Trust model and purpose of token-based authorization for Scenario D

Ericsson 2020-01-23

Prelude on token-based authorization in general

Purpose of token-based authorization (Rel-15 and 16)

Token-based authorization is used to define and enforce trust domains inside the network.

Examples:

- Slice isolation (e.g. verticals slice isolated from public safety slice)
- Separation of virtual operators
- Separation of regions
- Separation of SUPI ranges
- Separation of use cases (IoT, MBB)
- NF type isolation (e.g. specific UDR only to be accessed by UDMs)

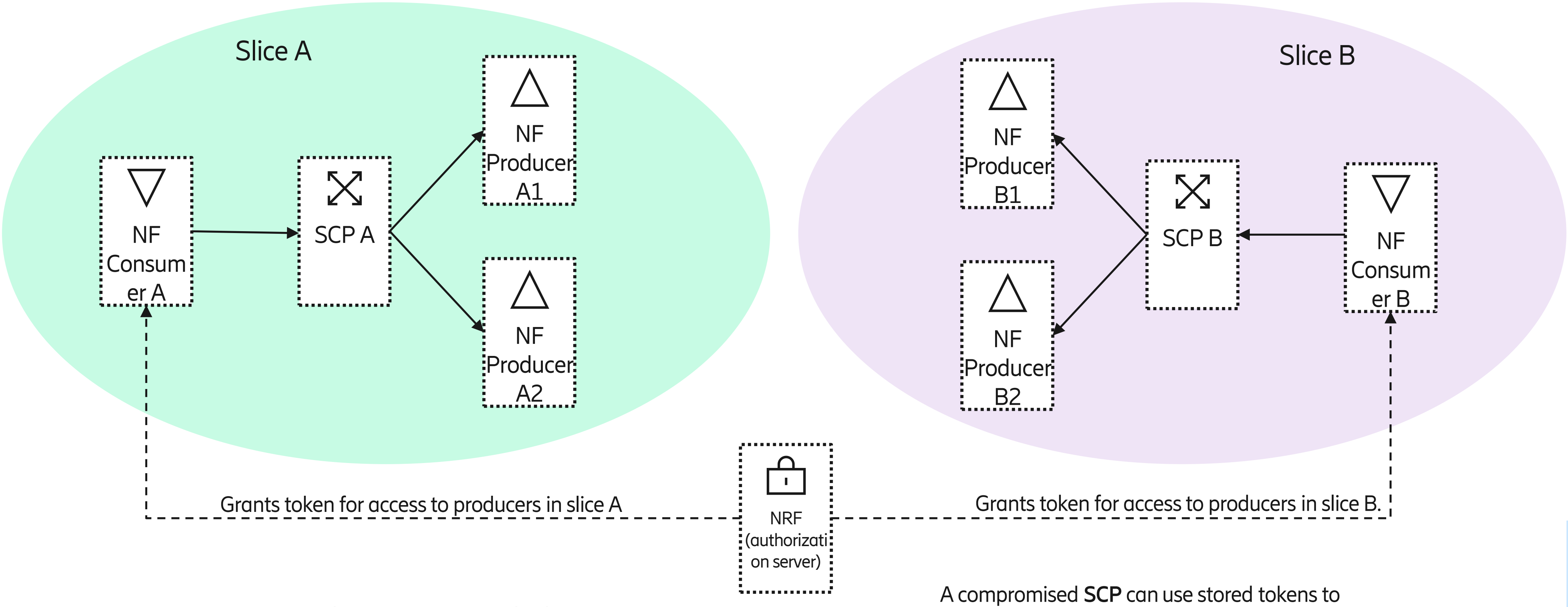
Provisioning of authorization information in the NRF (Rel-15 and 16)

1. NF producers provide authorization information in their profile (see Table 6.1.6.2.2-1 in TS 29.510)
   - allowedPlmns
   - allowedSnpns
   - allowedNfTypes
   - allowedNfDomains
   - allowedNssais
2. Out of band, e.g. by O&M (proprietary)

Token-based authorization for Scenario D

Example 1

In this example, the NRF grants tokens directly to the consumer.
Separate SCPs are deployed in each slice.


Slice A
NF Consumer A
SCP A
NF Producer A1
NF Producer A2
Slice B
NF Producer B1
NF Producer B2
SCP B
NF Consumer B
Grants token for access to producers in slice A
NRF (authorization server)
Grants token for access to producers in slice B.

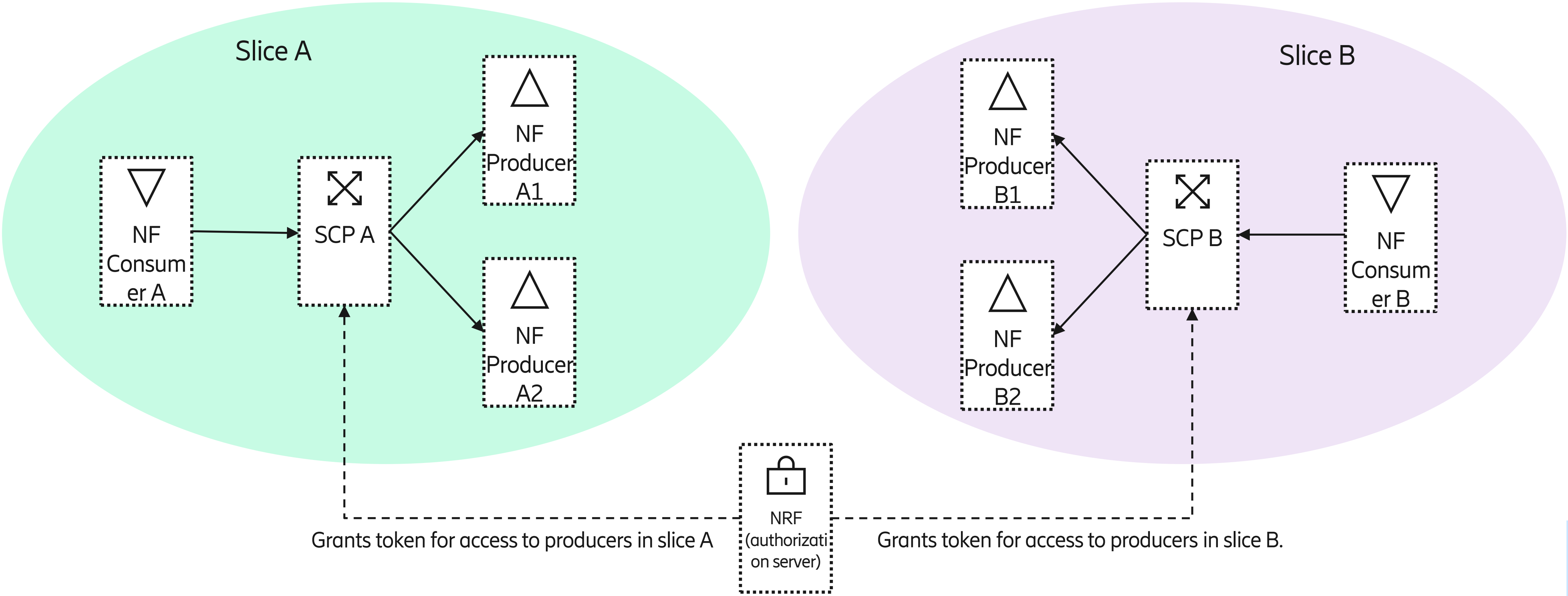
NRF controls separation between slices A and B, by granting tokens to consumers only for service access to producers in their own slice.

A compromised **SCP** can use stored tokens to impersonate consumers in its slice. However, this is the only way in which the SCP can extend its privileges. A compromised consumer or SCP cannot access producers in another slice.

In this example, token-based authorization is used for **slice isolation**.

Example 2

In this example, the NRF grants tokens to the SCP.
Separate SCPs are deployed in each slice.



Grants token for access to producers in slice A

Grants token for access to producers in slice B.

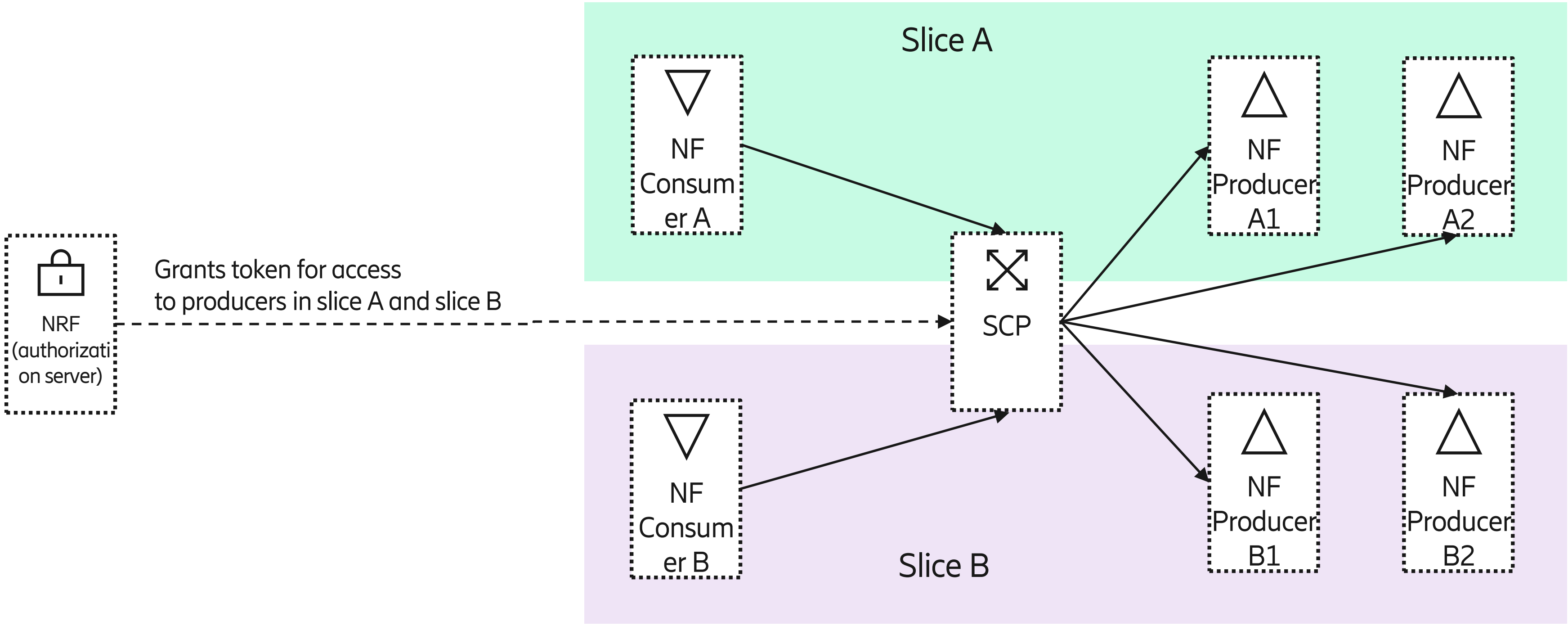
NRF can control separation between slices A and B **if it is aware of which SCP serves which slice**.

A compromised **SCP** can impersonate consumers in its slice. A compromised consumer or SCP cannot access producers in another slice.

In this example, token-based authorization is used for **slice isolation**.

Example 3

In this example, the NRF grants tokens to the SCP.
The slices are not completely separated, because the SCP belongs to both slices.
This example shows only two slices, but SCPs could span over more than two slices.

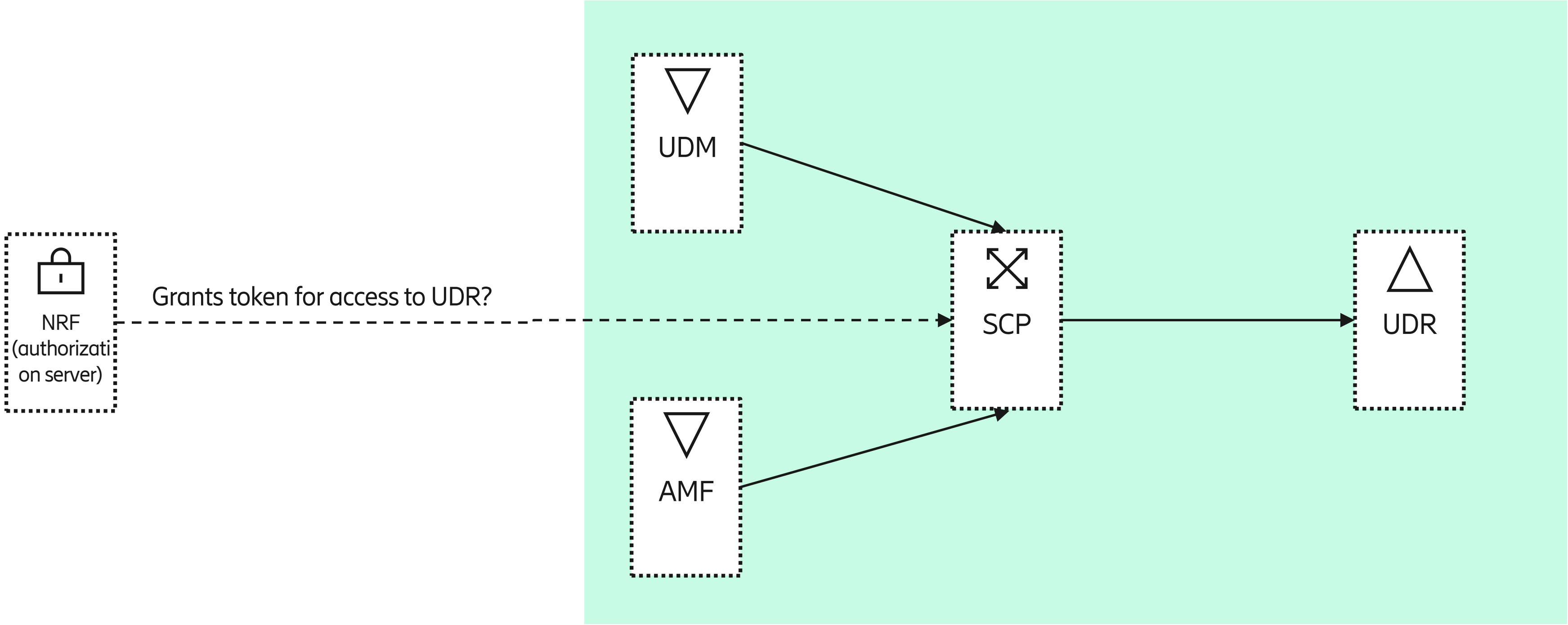


NRF cannot control separation between slices A and B.
Token-based authorization fails to serve its purpose (slice isolation). NRF may not be able to follow authorization information provided by producer in their profile.

A compromised **SCP** can impersonate consumers in slices A and B. A compromised consumer or SCP cannot access producers in other slices than A and B.

Example 4

In this example, the NRF grants tokens to the SCP.
The UDR only allows access by consumers of type UDM.
The SCP represents consumers of different types.



Should the **NRF** grant the SCP access to the UDR?

If yes, the UDR may receive service requests from NFs of wrong type.

If no, the UDM may not be able to receive services from the UDR.

Token-based authorization fails to serve its purpose (NF type isolation). NRF may not be able to follow authorization information provided by producer in their profile.

In this example, token-based authorization is used for **isolation between NF types**.

Observations and way forward 1/2

Observations on security:

Without mutual authentication between NRF and consumer, NRF does not have any verified information about the consumer. Anyone can update the profile of the consumer.

If the NRF is not aware of which SCP is allowed to represent which consumer, but the SCP requests the token on behalf of the consumer, anyone can request tokens on behalf of the consumer.

Way forward:

- Mutual authentication between NRF and consumer.
- If the SCP can request tokens on behalf of the consumer, the NRF needs to be aware of which SCP is allowed to represent which consumer.

Otherwise, token-based authorization will not be able to provide the separation of trust domains that is its purpose.

Observations and way forward 2/2

Observations on practical considerations:

In Scenario D, the consumer does not have the result of the discovery request.

The authorization information for a specific consumer can be rather complicated, e.g. because of the elaborated requirements in the producer profiles.

The NRF needs some guidance on the desired audience in the token request, however the consumer cannot provide this information.

Way forward:

In addition to the security requirements from the previous slide, the SCP's knowledge of the discovery result is needed to guide the NRF on the desired audience for the authorization token.

The desired solution for token-based authorization in Scenario D needs both

- **Confirmation by the consumer of the validity of the request, and**
- **Guidance on the audience of the token from the discovery result.**

Proposed solution options

1. Consumer requests authorization code from NRF, authorization code is used in access token request by SCP.
   - requires mutual authentication between consumer and NRF

2. Consumer issues authorization code and signs with its own private key, authorization code is used in access token request by SCP.
   - requires consumer private keys

3. Consumer includes in their profile which SCPs are allowed to represent it, SCP requests token, NRF checks whether SCP is allowed to represent consumer.
   - requires mutual authentication between consumer and NRF

4. Consumer provides to the NRF out of band which SCPs are allowed to represent it, SCP requests token, NRF checks whether SCP is allowed to represent consumer.
   - requires out of band mechanism