



[projectorigin2023@gmail.com](mailto:projectorigin2023@gmail.com)

## *Verbale* esterno del 14 luglio 2023

<b>Versione</b>	0.1.0
<b>Responsabile</b>	
<b>Redattori</b>	Andreetto Alessio
<b>Verificatori</b>	Ibra Elton
<b>Uso</b>	Esterno
<b>Destinatari</b>	<i>Project Origin</i> Prof. Vardanega Tullio Prof. Cardin Riccardo

### **Descrizione**

*Verbale* riguardante il meeting tenuto il 14 luglio 2023 con l'azienda *InfoCert S.p.A*

## Registro delle modifiche

Vers.	Data	Autore	Ruolo	Descrizione
0.1.0	2023-07-16	Ibra Elton	Verificatore	Verifica documento
0.0.1	2023-07-16	Andreetto Alessio	Analista	Redazione documento

## Indice

<b>1</b>	<b>Informazioni generali</b>	<b>3</b>
1.1	Dettagli sull'incontro . . . . .	3
<b>2</b>	<b>Ordine del giorno</b>	<b>4</b>
2.1	File di configurazione idp-config.json e verifier-config.json . . . . .	4
2.2	Store di una credenziale . . . . .	4
2.3	Controllo di flusso . . . . .	4

# 1 Informazioni generali

## 1.1 Dettagli sull'incontro

- **Luogo:** Incontro telematico tramite piattaforma *Microsoft Teams*<sub>g</sub>;
- **Data:** 14-07-2023;
- **Ora di inizio:** 11:00;
- **Ora di fine:** 12:20;
- **Presenze:**
  - *InfoCert*:
    - \* Manzi Paolo
    - \* Porro Davide
  - *Project Origin*:
    - \* Andreetto Alessio (asincrona)
    - \* Beschin Michele
    - \* Bobirica Andrei Cristian
    - \* Corbu Teodor Mihail
    - \* Ibra Elton
    - \* Lotto Riccardo (asincrona)

## 2 Ordine del giorno

1. File di configurazione idp-config.json e verifier-config.json
2. Store di una credenziale
3. Controllo di flusso

### 2.1 File di configurazione idp-config.json e verifier-config.json

Con questi 2 file di configurazione, ci riferiamo alla parte di verifica delle credenziali. Nello specifico, idp-config.json va ad effettuare la mappatura tra le verificable credential e i profili interni. Questo passaggio serve agli autenticator per effettuare il mapping tra i dati. Questo meccanismo risulta fondamentale nella comunicazione tra wallet e verifier. Noi, da wallet, offriamo una verificabile id a un verifier, e lui ci fornisce un access token per accedere ai suoi servizi. Il problema riscontrato da parte nostra è l'ottenimento di tale token. Da parte dell'azienda, ci è stato consigliato, inoltre, di utilizzare dei DID di tipo key in ogni campo dati che necessita di DID sia esso qualsiasi attore del sistema. Discutendo con l'azienda inoltre si è giunti alla conclusione che durante l'issuing di una credenziale, il campo DID del verifier fa riferimento all'issuer in quanto verificatore dei documenti immessi al momento di richiesta credenziale.

### 2.2 Store di una credenziale

Discutendo con l'azienda si è giunti alla conclusione che lo store della credenziale non va eseguito su un unico repository fornito da SSI Kit. Abbiamo già creato 2 container:

- walletssikit,
- issuing-Kit.

In questa modo, una volta istanziata una credenziale, possiamo andarla a memorizzare nel repository wallet. Tuttavia, questa procedura risulta ancora poco chiara per il nostro gruppo, quindi intendiamo approfondirla nel breve periodo per comprenderla meglio.

### 2.3 Controllo di flusso

Andremo a creare una simulazione del flusso, presente nella documentazione di waltid, tramite una collection di chiamate. Tale flusso verrà discusso insieme all'azienda al fine di chiarire i dubbi che abbiamo riscontrato sulla strutturazione dei container Docker per seguire il flusso indicato in OpenID.