ProSe discovery using direct radio signals

Possible system architecture and required functionality

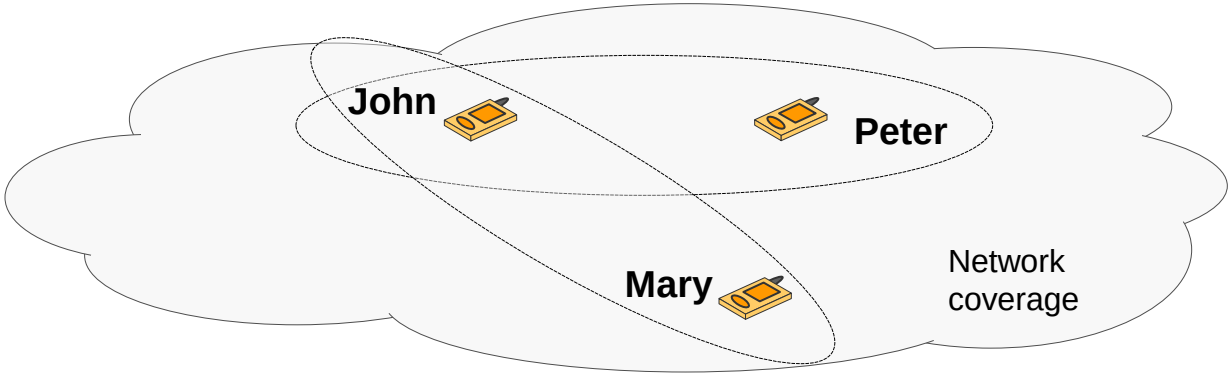
Intel

S2-130325

Introduction

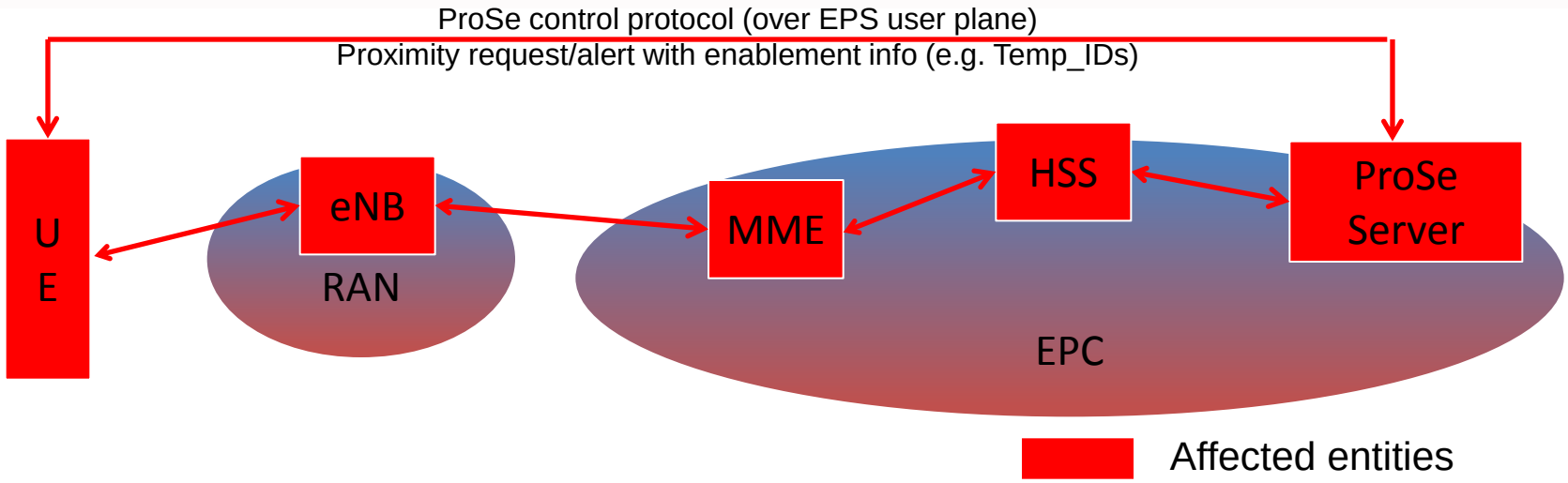
- The ProSe WID (S2-120395) has three broad objectives:
  - ***ProSe Discovery over E-UTRA***
  - ProSe Communication over E-UTRA
  - EPC support of ProSe Communication over WLAN
- The focus of this presentation is on the first item when ProSe discovery is performed using direct radio signals
- The objective of this presentation is to:
  - Discuss a possible architecture, required functionality and call flows for the basic use case (Restricted ProSe Discovery using direct radio signals)
  - Point to a possible clear-cut split between work in SA groups and RAN groups

Basic use case: Restricted ProSe Discovery using direct radio signals



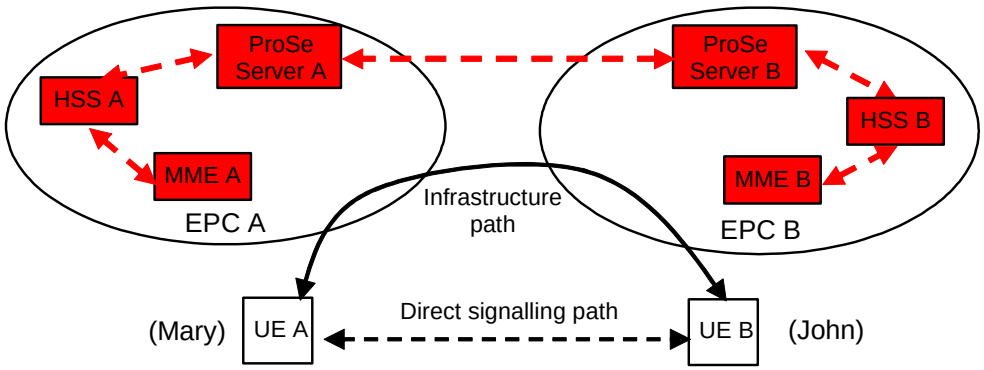
- Simplified description (from TR 22.803 clause 5.1.1):
  - Mary, Peter and John (subscribers of the same or different PLMNs) use a social networking application. In the context of this application:
    - Mary and John are friends
    - John and Peter are friends
    - Mary and Peter are *not* friends
  - Mary's UE detects (using direct radio signals or via the operator's network) that John's UE is in its proximity
  - John's UE detects that Mary's UE is in its proximity
  - John's UE detects that Peter's UE is in its proximity
  - Their respective social networking applications learn the proximity status of the friends and display the status in the contact list

Possible architecture



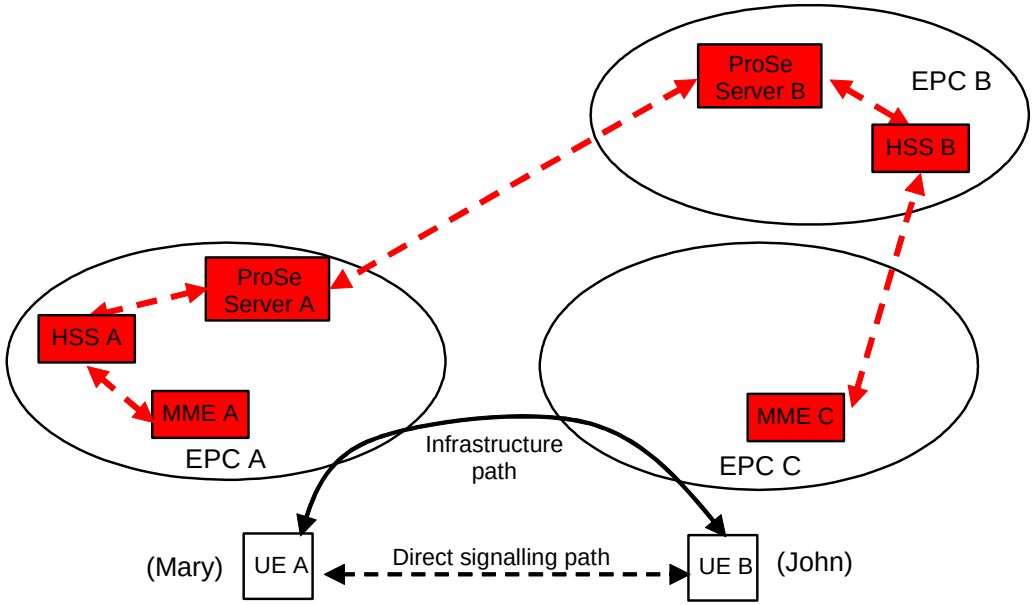
- ProSe Server – new functional entity residing in the EPC
  - Stores user's profile for the ProSe service
    - some functions can actually be hosted by a 3rd party App Server (see S2-130324 for similar discussion)
  - Assigns temporary radio identifiers (Temp_IDs) to be used by the UE for identification purposes on the radio and pushes them into HSS for authorisation purposes
  - Mediates in obtaining Temp_IDs of the user's "friends"
  - Communicates with ProSe Server peers in other PLMNs to support cross-PLMN ProSe discovery and roaming scenarios
  - May have interfaces towards the charging architecture
- The main purpose of Temp_ID is enabling privacy on the radio. The Temp_ID should be long enough to ensure uniqueness and short enough to minimise radio resource usage
  - e.g. may be comparable in size with IEEE MAC addresses (48 bits)

Architecture for cross-PLMN and roaming scenarios

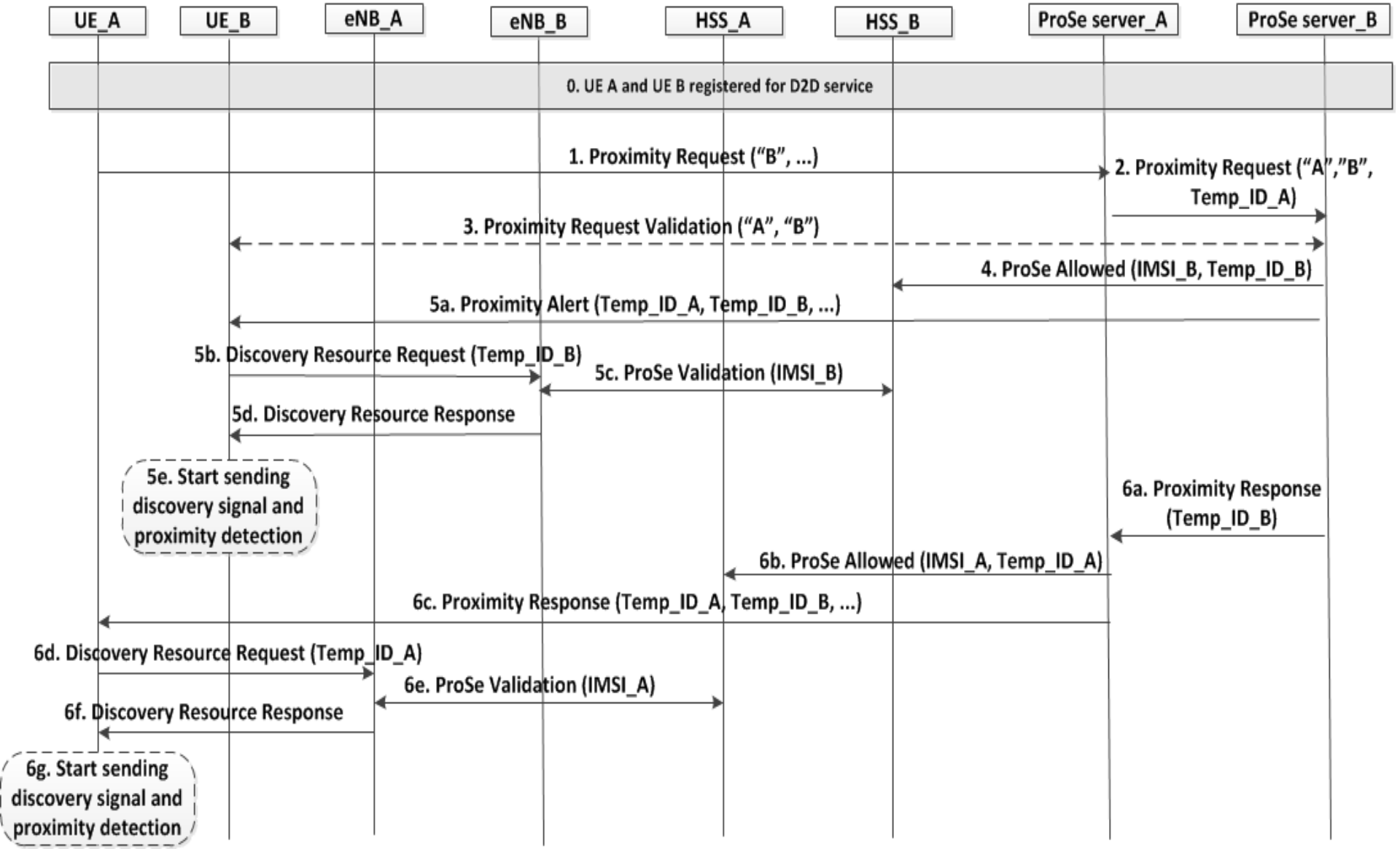


- Cross-PLMN scenario
  - User A and user B are subscribers of PLMN A and PLMN B, respectively
  - Each PLMN hosts its own ProSe Server

- Roaming scenario
  - In this scenario, user B, a subscriber of PLMN B, is currently roaming in PLMN C
  - The ProSe Server for user B resides in his home network i.e. in EPC B



Basic call flow (1/2)



Basic call flow (2/2)

- 0. UE A and UE B are registered for ProSe service with their respective ProSe Server.
- 1. When UE A wishes to discover UE B, UE A sends a Proximity Request message including “B” - a permanent application layer identifier of user B;
- 2. ProSe Server A identifies the ProSe Server that owns information for user B (i.e. ProSe Server B) by inspecting the application layer identifier “B”. ProSe Server A assigns a temporary radio ID for user A (Temp_ID_A) on-the-fly (if not already assigned) and forwards it to ProSe Server B in a Proximity Request message;
- 3. Depending on user B’s ProSe service profile, ProSe Server B optionally checks whether user B accepts user A’s request;
- 4. ProSe Server B assigns a temporary radio ID for user B (Temp_ID_B) on-the-fly (if not already assigned) and pushes it into HSS_B for validation purpose in the future;
- 8a. ProSe Server B sends a Proximity Alert message to UE B, including Temp_ID_B and Temp_ID_A;
- 8b. UE B sends a Discovery Resource Request message to its current eNodeB (eNodeB_B), including its assigned radio identifier (Temp_ID_B);
- 8c. eNodeB_B validates the request by making a check with HSS_B (via MME_B);
- 8d. eNodeB_B sends a Discovery Resource Response message to UE_B indicating the radio resources that UE should use for the beacon signal that will carry Temp_ID_B;
- 8e. UE_B starts sending its beacon signal (with Temp_ID_B) and starts seeking for a beacon signal carrying Temp_ID_A;
- 9a. ProSe Server B sends a Proximity Response message to ProSe Server A, including Temp_ID_B;
- 9b. If Temp_ID_A was assigned on the fly in step 5, ProSe Server A pushes it into HSS_A for validation purpose in the future;
- 9c. ProSe Server A sends a Proximity Response message to UE A, including Temp_ID_A and Temp_ID_B;
- 9d.-9g. Similar to steps 8b.-8e. for UE A

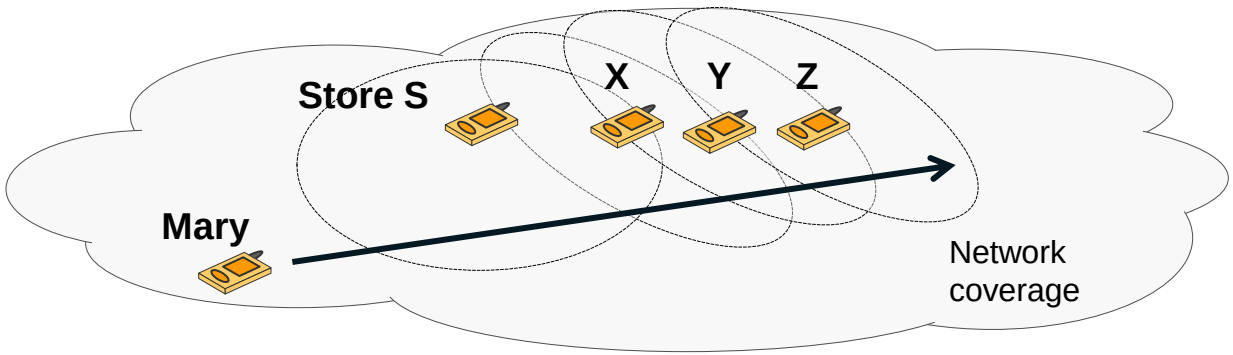
Tentative conclusions

- Functionality needed for the basic use case of Restricted ProSe Discovery using direct radio signals:
  - ProSe Server description (refer to slide #6)
  - Procedures for ProSe service registration, proximity request, etc. (UE-to-Server; Server-to-Server)
  - Expected impact on HSS, MME and eNodeB

- RAN groups to decide whether an efficient solution for discovery using direct radio signals and based on temporary radio identifiers (e.g. 48-bit long) is feasible
  - If such a solution is possible, there can be a clear-cut split between work in SA groups and RAN groups as far as ProSe Discovery is concerned
  - Recommend sending an LS to the RAN groups to this effect

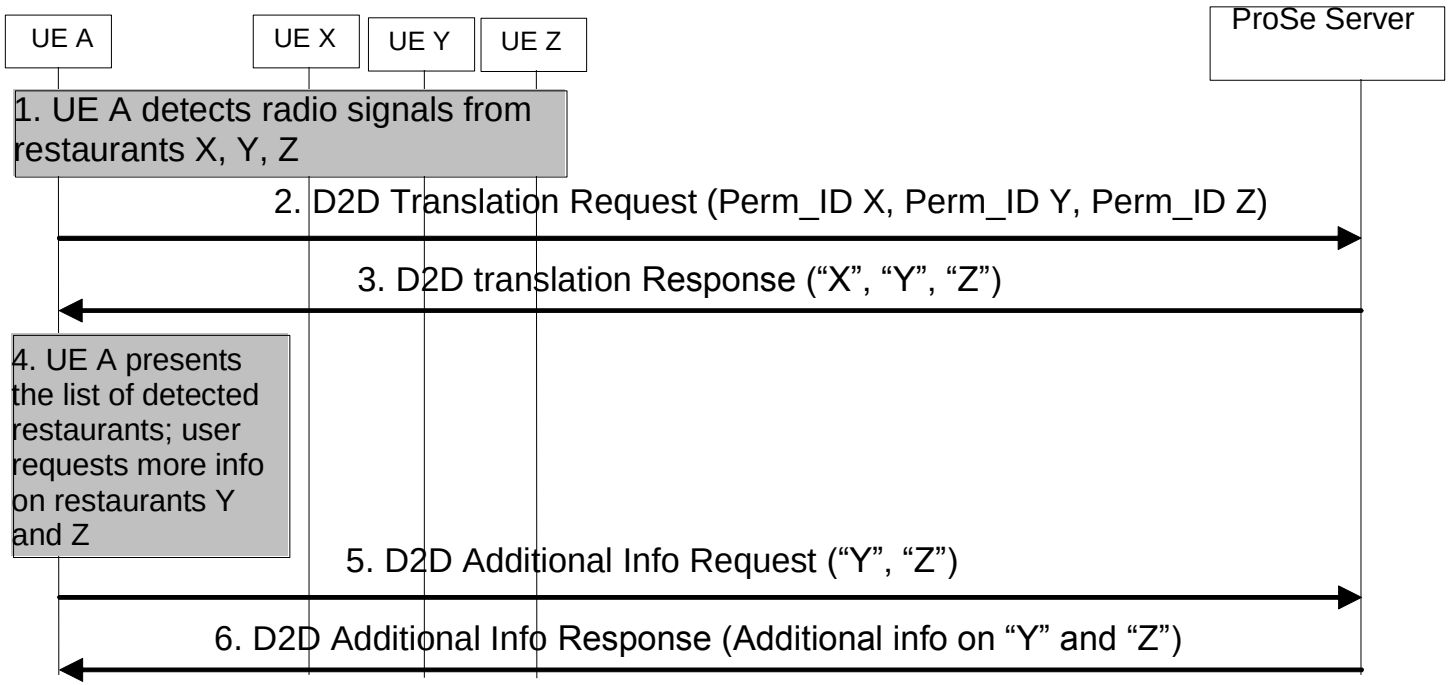
Backup

Considerations on Open ProSe Discovery (1/2)



- Simplified description (from TR 22.803 clause 5.1.2):
  - Mary uses a given application. In the context of this application it is assumed that:
    - Mary, Store S and Restaurants X, Y and Z use ProSe-enabled UEs
    - Mary and the owners of the store and restaurant UEs are subscribed to an operator service that allows them to use ProSe
    - There might be hundreds of other stores/restaurants with ProSe-enabled UEs in the vicinity of Mary
    - The operator has enabled the application to access this ProSe feature
  - As Mary walks into the neighbourhood where Store S is located, Mary is notified of the proximity of Store S. Mary then decides to look for a restaurant, and thus (e.g. following interaction with her application), Mary is notified of the proximity of Restaurant X, Y and Z
  - Mary is not notified of the proximity of other establishments which are not determined to be of interest according to the application she is using

Considerations on Open ProSe Discovery (2/2)



- To enable Open ProSe Discovery:
  - The static UEs (e.g. restaurants) are assigned long-lived or permanent radio identifiers (Perm_ID_X, Perm_ID_Y, Perm_ID_Z)
  - A portion of the Perm_ID (e.g. 8 bits out of 48) is used for encoding of identifier categories (e.g. “restaurants”, “store”, “hospitals”, etc.)
  - UE performs category-based filtering on the detected radio identifiers before presenting them to the user