draft-urien-coinrg-iose
(Internet Of Secure Elements)
"*Architecture of secure elements in the Internet whose resources are identified by URIs*"

Pascal.Urien@Telecom-Paris.fr

About Secure Elements

- A Secure Element contains a certified microcontroller and embedded software. Its *Evaluation Assurance Level* (EAL) is up to EAL6+, given a scale ranging from one to seven, according to Common Criteria (CC) standards.
- 9 billions Secure Elements shipped in 2021.
- Today 8/16 bits CPUs, up to 10KB SRAM, 100KB non volatile memory + crypto processors
  - Next generation: 32bits core, 60MHz clock, up to 2048KB FLASH, 64KB SRAM + crypto processors
- Legacy communication: serial (ISO7816) , emerging I^2C, SPI
- Binary Encoding Rules: small packets (about 256 bytes), i.e. ISO7816 APDUs
- Programming environments: Javacard (a subset of Java) six billions devices deployed every year, other languages (C…).
- Secure software management(list/delete/upload) framework: Global Platform with Secure Channel Protocol (SCP), using ISO7816 APDUs.

Why connecting Secure Elements to Internet ?

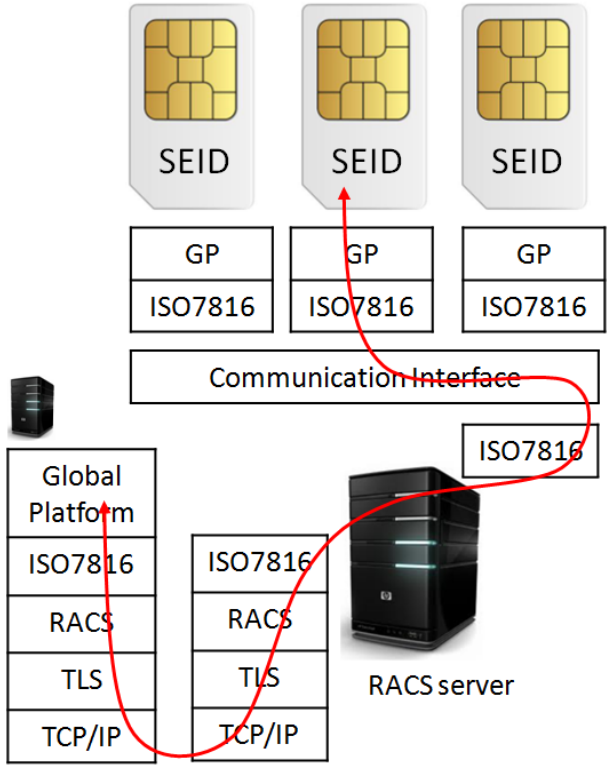
- On-line trusted cryptographic resources for internet user.
  - Identified by Uniform Resource Identifiers.
- Issues:
  - Additional processor (server) is required with network interface and TCP/IP connectivity.
  - Global Platform support for on-demand applications.
  - Protocol to access to secure element resources.
  - Secure element naming.
  - Attestation procedure for on-demand applications.

draft-urien-coinrg-iose

- IOSE creates cryptographic resource URIs
  - **schemeS://sen:psk@server.com:port/?query**
  - in which:
    - server.com:port is the TCP/IP socket for the front TLS server
    - sen is the secure element name (TLS server name)
    - psk is the pre-shared-key value (256 bits)
    - schemeS (S meaning secured by TLS) identifies the syntax used by the application embedded in the secure element
    - query is a request, encoded according to scheme syntax
- IOSE Server Components
  - The Administration plane: RACS (TCP Deamon)
  - The Service Plane: TLS-SE (TCP Daemon)
  - The Attestation procedure. It transfers secure element control to user. Its security relies two properties:
    - 1) secure elements can not be cloned,
    - 2) and they manage only one TLS session at a given time.

Administration Plane: RACS

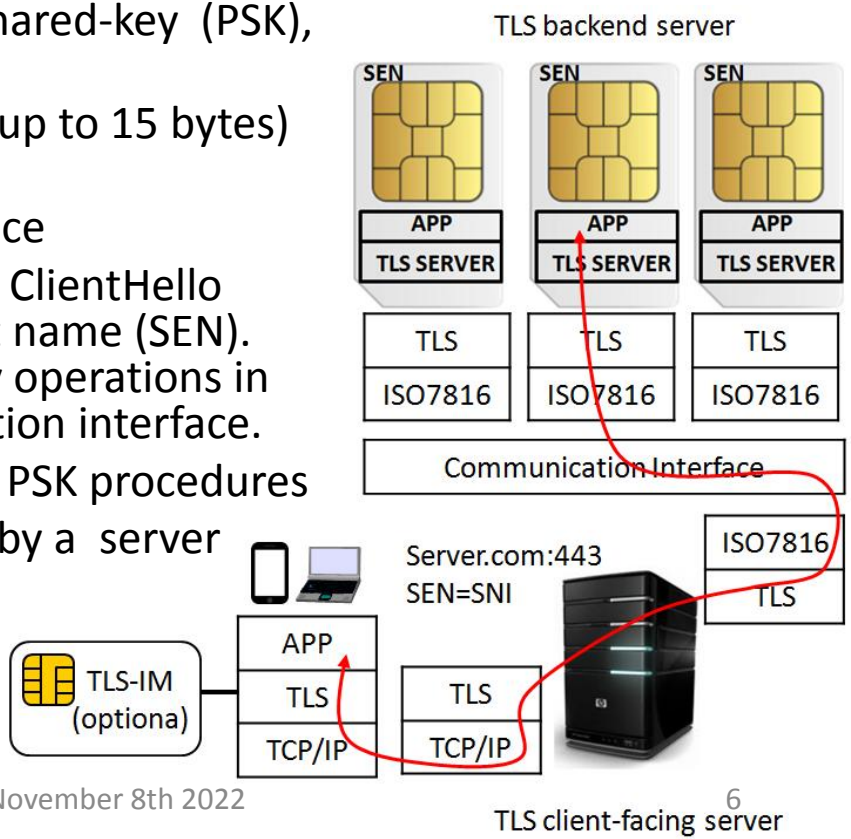
- Secure element applications are securely downloaded thanks to Global Platform (GP) protocols, working over ISO7816.
- Remote APDU Call Secure (RACS[1]), transports GP protocols, over TLS sessions
- End entities are mutually authenticated by X509 certificates
- In this context secure elements are identified by *Secure Element Identifier* (SEID), inserted in RACS messages.



[1]https://datatracker.ietf.org/doc/draft-urien-core-racs/

Service Plane: TLS-SE

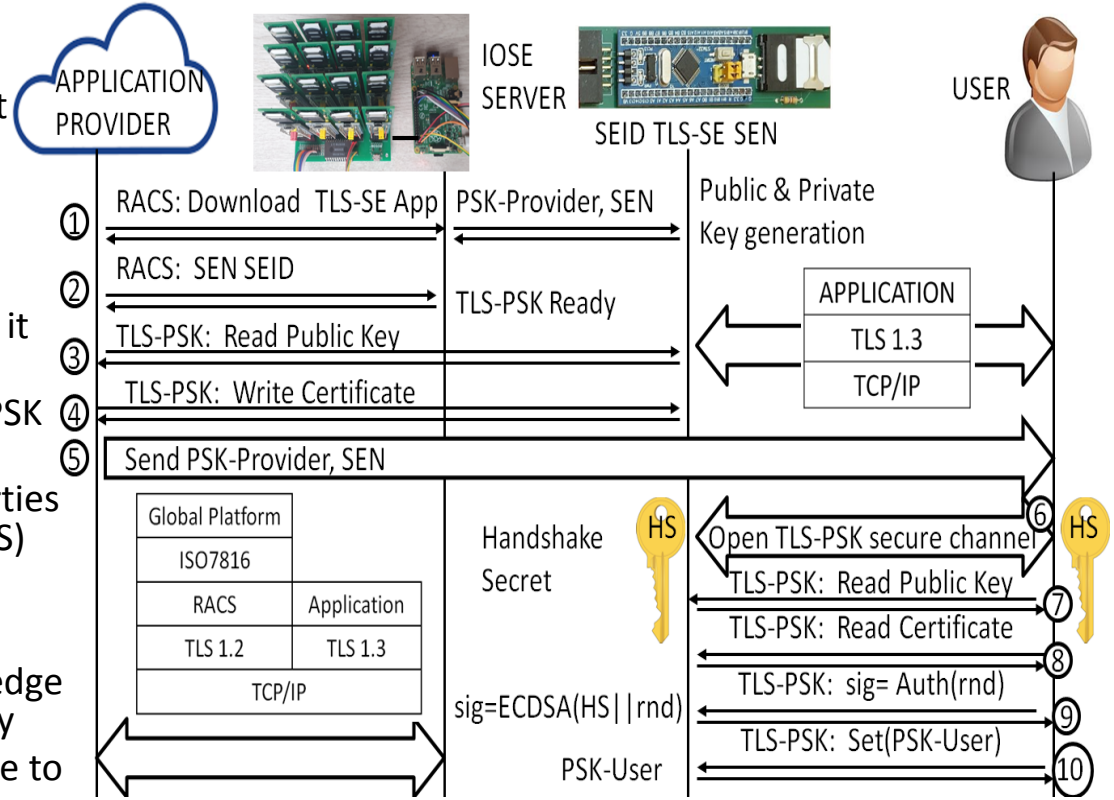
- Secure Elements run TLS1.3 server[1], using pre-shared-key (PSK), and a server name (SEN).
- The server name is found in the historical bytes (up to 15 bytes) of the secure element Answer To Reset (ATR).
- TLS packets are transported over ISO7816 interface
- The client-facing server (server.com) finds in the ClientHello Server Name Indication (SNI) the secure element name (SEN). Thereafter it performs segmentation/reassembly operations in order transport TLS packet over the communication interface.
- Optional TLS Identity Module (TLS-IM[2]) compute PSK procedures
- Secure element is the backend server, identified by a server name



[1]https://datatracker.ietf.org/doc/draft-urien-tls-se/
[2]https://datatracker.ietf.org/doc/draft-urien-tls-im/

On-demand Application & Attestation

- 1-The application is downloaded in the secure element SEID, thanks to RACS. It generates a pair of public/private keys
- 2- The app provider binds the secure element name (SEN) to its SEID
- 3- It reads the public key via TLS-PSK
- 4- It generates a certificate and pushes it over TLS-PSK session
- 5- The user receives the app provider PSK and SEN
- 6- It opens a TLS-PSK session, both parties compute the TLS Handshake Secret (HS)
- 7-It reads the public key
- 8- It reads the certificate
- 9-It checks the secure element knowledge of TLS handshake secret and private key
- 10- And Finally it modifies the PSK value to PSK-User



Open Resources

- TLS-SE for javacard (JC 3.0.4)
  - https://github.com/purien/TLS-SE
  - ASCII command lines over TLS
- IOSE Server v5 (Windows, Ubuntu, Raspberry Pi)
  - https://github.com/purien/IoSE
  - 2 TCP Daemons, RACS + TLS
  - Multiple communication interfaces
    - PC/SC, I^2C, SIM Array

Could this draft become a working group item ?

(4) Research on potential new transport protocol, new privacy and security mechanisms required or enabled by in-network compute.

P. Urien, "Personal HSM, Privacy for Subscribers in 5G/6G Networks," 2022 1st International Conference on 6G Networking (6GNet), 2022, pp. 1-6, doi: 10.1109/6GNet54646.2022.9830453.
P. Urien, "Internet Of Secure Elements Concepts And Applications. : Invited Paper," 2022 Seventh International Conference On Mobile And Secure Services (MobiSecServ), 2022, pp. 1-6, doi: 10.1109/MobiSecServ50855.2022.9727207.
P. Urien, "Demonstrating Internet Of Secure Elements Server," 2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC), 2022, pp. 923-924, doi: 10.1109/CCNC49033.2022.9700553.
P. Urien, "A New IoT Trust Model Based on TLS-SE and TLS-IM Secure Elements: A Blockchain Use Case," 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC), 2021, pp. 1-2, doi: 10.1109/CCNC49032.2021.9369485.