLog$ ) ottenendo  $\mathbb{G}_q$ ,  $q$ ,  $g$  come informazioni, dove:

- $\mathbb{G}_q$  è un gruppo ciclico di ordine primo  $q$
- $q$  è il numero di elementi presenti nel gruppo
- $g$  è un generatore del gruppo

A questo punto, è possibile selezionare un certo valore  $x \in \mathbb{Z}_q$  e calcolare  $y = g^x \bmod q$ . La chiave pubblica, indicata con  $pk$ , è:

$$\langle G_q, q, g, y \rangle$$

La chiave privata, indicata con  $sk$ , è:

$$\langle G_q, q, g, x \rangle$$

### 2.3.2 Sign<sub>sk</sub>

$Sign_{sk}(m)$  è un algoritmo di firma digitale, in particolare, si fa riferimento allo schema di firma di Schnorr. Quest'ultimo, prendendo in input un messaggio  $m$  ed una chiave privata  $sk$ , restituisce una coppia  $(z, a)$ , rappresentante la firma indicata generalmente con  $\sigma$ , dove:

- $a$  rappresenta un contributo casuale del gruppo, tale che  $a = g^r \bmod q$ , con  $r \in \mathbb{Z}_q$
- $z = r + H(y||a||m)x$ , dove  $H(\cdot)$  è un *random oracle*, implementato con SHA-256

### 2.3.3 Verify<sub>pk</sub>

La funzione  $Verfy_{pk}(m, \sigma) : \{0, 1\}$ , prendendo in input una chiave pubblica  $pk$ , un messaggio  $m$  e una firma  $\sigma = (z, a)$ , verifica che  $\sigma$  sia una firma valida per  $m$ , ciò avviene verificando che  $g^z \equiv ay^c \bmod q$ .

## **2.4 Protezione della Privacy e Integrità delle Credenziali**

Per proteggere la privacy degli utenti e l'integrità delle credenziali:

- L'utilizzo di firme digitali e connessioni TLS garantisce la sicurezza e l'integrità delle comunicazioni
- Le credenziali sono emesse come documenti X.509 e contengono solo le informazioni strettamente necessarie per l'accesso ai servizi, riducendo l'esposizione di dati personali
- L'identità dell'utente non è rivelata al server del servizio qualificato attraverso i documenti X.509 delle credenziali, a meno che non sia strettamente necessario
- Le credenziali sono firmate digitalmente dalle autorità di rilascio, garantendo che non possano essere alterate o falsificate

## **2.5 Conclusione**

La soluzione proposta mira a raggiungere un compromesso tra efficienza, trasparenza, confidenzialità e sicurezza. Il sistema descrive dettagliatamente le azioni delle parti oneste coinvolte, garantendo che le credenziali siano emesse e verificate in modo sicuro e trasparente. L'uso di certificati X.509 firmati digitalmente dalle autorità di certificazione (CA) assicura che le credenziali siano autentiche e non alterabili. Inoltre, l'adozione del protocollo TLS per le comunicazioni e dello schema di firma di Schnorr per l'autenticazione, minimizza il coinvolgimento di terze parti fidate durante l'accesso ai servizi, proteggendo la privacy e l'integrità degli utenti. Questa architettura bilancia le esigenze di sicurezza e confidenzialità con la necessità di un sistema efficiente e utilizzabile, riducendo al minimo i rischi associati agli attacchi su larga scala e garantendo un alto livello di affidabilità e trasparenza.

---

---

# CAPITOLO 3

---

## WORK PACKAGE 3

In questo capitolo si andrà a sottoporre ad analisi la soluzione proposta nel WP2 tenendo conto delle proprietà espresse all'interno del WP1, ovvero *confidenzialità*, *integrità*, *trasparenza* ed *efficienza*.

### 3.1 Confidenzialità

- **Malevolent Server**

Questa tipologia di attaccante mina la confidenzialità delle informazioni trasferite dall'utente al server erogatore del servizio, essendo quest'ultimo capace di visionare in chiaro le credenziali interne al certificato. Il server malevolo potrebbe, quindi, collezionare i dati al fine di utilizzarli per scopi illeciti.

- **Identity Thief**

Questo avversario, una volta ottenute le informazioni circa l'identità o la credenziale relativa ad un utente, può richiedere in maniera fraudolenta l'accesso ad un determinato servizio. Le informazioni sugli utenti possono essere ottenute per via traverse, come data leak o data breach, o tramite attacco man-in-the-middle. L'adozione di canali TLS di comunicazione, impedisce tale tipo di attacco; inoltre, pur avendo possesso di informazioni personali, non è possibile dimostrare l'identità se non si possiede anche la chiave privata, in quanto l'algoritmo di verifica della firma di Schnorr richiede tale passaggio.

- **The Eavesdropper & Data Miner**

Questo tipo di malintenzionati possono mettersi in ascolto sul canale di comunicazione

cercando di carpire informazioni personali con l'intento di raccoglierle o rivenderle. Anche questi tipi di attacco non hanno successo, in quanto, come detto in precedenza, i canali di comunicazione sono tutti canali sicuri TLS.

- **Evil Insider**

L'attaccante in questione può condurre attacchi sia verso utenti sia verso autorità o server, poiché è un operatore interno. Nello specifico, per quanto concerne i danni arrecati agli utenti, l'*Evil Insider* può effettuare un'esfiltrazione dei dati personali o delle credenziali per fini illegittimi.

- **The Data Harvesters**

Questi attaccanti raccolgono grandi quantità di dati per analisi o rivendita. Nonostante la raccolta di dati in sé non sia sempre malevola, può portare a violazioni della privacy se i dati sono utilizzati senza il consenso dell'utente.

Analizzando le proprietà descritte nel WP1, è possibile notare che:

1. **C1:** questa proprietà viene garantita in quanto, grazie alla struttura di gestione delle informazioni personali, l'utente fornisce tramite certificato i propri dati anagrafici solo alle CA fidate, mentre ai server invia i certificati contenenti solo la credenziale necessaria per l'accesso al servizio specifico
2. **C2 - C3:** le proprietà in esame vengono garantite grazie alla presenza di un canale di comunicazione basato su TLS sia tra l'utente e le CA sia tra l'utente e i server
3. **C4:** tale proprietà è garantita **a patto che l'utente non ceda volontariamente o venda le proprie chiavi private**, associate alle chiavi pubbliche presenti nella CIE e/o sulle credenziali. Tuttavia, in codesti casi, le precauzioni adottate possono intervenire in favore, in quanto:
  - *se l'utente non cede volontariamente* le proprie chiavi private, può richiedere una revoca dei certificati di cui ha notato un abuso, di modo da associare a questi ultimi delle nuove chiavi segrete;
  - *se l'utente cede volontariamente* queste informazioni non è possibile in alcun modo intervenire

## 3.2 Integrità

- **Evil Insider**

Un insider malevolo mira ad alterare i dati critici sui server, come le credenziali degli utenti o i registri delle transazioni, compromettendo l'integrità del sistema. L'insider ha un accesso privilegiato e può modificare i dati senza essere immediatamente rilevato.

- **Identity Thief**

Un ladro di identità può utilizzare credenziali rubate per accedere ai servizi e alterare i dati dell'utente legittimo o per falsificare la propria identità mascherandola con quella della vittima. Ad esempio, un attaccante di questo tipo potrebbe accedere al conto bancario di un utente e trasferire fondi senza autorizzazione.

- **The Cryptographer**

Utilizzando attacchi crittografici avanzati, questo attaccante potrebbe alterare dati durante la trasmissione, compromettendo l'integrità delle informazioni. Potrebbe, ad esempio, utilizzare attacchi di tipo "man-in-the-middle" per alterare i messaggi scambiati tra due parti.

- **Crypto Conspirators**

Questo gruppo di attaccanti potrebbe collaborare per compromettere i protocolli crittografici, permettendo la modifica dei dati durante la trasmissione. Ad esempio, potrebbero introdurre una vulnerabilità in un protocollo di crittografia utilizzato per la comunicazione sicura tra utenti e server. Un caso d'esempio potrebbe essere rappresentato dalla possibilità che un gruppo di hacker riesca ad inserire una backdoor nel software crittografico utilizzato da una compagnia, ottenendo la capacità di alterare documenti riservati senza essere scoperti.

Analizzando le proprietà relative all'integrità, si ha:

1. **I1:** per quanto riguarda questo aspetto, si possono identificare due scenari:
  - (a) nel primo caso, si assumono come onesti tutti i componenti del sistema (utenti, CA, server). In questo caso, non vi è alcun pericolo per l'integrità delle informazioni circolanti nei canali di comunicazione;
  - (b) nel secondo caso, invece, assumendo un possibile attaccante all'interno dei server, non è possibile garantire l'integrità. Si immagini un caso d'esempio tale per cui un attaccante interno (*Evil Insider*) ad un server, visionando in chiaro le credenziali degli utenti, raccolga le informazioni sensibili al fine di strutturare il profilo dell'utente e rivendere i dati personali o utilizzarli per scopi illeciti.

A causa della centralizzazione del sistema non è possibile difendersi totalmente da questo tipo di attacchi, ma è possibile limitarli parzialmente tramite l'adozione di log, che vadano a mantenere una lista delle transazioni effettuate, e controlli incrociati sulle attività svolte
2. **I2:** l'adozione di diversi protocolli sicuri, come TLS per la trasmissione e lo schema di firma per i certificati, previene modifiche non autorizzate ai dati durante la

trasmissione. Anche se un attaccante intercettasse i dati, non potrebbe alterarli senza che la modifica venga rilevata

3. **I3:** tale proprietà è garantita dalla presenza, sulla CIE, di un certificato interno, firmato dall'IPZS, e un PIN associato alla scheda. Si è inoltre supposto che l'IPZS (che è una parte fidata) abbia rilasciato pubblicamente la procedura con cui la CIE viene costruita e firmata
4. **I4:** questa proprietà viene garantita dalla firma, apposta da ogni utente sui certificati X.509v3, generata tramite lo schema di firma di Schnorr. Infatti, nel caso in cui il documento fosse stato rubato, sarebbe verificabile la sua validità ma non l'appartenenza al malintenzionato poiché, a fronte di una *Zero Knowledge Proof*, non sarebbe in grado di provare ad un *Verifier* che è a conoscenza dell'informazione segreta (la chiave privata  $sk$ ) legata alla chiave pubblica, derivabile dalla firma, presente all'interno del documento

### 3.3 Trasparenza

- **Anti-Tech Theorist**

Diffondendo informazioni false o allarmistiche, può minare la fiducia degli utenti nella trasparenza del sistema. Si pensi ad un esempio in cui un oppositore all'avanzamento tecnologico vada a diffondere voci su vulnerabilità inesistenti in un sistema di autenticazione per generare panico e sfiducia tra gli utenti.

- **Evil Insider**

Questo attaccante interno può manipolare i sistemi di autenticazione o di rilascio delle credenziali per nascondere le proprie attività malevole. Ad esempio, un dipendente con accesso privilegiato potrebbe modificare i log di accesso per nascondere il furto di dati sensibili, rendendo così difficile il rilevamento dell'attacco.

- **The Cryptographer**

Utilizzando tecniche avanzate, questo attaccante potrebbe rendere difficile la verifica delle credenziali rilasciate, compromettendo la trasparenza del processo di autenticazione. Un attaccante, ad esempio, può falsificare dei certificati digitali, rendendo difficile per gli utenti e gli amministratori distinguere i certificati validi da quelli compromessi.

- **Identity Thief**

Operando sotto identità rubate, questo attaccante rende complesso il tracciamento delle sue attività malevole, compromettendo la trasparenza generale del sistema.

Circa le proprietà inerenti alla trasparenza, è denotabile che:

1. **T1:** tale proprietà viene rispettata tramite l'utilizzo di librerie e tool open source, le quali permettono un'analisi accurata dei protocolli scelti, al fine di poter fornire prove di assenza di manomissione, dando motivo agli utenti di fidarsi del sistema
2. **T2:** il rilascio di credenziali spetta solo alle CA, prese come parte fidata, limitando così la dipendenza da organi terzi e favorendo la velocità di verifica delle credenziali da parte dei server. Inoltre, l'adozione di log per le transizioni potrebbe migliorare ulteriormente questo scenario, sfavorendo la generazione di credenziali false o alterate
3. **T3:** questa proprietà è garantita dall'architettura del sistema, in quanto ogni server, che eroga un servizio specifico, richiede delle credenziali all'utente, che desidera connettersi ad esso. Tuttavia, è possibile, da parte dell'utente, fornire solo ed esclusivamente le credenziali necessarie per l'accesso al servizio, evitando così di divulgare informazioni superflue alla fruizione
4. **T4:** il sistema prevede l'utilizzo dello schema di firma di Schnorr in accoppiata con canali TLS, che garantisce l'autenticità della fonte tramite *Zero Knowledge Proof*, per i certificati digitali. Inoltre, i certificati hanno una scadenza fissa e, come ulteriore protezione, nel caso in cui un utente notasse un abuso di credenziali legate al proprio profilo, potrebbe richiederne la revoca (e le CA hanno le CRL).

## 3.4 Efficienza

- **The Denier**

Un attaccante che nega l'autenticità dei certificati potrebbe sovraccaricare il sistema con richieste di verifica, riducendo l'efficienza. Un esempio potrebbe riguardare un possibile attacco di tipo denial-of-service (DoS), il quale potrebbe inondare i server di richieste di verifica delle credenziali.

- **Anti-Tech Theorist**

Generando panico e disinformazione, questo attaccante può causare un aumento delle richieste di assistenza e verifiche, rallentando il sistema. La diffusione di notizie allarmistiche può portare a un sovraccarico dei canali di supporto.

- **Evil Insider**

Introducendo backdoor o manipolando politiche di accesso, questo attaccante può causare rallentamenti o inefficienze nei processi di autenticazione e accesso ai servizi. Ad esempio, un dipendente malintenzionato potrebbe alterare le configurazioni del server di autenticazione, causando un aumento dei tempi di risposta e una riduzione della reattività del sistema.



- **The Cryptographer**

L'ipotesi di un attacco ai canali TLS da parte di questo attaccante può introdurre sovraccarichi nei server e nella rete, causando un aumento del tempo di elaborazione per ogni richiesta e rallentando significativamente il sistema.

Analizzando le proprietà di efficienza descritte nel WP1, è possibile notare che:

1. **E1:** l'utilizzo di credenziali per l'accesso ai server, rende modulare la fruizione dei servizi, permettendo anche l'utilizzo di requisiti complessi (es. "Residente a Salerno" OR "Nato a Salerno").
2. **E2:** la gestione delle credenziali tramite certificati X.509v3 permette una verifica rapida, evitando così la creazione di code di attesa e una reattività maggiore del sistema. Ciò è possibile, poiché le CA sono intese come parte fidata e i certificati delle credenziali sono firmati da queste ultime.
3. **E3:** anche qui, l'utilizzo di certificati X.509v3, permette un'alta velocità di autenticazione, senza sacrificare la sicurezza nella comunicazione, in quanto la CIE è protetta dal PIN, usato per ottenerne una firma ECDSA del certificato interno
4. **E4:** Il processo di autenticazione ed identificazione dell'utente richiede lo scambio di 3 messaggi con il server più la verifica della firma. Ciò che potrebbe rallentare in parte il sistema è l'adozione del puzzle parametrico prima dell'accesso ad un servizio, difatti questo arreca un costo computazionale al server e richiede l'interazione dell'utente. Tuttavia, si è preferito l'utilizzo di questo algoritmo per evitare eventuali problemi legati ad attacchi di tipo DoS/DDoS, favorendo quindi la sicurezza e l'integrità dell'architettura di sistema

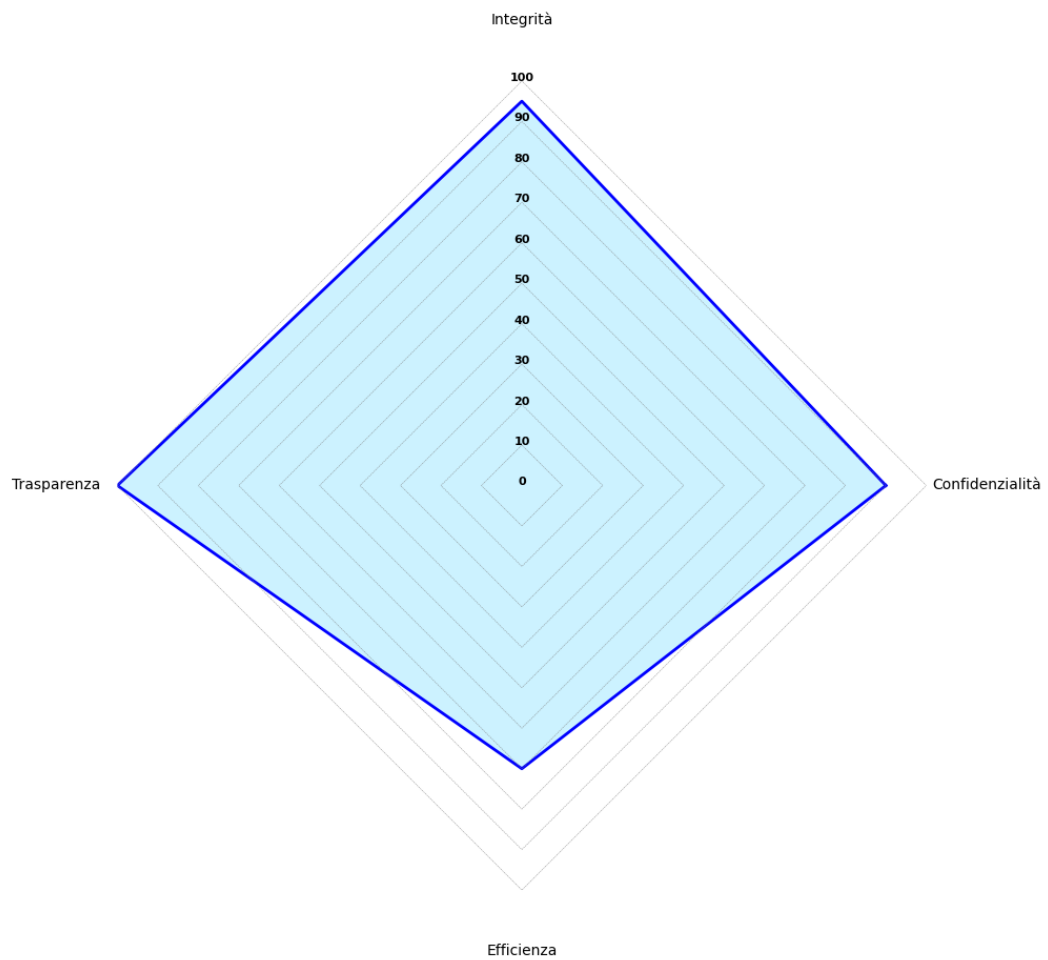


Figure 3.1: Grafico radar relativo alle proprietà

---

# CAPITOLO 4

---

## WORK PACKAGE 4

### 4.1 Introduzione al Work Package 4

In questa sezione si è analizzata l'implementazione pratica delle specifiche definite nel (WP2), all'interno del quale è stato delineato un quadro teorico per la generazione e l'utilizzo delle credenziali attraverso un sistema di identificazione basato su certificati X.509v3 e aggiornamento delle liste di revoca degli stessi (CRL). È stato sviluppato uno script in linguaggio **bash**, utilizzando il set di comandi predisposto dalla libreria **openssl**, per implementare queste specifiche, garantendo che il sistema proposto possa gestire in modo sicuro ed efficiente l'autenticazione e l'accesso ai servizi digitali. In particolar modo, ci si è focalizzati sulla generazione di coppie di chiavi privata-pubblica per CA, utente e server, dopodiché il focus è stato spostato su configurazione dei certificati, gestione delle credenziali e connessioni TLS, assicurando che ogni componente del sistema funzioni in conformità con quanto descritto nel *WP2*.

### 4.2 Inizializzazione e Configurazione Iniziale

#### 4.2.1 Variabili e Cartelle

Il codice inizia definendo variabili per le cartelle principali e creando queste cartelle se non esistono già. Queste includono:

- **cert.management**: la cartella principale per la gestione dei certificati.
- Sottocartelle:

- **keys**: contiene le coppie di chiavi private-pubbliche di CA, utente e server e anche il file di configurazione personalizzato per le credenziali;
- **certificates**: contiene le richieste di certificato e i certificati generati per utente e server;
- **ca.certificates**: contiene i certificati dell'autorità di certificazione (CA), la lista dei numeri seriali relativi ai certificati emessi, il file di configurazione per i certificati X.509v3 e la possibile CRL.

### 4.2.2 Definizione del PIN CIE

Il PIN CIE (`user_pin`) è impostato di default a 1234.

## 4.3 Generazione di Chiavi RSA

Per la generazione di chiavi RSA è stata predisposta una funzione apposita (`generate_rsa_keys()`), il cui funzionamento prevede:

- la generazione di una chiave privata su 2048 bit;
- l'estrazione di una chiave pubblica corrispondente.

### 4.3.1 Generazione chiave ECDSA

È stata altresì predisposta una funzione per la generazione di chiavi ECDSA (`generate_ecdsa_keys()`), la quale non viene utilizzata concretamente nello script, ma è possibile implementarla per ottenere il certificato interno alla CIE, sfruttando un modulo dedicato e un lettore di smart-card.

## 4.4 Funzioni di Gestione di Certificati

### 4.4.1 Creazione e Firma di Richieste di Certificato

**create\_certificate\_request**: crea una richiesta di certificato utilizzando una chiave specificata e la sottopone alla CA per la firma;

**sign\_certificate\_request**: firma una richiesta di certificato con la chiave della CA, generando il certificato effettivo.

Queste funzioni utilizzano OpenSSL per gestire le operazioni di crittografia asimmetrica e la creazione di certificati X.509v3.

### 4.4.2 Creazione di Certificati con Credenziali

`create_credentialed_certificate`: crea un certificato che include dati di credenziali, come specificato dall'utente tramite il file di configurazione personalizzata `credential_ext.cnf`. Questa funzione include la generazione di una richiesta di certificato e l'inclusione di una firma ECDSA, la quale è ottenuta tramite un'interazione con la CIE.

## 4.5 Gestione di Server TLS

### 4.5.1 Avvio e Connessione al Server TLS

`start_tls_server`: avvia un server TLS utilizzando la chiave privata e il certificato dello stesso, validato dalla CA;

`connect_to_tls_server`: permette la connessione a un server TLS da parte di un client, utilizzando la chiave privata e il certificato di quest'ultimo, verificando l'autenticità dello stesso tramite il certificato della CA.

## 4.6 Configurazione di Apache con HTTPS (TLS)

### 4.6.1 Configurazione del Server Apache

`configure_apache_tls`: configura Apache per utilizzare HTTPS, copiando le chiavi e i certificati necessari nei percorsi appositi del sistema operativo e configurando il file di virtual host per utilizzare SSL/TLS <sup>1</sup>.

## 4.7 Interazioni Utente-CA e Utente-Server

`user_ca_interaction`: gestisce l'interazione tra l'utente e la CA per la richiesta e la firma di certificati;

`user_server_interaction`: gestisce l'interazione tra l'utente e il server.

## 4.8 Configurazione della CA

### 4.8.1 Definizione delle Politiche di Certificazione

Viene creato un file di configurazione (`ca_config.cnf`) che specifica le politiche di certificazione per la CA, inclusi i vincoli di chiave e l'identificatore del soggetto.

---

<sup>1</sup>La configurazione del server Apache risulta funzionante solo su sistemi Red-Hat based, nello specifico le piattaforme di testing utilizzate si basano su Fedora 39 e Fedora 40.

## 4.9 Flusso Principale dello Script

### 4.9.1 Sequenza Operativa

Lo script inizia generando le chiavi per la CA, l'utente e il server, poi procede a:

- creare un certificato per la CA, se non è già presente, altrimenti riutilizza lo stesso, se ancora valido;
- gestire l'interazione tra l'utente e la CA per richiedere e firmare certificati;
- generare un certificato per il server;
- avviare un server TLS e consentire l'interazione tra utente e server attraverso connessioni sicure;
- configurare Apache per utilizzare HTTPS;
- consentire all'utente di interrompere e rimuovere i file del server.

## 4.10 Revoche e Pulizia

Lo script include anche operazioni per la revoca dei certificati e la generazione di elenchi di revoca (CRL), se necessario, oltre a pulire e rimuovere i file temporanei o non più necessari.

## 4.11 Output Script

```
1 Generating RSA keys for ca_key...
2 RSA keys generated for ca_key.
3
4 Generating RSA keys for user_key...
5 RSA keys generated for user_key.
6
7 Generating RSA keys for server_key...
8 RSA keys generated for server_key.
9
10 Creating CA certificate...
11
12 Creating certificate request for ca_key...
13 Certificate request created for ca_key.
14
15 CA certificate (./ca_certificates/ca_cert.pem) not found. Creating...
16 CA certificate created.
17
```

## 4. WORK PACKAGE 4

---

```
18 Creating certificate request for server_key...
19 Certificate request created for server_key.
20
21 Signing certificate request ./certificates/server_request.csr...
22 Certificate request signed and certificate generated.
23
24 Creating certificate request for user_key...
25 Certificate request created for user_key.
26
27 Signing certificate request ./certificates/user_request.csr...
28 Certificate request signed and certificate generated.
29
30 Creating certificate request for user_key...
31 Certificate request created for user_key.
32
33 Creating credentialed certificate for ./certificates/user_cert.pem...
34 Credentialed certificate created for ./certificates/user_cert.pem.
35
36 TLS server started with PID 141804
37
38 Schnorr verification in progress...
39 Schnorr verification successfully completed
40
41 Using default temp DH parameters
42   0 items in the session cache
43   0 client connects (SSL_connect())
44   0 client renegotiates (SSL_connect())
45   0 client connects that finished
46   0 server accepts (SSL_accept())
47   0 server renegotiates (SSL_accept())
48   0 server accepts that finished
49   0 session cache hits
50   0 session cache misses
51   0 session cache timeouts
52   0 callback cache hits
53   0 cache full overflows (128 allowed)
54 CONNECTED(00000003)
55 ---
56 Certificate chain
57  0 s:CN=example.com
58   i:CN=CA-Root, O=MyOrganization, C=IT
59   a:PKEY: rsaEncryption, 2048 (bit); sigalg: RSA-SHA256
60   v:NotBefore: Jul  7 17:29:40 2024 GMT; NotAfter: Jul  7 17:29:40 2025 GMT
61  1 s:CN=CA-Root, O=MyOrganization, C=IT
62   i:CN=CA-Root, O=MyOrganization, C=IT
63   a:PKEY: rsaEncryption, 2048 (bit); sigalg: RSA-SHA256
64   v:NotBefore: Jul  7 17:29:40 2024 GMT; NotAfter: Jul  7 17:29:40 2025 GMT
```

#### 4. WORK PACKAGE 4

```
65 ---
66 Server certificate
67 -----BEGIN CERTIFICATE-----
68 MIIDHTCCAgWgAwIBAgITQvUYODdX4TI5MzYvFVAkpZX6/zANBgkqhkiG9w0BAQsF
69 ADA4MRAwDgYDVQQDDAdDQs1Sb290MRcwFQYDVQQKDA5NeU9yZ2FuaXphdG1vbjEL
70 MAKGA1UEBhMCSVQwHhcNMjQwNzA3MTcyOTQwWhcNMjUwNzA3MTcyOTQwWjAWMRQw
71 EgYDVQQDDAtleGtfcGx1LmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoC
72 ggEBAKjxaBNm4qgVm/aRx6uVhfZtS0zkv2pxVwj1MqmpSMDSDpuxDKbd7Z7JE3s4
73 Asp0o4LIcBzX//zuq6oE9MdIUJPBuCwQgxFOnrhX5e9t5/ezRY6j0hgYK9da/40S
74 qLnVk32GcFZqPjHBT8gkNb6RBz4dQQRyQVYzp7LNVcXM9xWepoUEjBd0UaH2zVh6
75 7ZC3cxViIYvd3fyuhUUrYasxoRn7ie2c0ZVufh/ncnC2V9mueDnsdTKgc1zOienW
76 kZI6P9QYS2QwchkwlZX19Exm+6V/kQXtoS7wur9d17nMH1Hi0PH0LmjBzmMS1ylh
77 fdjshAHkf3qzaRdpwnfhIjU+DRsCAwEAAaNCMEAwHQYDVRO0BBYEFN3uJRIJxgFf
78 UvEnDkcOpEM/BcbrMB8GA1UdIwQYMBaAFcQN1VcYn97v7yNsULo8jqUcmYOFMAOG
79 CSqGS1b3DQEBcWUAA4IBAQArc9LyZC5JP0cZQTVvsKDMiyvDlfdMsf30mW92/U
80 kFeeqWw6+AIDDB5nn0gTev6D7DBAzTYe5vX3VJQgkzoBuSqySRB4eHqMfw0y3B1
81 5GwSKef0zbF7qvgQKAu351hS93NoDviqeCs9Qo2M47M57v3080tWRJNwOSz6S0gY
82 SCzJ/VAyn0Dfc1HhLxM8G8N7Z1KEEEzND4+5mX9cHAWUZwvGxoQ5aWVQlog8Gh14
83 G2YHxFF9oc4sd10VEJbjcUWm1tScJ7ZoVhdgOubewAyXmarqJWHHKBZBm3Qw7uHf
84 ubdniGL5NI18YNMmTHEviG6EH2AK3TY7MiVb7c8aGMpk
85 -----END CERTIFICATE-----
86 subject=CN=example.com
87 issuer=CN=CA-Root, O=MyOrganization, C=IT
88 ---
89 No client certificate CA names sent
90 Peer signing digest: SHA256
91 Peer signature type: RSA-PSS
92 Server Temp Key: X25519, 253 bits
93 ---
94 SSL handshake has read 2165 bytes and written 382 bytes
95 Verification error: unsuitable certificate purpose
96 ---
97 New, TLSv1.3, Cipher is TLS_AES_256_GCM_SHA384
98 Server public key is 2048 bit
99 This TLS version forbids renegotiation.
100 Compression: NONE
101 Expansion: NONE
102 No ALPN negotiated
103 Early data was not sent
104 Verify return code: 26 (unsuitable certificate purpose)
105 ---
106 ---
107 Post-Handshake New Session Ticket arrived:
108 SSL-Session:
109     Protocol : TLSv1.3
110     Cipher   : TLS_AES_256_GCM_SHA384
111     Session-ID: 496427A89F32953216A2C8BD408054E5480EA5D45F3F2F838BF0D9F7AB38A5A5
```



#### 4. WORK PACKAGE 4

```
112 Session-ID-ctx:
113 Resumption PSK:
114 6F71B53AA69DF193880420907C5C165F42CF8B164AF5996BE53F957FA6C1EB5A895C5DAA2FB9FCDD00552F7D2E060E5C
115 PSK identity: None
116 PSK identity hint: None
117 SRP username: None
118 TLS session ticket lifetime hint: 300 (seconds)
119 TLS session ticket:
120 0000 - c2 03 99 47 4d 5d bf 6e-51 ac 79 7d d2 6a 57 5d ...GM].nQ.y}.jW]
121 0010 - e1 69 73 b2 18 07 2a fb-18 19 b3 9f 08 2c 21 2d .is...*.....!-
122 0020 - 5e ca ed 41 bd 14 cf 32-d2 1c 02 14 4d 83 a3 77 ^..A...2....M..w
123 0030 - 48 41 97 0d 8d 1a 3b 2f-19 84 46 a0 e6 49 c6 95 HA....;/..F..I..
124 0040 - 41 60 8b b0 aa eb 8e 89-15 49 b6 f8 31 0b 05 b5 A'.....I..1...
125 0050 - 1c eb 0b 8a 8c d2 51 e4-43 09 90 aa f3 7f ec 14 .....Q.C.....
126 0060 - 68 3b 5c f0 3a ff 27 20-93 2e fc 0b d9 d6 9e c3 h;\...' .....
127 0070 - cb 36 c7 00 ce 44 d1 6a-1b b1 a9 d5 f9 5f 59 2f .6...D.j....._Y/
128 0080 - f2 2c 42 ba e7 46 0d 09-83 a8 d4 f9 ee c1 b0 3b .,B..F.....;
129 0090 - ec e1 80 45 e6 92 62 39-b7 58 23 4e 4d 5f b6 cc ...E..b9.X#NM_..
130 00a0 - 31 a1 05 0d ce 52 40 d2-f2 d6 5f 05 51 0a fb a4 1....R@..._.Q...
131 00b0 - 10 e2 6e 23 43 cd d2 35-59 00 17 0c e9 16 eb dd ..n#C..5Y.....
132 00c0 - 2b 13 58 28 b6 d7 45 7d-5f b4 2a a8 a5 a3 d3 63 +.X(..E}_.*....c
133 00d0 - a9 13 3b c3 50 ac 65 45-b7 30 04 01 aa db ee 65 ...;.P.eE.O.....e
134 00e0 - 5f f6 1e a6 e9 aa 42 49-94 43 06 8e e9 76 c2 63 _.....BI.C...v.c
135
136 Start Time: 1720382226
137 Timeout : 7200 (sec)
138 Verify return code: 26 (unsuitable certificate purpose)
139 Extended master secret: no
140 Max Early Data: 0
141 ---
142 read R BLOCK
143 ---
144 Post-Handshake New Session Ticket arrived:
145 SSL-Session:
146 Protocol : TLSv1.3
147 Cipher : TLS_AES_256_GCM_SHA384
148 Session-ID: 92EC9111655A1049A01863B4417C48F732F62F129EE38F11D4F85655DF66803B
149 Session-ID-ctx:
150 Resumption PSK:
151 63783ABA445605C90D7D97DDF503B652C300C1FF16F21B6B3331621EC7557F6E0276B88E296F0E7D366111C3CE306944
152 PSK identity: None
153 PSK identity hint: None
154 SRP username: None
155 TLS session ticket lifetime hint: 300 (seconds)
156 TLS session ticket:
157 0000 - c2 03 99 47 4d 5d bf 6e-51 ac 79 7d d2 6a 57 5d ...GM].nQ.y}.jW]
158 0010 - 37 bc 6d 44 33 bb d4 51-44 c5 34 58 fd bc dc 0a 7.mD3..QD.4X....
```

#### 4. WORK PACKAGE 4

---

```
159 0020 - 3c 99 bf 6c 28 69 90 14-e7 3a bf 93 33 20 2c 32 <..l(i.....3 ,2
160 0030 - 11 19 99 7e 50 0d 66 9d-28 4d f5 b4 bf 6e b8 81 ...~P.f.(M...n..
161 0040 - fa 4a c7 bf 4a 2e 9c ce-b7 57 de b9 dc e3 3f f5 .J..J....W....?.
162 0050 - b6 e8 52 2f 67 3b a8 a3-28 71 f6 96 13 0b 2a fb ..R/g;..(q....*.
163 0060 - 61 12 8e 8e 58 b4 53 d7-a7 7c 49 39 1b fb 78 cf a...X.S..|I9..x.
164 0070 - 23 c0 95 c1 f1 00 54 26-58 c5 a9 22 c0 10 d4 c9 #.....T&X.."....
165 0080 - 16 f9 b7 9d 90 2c b1 67-5b d3 c0 6d 97 29 09 11 .....,.g[.m.)..
166 0090 - 18 d2 68 b3 4e a0 94 2c-d2 05 7f 3b e7 f6 85 33 ..h.N.,...;...3
167 00a0 - e4 9a 1a 63 28 2c af fd-07 0d f7 e9 c2 cc 95 46 ...c(,.....F
168 00b0 - dd 1b 40 20 b0 6d d9 89-3d 59 54 51 6f a0 11 c3 ..@ .m..=YTQo...
169 00c0 - 53 68 09 c1 49 fd 2a ee-60 a8 6f a1 73 ec 83 0c Sh..I.*.'o.s...
170 00d0 - 2e 60 dd f3 50 a3 da 2e-f0 98 a2 ca 33 0b 85 c1 .'..P.....3...
171 00e0 - a6 3c 28 78 26 f6 51 86-86 d7 43 98 f8 5a 41 95 .<(x&.Q...C..ZA.
172
173 Start Time: 1720382226
174 Timeout : 7200 (sec)
175 Verify return code: 26 (unsuitable certificate purpose)
176 Extended master secret: no
177 Max Early Data: 0
178 ---
179 read R BLOCK
180 closed
181
182 Hello from the TLS client
183
184 Connection to TLS server with PID 141804 terminated successfully
185
186 Do you want to delete the server files? (y/n)
187 Server files deleted
```