Guida alla configurazione del RAO Pubblico

Generazione certificati RAO

(IP server 192.168.1.29)

1. Clonare il Repository con l'utility per la generazione:
   `git clone https://github.com/psmiraglia/spid-compliant-certificates.git`
2. Accedere alla directory con lo script in python:
   `cd python/spid_compliant_certificates/generator`
3. installare i moduli necessari:
   `pip install spid-compliant-certificates`
4. generare i certificati:
   `bin/spid-compliant-certificates generator --key-size 3072 --common-name "Comune di Miane" --days 365 --entity-id https://raopubblico.comunedimiane.it/public/metadata --locality-name Miane --org-id "PA:IT-c_f190" --org-name "Comune di Miane" --sector public`

   Saranno stati generati i file: key.pem, csr.pem, crt.pem che ci servono per accreditarci

1. Determino l'hash con il comando:
   `sha1sum csr.pem`
   Questo Hash ci servirà per il modulo di accreditamento

Container per la Web App di gestione del RAO Pubblico

1. Clonare il nuovo Rao pubblico:
   `git clone https://github.com/Sielte/rao-pubblico.git`
2. Creare il volume permanente che conterrà il database di richieste dell'ente e i dati di accesso:
   `docker volume create "volumemiane" &> /dev/null || true`
3. Compilare file /compose/local/rao/rao.env:

   SIGN_URL=https://rao-signature.sitbl.it/v2/
   BASE_URL=https://raopubblico.comunedimiane.it
   SECRET_KEY=newodpoewmdpomewdmewomwmepdwdwmmd3322542432fde3
   DATABASE_NAME=/data/comunedimiane.sqlite3
   MAIL_LOG_LEVEL=ERROR
   PORTAL_LOG_LEVEL=INFO
   AGENCY_LOG_LEVEL=INFO
   SECRET_KEY_ENC=mewmdewlmdewmpdjmw4320u0324329040328fwefewf
   RAO_NAME=Comune di Miane

4. Prima di costruire i container rendo eseguibile il seguente file:
   `chmod +x compose/local/rao/start`

5. Quindi rendo scrivibile la seguente directory:
   `chmod 2777 rao/data`
6. In questo comando è importante modificare il nome del container (–name), il nome del volume e la porta di ascolto:
   ```
   docker run -d --add-host rao-signature.sitbl.it:192.168.1.29
   --name "comunedimiane" --env-file "./compose/local/rao/rao.env"
   --mount type=volume,source="volumemiane",target="/data" -p
   "8001:8000" "rao-app:latest" "/start"
   ```

Inoltre se si aggiungono nuovi enti la porta 8001 dovrà essere incrementata di conseguenza (8002,8003...)

Per mantenre un unico container per la firma delle richieste (rao-signature.sitbl.it) aggiunto la entry –add-host in modo da non far inviare la richiesta di collegamento verso internet.

Virtual Host Apache e SSL LetsEncrypt

1. Creare il file con il nome rao-pubblico-<nome ente>.conf
2. Inserire quanto segue nel virtualhost:
   “‘

   ServerName raopubblico.comunedimiane.it ErrorLog ${APACHE_LOG_DIR}/raopubblicomiane_error.lo
   CustomLog ${APACHE_LOG_DIR}/raopubblicomiane_access.log combined Alias /public /var/www/metadata/miane Require all granted ProxyPass http://localhost:8001/ flushpackets=on ProxyPassReverse http://localhost:8001 Options All AllowOverride All order allow,deny allow from all

Questo codice indica ad apache che la chiamata diretta all'url indicato in ServerName deve essere rediretta sulla porta 8001 (in questo caso) che è il container Docker che gestisce la Webapp, inoltre se l'url ha come sotto path la voce /public deve andare a leggere nella directory indicata cosa importante per eventualmente inserire dei file statici.

> Chiaramente nel caso di aggiunta di un ente nuovo, sarà necessario modificare opportunamente il numero di porta e il dominio.

A questo punto usando LetsEncrypt e il suo tool certbot è possibile andare ad impostare i certificati SSL per il nuovo dominio, genererà in automatico il file virutalhost per la porta 443 https configurando in automatico apache.