Camara assumptions and missing concepts

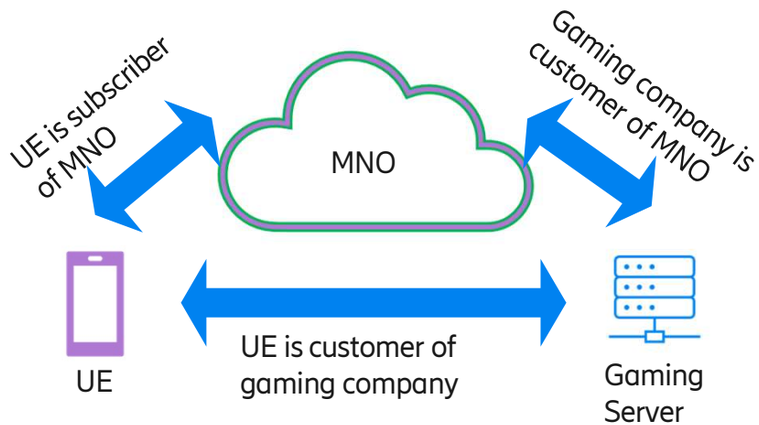
Contents

Background

- Camara is defining the QoD API with the purpose of enabling an external service consumer to manage the Quality of Service of certain data flows.
- The QoD API needs to identify the UE.
  - The current proposal is to identify UEs by its IP address (as seen from the Application Server). We are concerned that this will limit the use of the API and increase complexity for using the API.
  - Alternative ways to identify the UE should be investigated.
- Additionally, the QoD API does not provide means to identify the related contracted service.
  - It should be investigated the need for identifying the service related to the API call.
- Ericsson believes that these issues are relevant to most of the APIs that Camara intends to define.

The gaming example

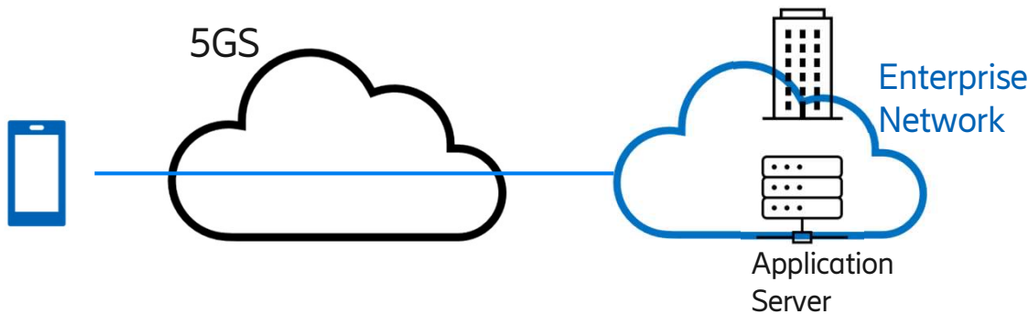
- Example: A consumer has a consumer subscription (e.g., eMBB) with the MNO. The consumer has acquired services from a gaming service provider. When the consumer starts playing the game, the gaming service provider invokes the QoD API for improving the QoS of the data to/from the gaming server.
  - Note that the gaming service provider does not own the UE subscription.
- The Mobile Network Operator needs to identify the API invoker a given API request is related to, for authorizing the usage of the API call.
  - There is an implicit assumption that authentication/authorization is done at the time or prior to invoking the API.



- In addition, the Mobile Network Operator needs to verify that the service used by the API is duly contracted according to the subscription.
  - In the gaming example:
    - The subscription allows for the improvement of QoS according to the connectivity service parameters (e.g., eMBB).
    - The service is duly charged.
- Conclusion: two aspects must be considered in API calls:
  - The API invoker must be authorized to use the service API.
  - The recipient of the service (e.g., the UE) must be authorized to receive the service (e.g., QoD modification).

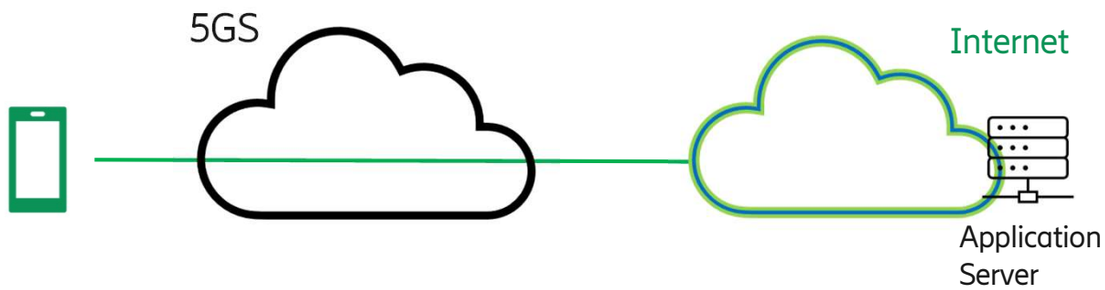
Network APIs need to support different scenarios

Enterprise owns the UE subscription



- Enterprise has a commercial agreement with an MNO for certain connectivity services.
- Enterprise owns the UE subscriptions and is aware of their SUPIs and GPSIs (because it has ordered SIM cards or SIM profiles)
- UE IP address may belong to the enterprise network address space.
- Enterprise controls the UE subscription (associated services).

Consumer owns the UE subscription



- Enterprise has commercial agreement with MNO including QoD management, but not for a connectivity service.
- Enterprise does not own any UE subscriptions nor is aware of SUPIs or GPSIs (because it has not ordered any SIM cards/profiles).
  - But may learn GPSI via side-by channels (e.g., supplied by the user or the application).
- UE IP address is typically a public IP address (from the MNO public address space).
- Consumer controls the UE subscription (associated services).

Identification of UEs

- The proposed QoD API uses the UE IP address for identifying the UE of interest.
  - However, it is not clear how the enterprise is made aware of the UE IP address.
  - IP addresses are typically dynamically allocated to the UE, only at the time the UE establishes a PDU session and may be NATted. Connectivity may be lost, and a new UE IP address may be allocated when the connectivity is reestablished.
  - In 5G-LAN or IIoT environments, the devices may operate with Ethernet PDU sessions, i.e., not supporting IP.
  - The problem is that the QoD API is limited to sessions between the UE and the same Application Server that is controlling the QoS thus, the UE has already established a session to the AS, and the AS is aware of the IP address of the UE whose QoS should be managed.
    - It is not possible for the AS to request improved QoS for not-yet-established sessions, for which the UE does not have an IP address.
    - The AS cannot request improved QoS for the very first packet, because it needs to wait for the UE to establish a PDU session and gets its IP address.
- Among the alternative ways to identify UEs are to use identifiers already known by the enterprises, such as:
  - The **SUPI** (e.g., IMSI) contains sensitive information in the network, and is deemed to be used internally in the network. It should be avoided in external API usage due to security concerns.
    - SUPI is only known to the enterprise if the enterprise owns the subscription.
  - The **GPSI** (e.g., MSISDN). A permanent external identifier of the UE.
    - In enterprise-owned subscriptions, the enterprise knows the GPSI of the UE.
    - In consumer-owned subscriptions, the user or a UE Application should provide its GPSI to the enterprise.
  - These identifiers requires the network to determine the UE IP address (or MAC address in Ethernet PDU sessions), which is dependent on the Network Slice and/or DNN used by the PDU session.
  - Without knowing who invokes the API, SUPI and GPSI are not enough for determining the UE IP address alone, for example in case the UE has more than one IP address.

Ericsson proposes enabling the usage of the GPSI as a general identifier of the UE in API calls.

Summary of Identification of UEs

Criteria	UE IP address	MAC address	SUPI	GPSI
Permanent or ephemeral	Typically, ephemeral. IPv4: UE gets an IP address IPv6: UE gets a /64 prefix. UE freely builds its IPv6 address. IPv4 address or IPv6 prefix is allocated at PDU session establishment.	Permanent	Permanent	Permanent
PDU session applicability	PDU sessions of type IPv4, IPv6 or IPv4v6 within a Network Slice and DNN	Only for PDU sessions of type Ethernet within a Network Slice and DNN	All types of PDU session	All types of PDU session
Security and Privacy aspects			Internal network ID, should not be used externally.	In consumer-owned subscriptions, users should provide their GPSI
3GPP support	Supported	Supported	Supported	Supported
NAT traversal	NATs modify UE IPv4 address and port numbers	Not applicable	Not applicable	Not applicable

Questions for discussion

- Is there an agreement on the need for defining concepts such as Customer or Enterprise Identifier, Connectivity Service, Device Connection, QoS types, etc.? And how are these concepts related one to each other?
- How do the different Camara APIs make use of these concepts in a consistent way?
- Should Camara work on APIs for enabling enterprises obtaining, listing, and managing Customer or Enterprise Identifier, Connectivity Service, Device Connections, QoS types, etc.?
- Should these concepts be captured in a document that would include an overview of the terminology?

Ericsson Proposal

- Camara (perhaps the Commonalities WG) should provide concepts, terminology, and architecture that can be used by any other functional API defined by Camara.
- Two aspects must be considered in API calls:
  - The API invoker must be authorized to use the service API.
  - The recipient of the service (e.g., the UE) must be authorized to receive the service (e.g., QoD modification).
- Use GPSI (e.g., MSISDN) as a general identifier of the UE in API calls.