



Open Source Telecom Signaling

PAiCore Network API GW

Reigniting telecom innovation and sustainable profitability via open source signaling solutions

Network API Gw



Multiple Commercial Options

The free Basic Version, publicly available as open source, fosters continuous innovation. PAiCore also offers PT Bundles for professional services, and CT Editions available through OPEX plans with 24/7 technical support and usage rights, or CAPEX models that provide ownership of enhanced, carrier-grade CT products.



Multi-protocol Capabilities

The Network API Gw supports multi-layered signaling interworking across legacy Circuit-Switched Core Networks using SS7/SIGTRAN MAP and Packet-Switched Core Networks through Diameter and REST/HTTP interfaces, providing a unified gateway for modern telecom applications.



Flexible Deployment

The Network API Gw supports dockerized or standalone JVM deployments, providing a scalable and reliable foundation for high-performance signaling in cloud or on-premises environments.



Integration with Internal and External Ecosystems

The Network API Gw acts as a bridge for tailor-made applications with a complete set of API interfaces.



High-processing Capabilities

The Network API Gw delivers a validated throughput of 9,000 transactions per second (TPS) through its decoupled, event-driven architecture, providing high-performance signaling translation with stable and repeatable performance under sustained load.



Compatible with Third-party Monitoring Systems

The Network API Gw is compatible with third-party monitoring systems to check the different running parameters and KPIs, thus enabling multi-protocol visibility of network health and traffic trends.



Silent Network Authentication (SNA) Support

The PAiCore Network API Gateway delivers passwordless, SIM-based Silent Authentication. By connecting digital services to operator networks via SS7/MAP or Diameter STa, it enables CAMARA Number Verification while completely hiding telecom signaling complexity from the application backend.

Network API Gw – CAMARA APIs



Location Retrieval

It allows to retrieve the area where a specific user device is located. Powered by the integration with PAiCore GMLC.



Location Verification

The “Location Verification” API determines whether a mobile device is within the proximity of a specified geographical area. Powered by the integration with PAiCore GMLC.



Geofencing Subscriptions

Using the “Geofencing Subscriptions” API, users can create subscriptions for pairs “mobile line – geographic area”, enabling them to receive notifications whenever a mobile line enters or exits the predefined area. Powered by the integration with PAiCore GMLC.



Number Verification

The “Number Verification” API is a service that enables businesses to confirm the ownership of a mobile phone number and retrieve the number directly from the user’s device.



Know Your Customer Age

Takes the network subscription identifier (e.g. the mobile phone number for a mobile network subscriber) and checks if the age of the subscriber is older than the age threshold in the request. (Q3-2026).



Silent Authentication

Verify that a mobile device is using the SIM linked to a phone number – without passwords, SMS codes, or user interaction.



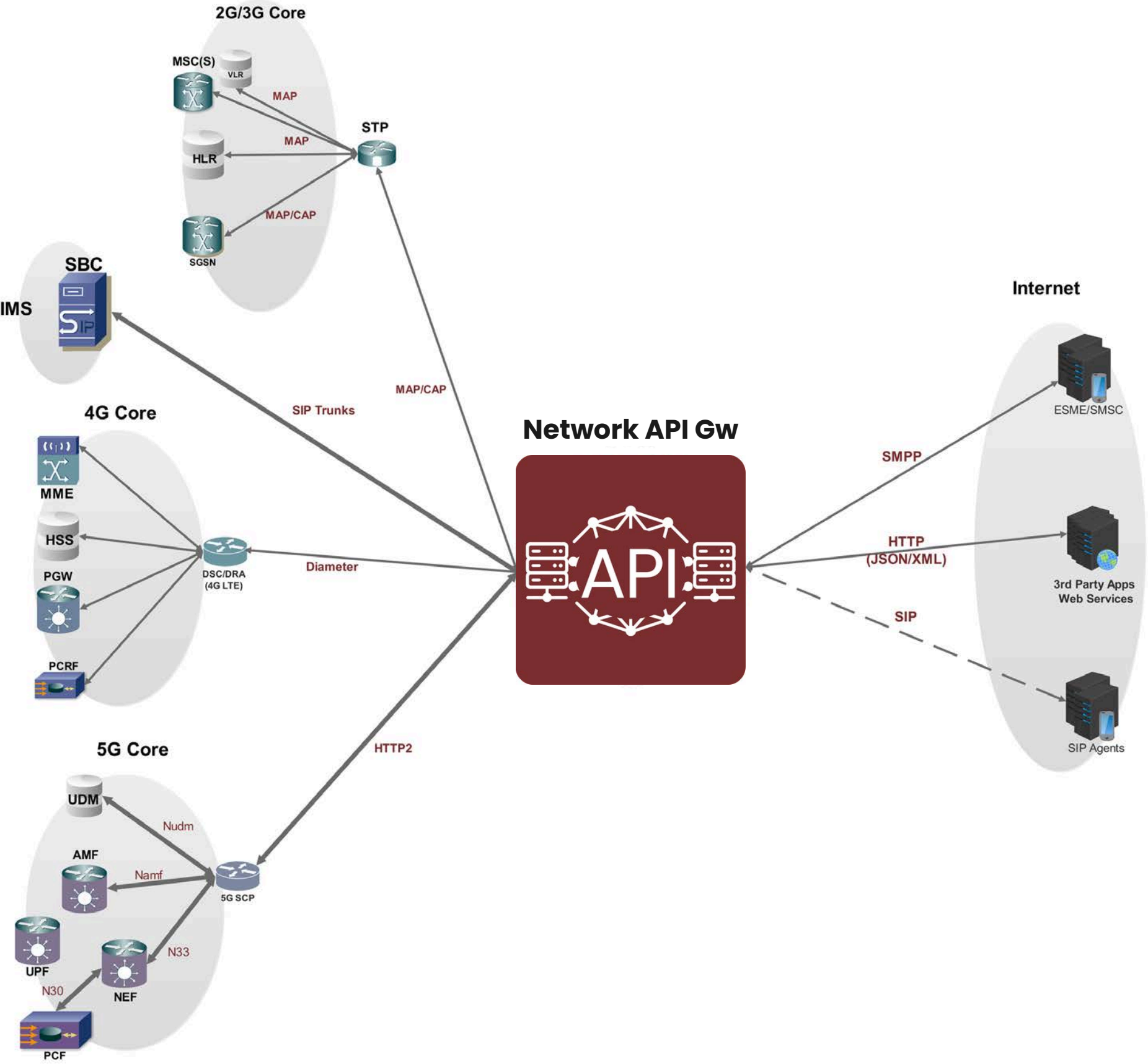
Open Source Telecom Signaling

Appendix – Technical Information



Architecture

Interconnection Points





Network API Gw

Simplifying Multi-protocol Interworking

PAiCore's **open-source** Network API Gateway bridges modern HTTP/API applications with reliable telecom protocols like SS7 and Diameter. It provides a robust foundation for STP, DRA, and multi-protocol proxying, significantly **simplifying complex interworking functions**.

Furthermore, its event-driven architecture enables advanced use cases such as **Silent Network Authentication** (SNA), delivering asynchronous, **CAMARA-aligned** identity verification through standard APIs.

Key Features



Multi-protocol interworking (REST/HTTP, SS7/MAP, and Diameter)



Containerized for ease of deployment & management



High availability with load balancing & fault tolerance



REST API support for external service support over telecom networks

Recent Success Stories



BTS deployed **PAiCore's Network API Gw** for STP/DRA interworking, enabling secure authentication vector retrieval to power their Silent Authentication services.



019Mobile relies on **PAiCore's Network API Gw** for multi-protocol interworking, ensuring efficient signaling & service delivery of MVNO services over their network.

Silent Authentication & Number Verification

The PAiCore Network API Gateway connects digital applications to mobile network authentication systems via standard APIs, hiding telecom signaling complexity from the service provider.

A simpler and more secure user journey

- No codes to type:

The verification happens silently in the background.

- Stronger protection:

No one-time code is exposed to interception or phishing.

- Faster onboarding:

Users can be verified with less friction and fewer steps.

- Built for digital services:

Supports secure identity checks across apps and online services.

Business value

- CAMARA-aligned Number Verification:

Supports standardized number-verification experiences for digital services.

- Flexible operator integration

Works with existing SS7/MAP or Diameter environments according to operator connectivity.

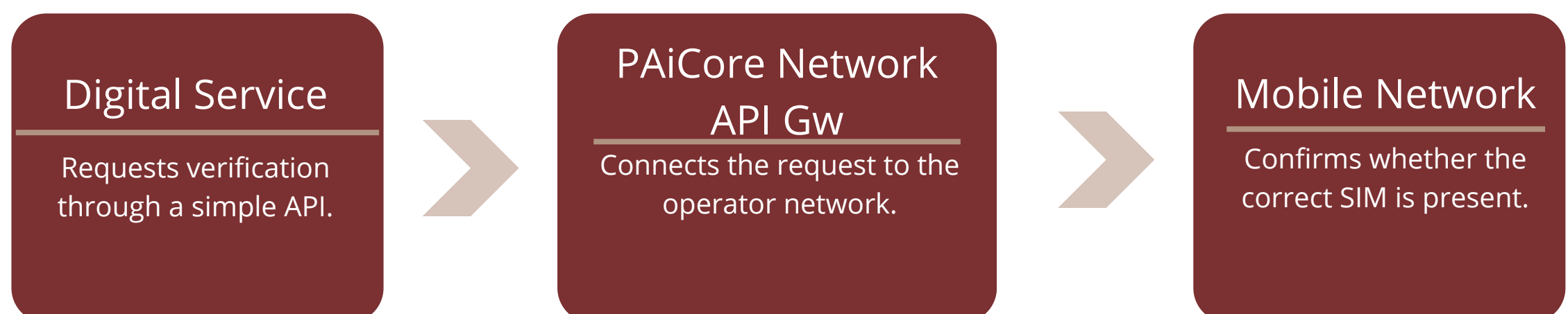
- Operator-controlled security

Authentication remains anchored in the mobile network and the subscriber SIM.

- Less friction, more trust

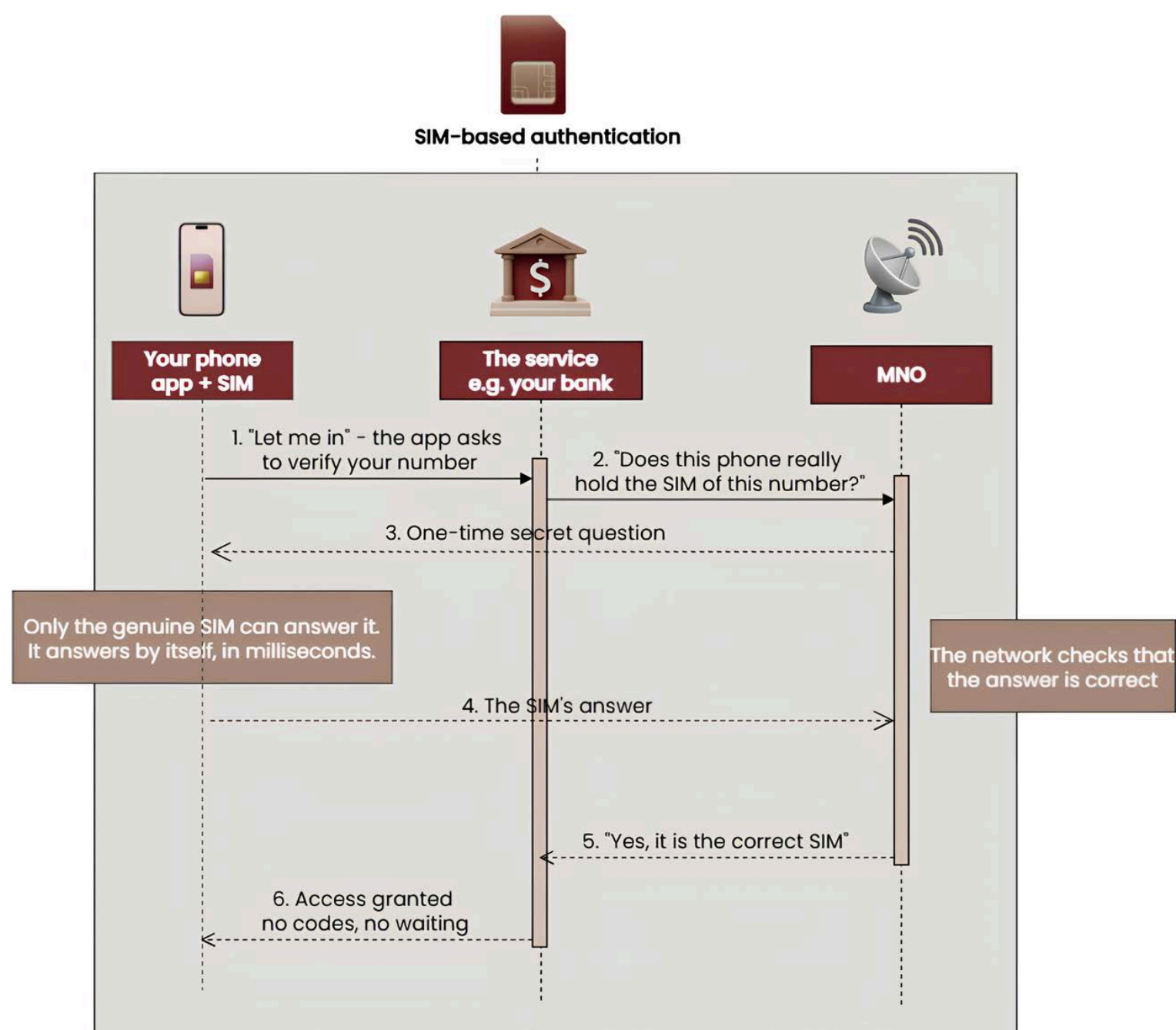
Helps enable seamless banking access, onboarding, identity verification, and anti-fraud journeys.

One gateway for secure network verification



Silent Authentication & Number Verification

PAiCore Network API Gw enables CAMARA-style Number Verification by exposing an HTTP/JSON interface over the operator signaling plane. It connects the Entitlement Server to the MNO through SS7/MAP or Diameter STa, while the device-side authentication remains governed by GSMA TS.43.

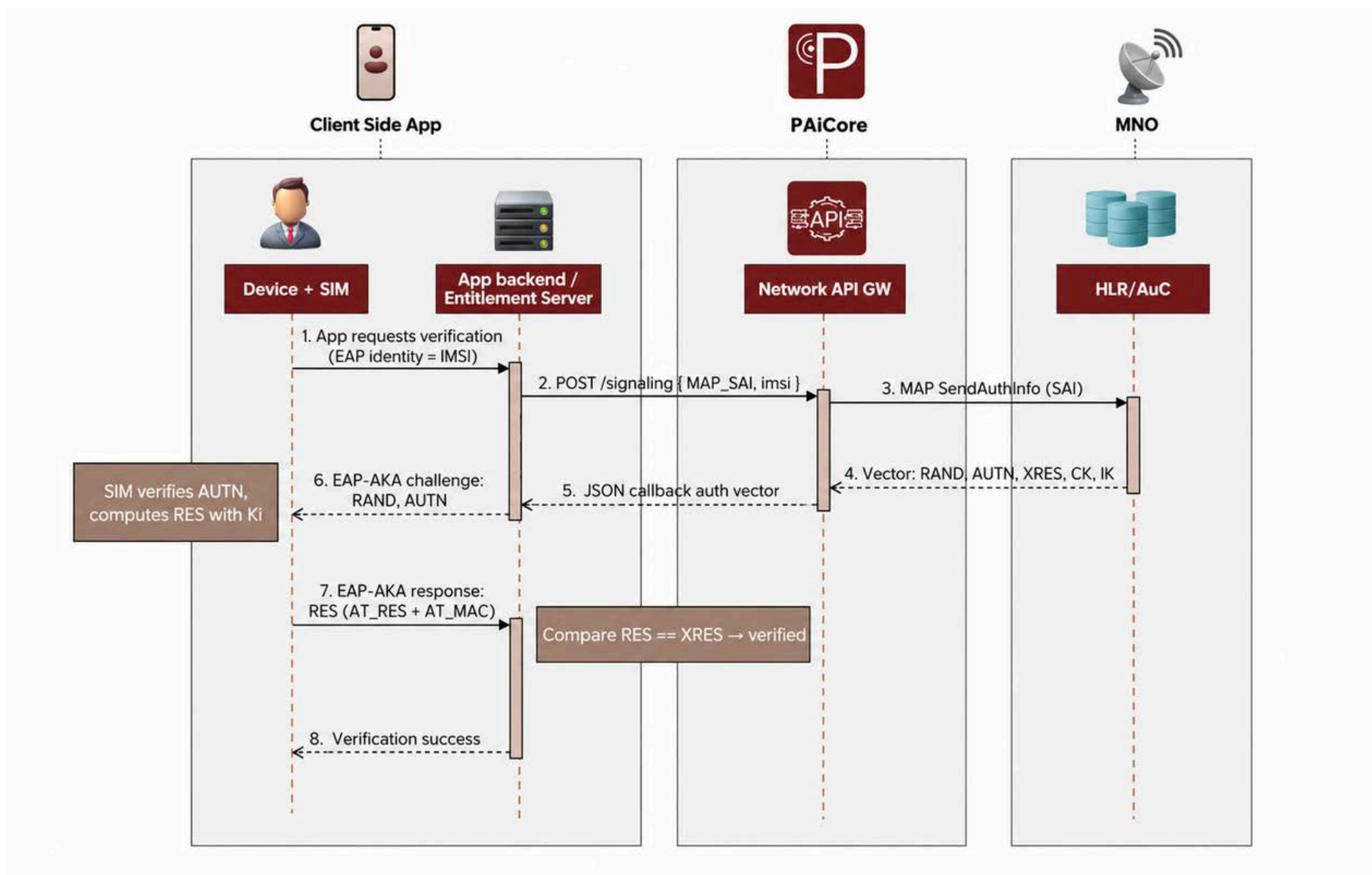


Protocol and component mapping

- **Device / Entitlement Client**
 - Runs the USIM AUTHENTICATE command and exchanges EAP-AKA messages. The standard path is the OS entitlement client defined by GSMA TS.43.
- **Entitlement Configuration Server (ECS)**
 - Orchestrates the device flow over HTTPS/TS.43 and invokes PAiCoreNetwork API Gw through HTTP/JSON.
- **PAiCore Network API Gw**
 - Translates API requests into MAP SendAuthenticationInfo or Diameter-EAP transactions and correlates the asynchronous response through a transaction ID and callback.
- **MNO Core**
 - Provides the authoritative authentication data from the HLR/AuC or 3GPP AAA/HSS.
- **Technical value**
 - Application systems do not need to implement or operate SS7, SIGTRAN, Diameter or EAP transport stacks.
 - HTTP requests are decoupled from signaling response times through transaction correlation and callbacks.
 - Operators can select SS7/MAP or Diameter STa according to available connectivity and security requirements.

Silent Authentication over SS7 / MAP

In SS7/MAP mode, Network API Gw acts as a VLR-facing signaling endpoint and requests an authentication vector from the HLR/AuC using MAP SendAuthenticationInfo (SAI). The Entitlement Server or application backend performs the final RES = XRES validation.



Signaling sequence

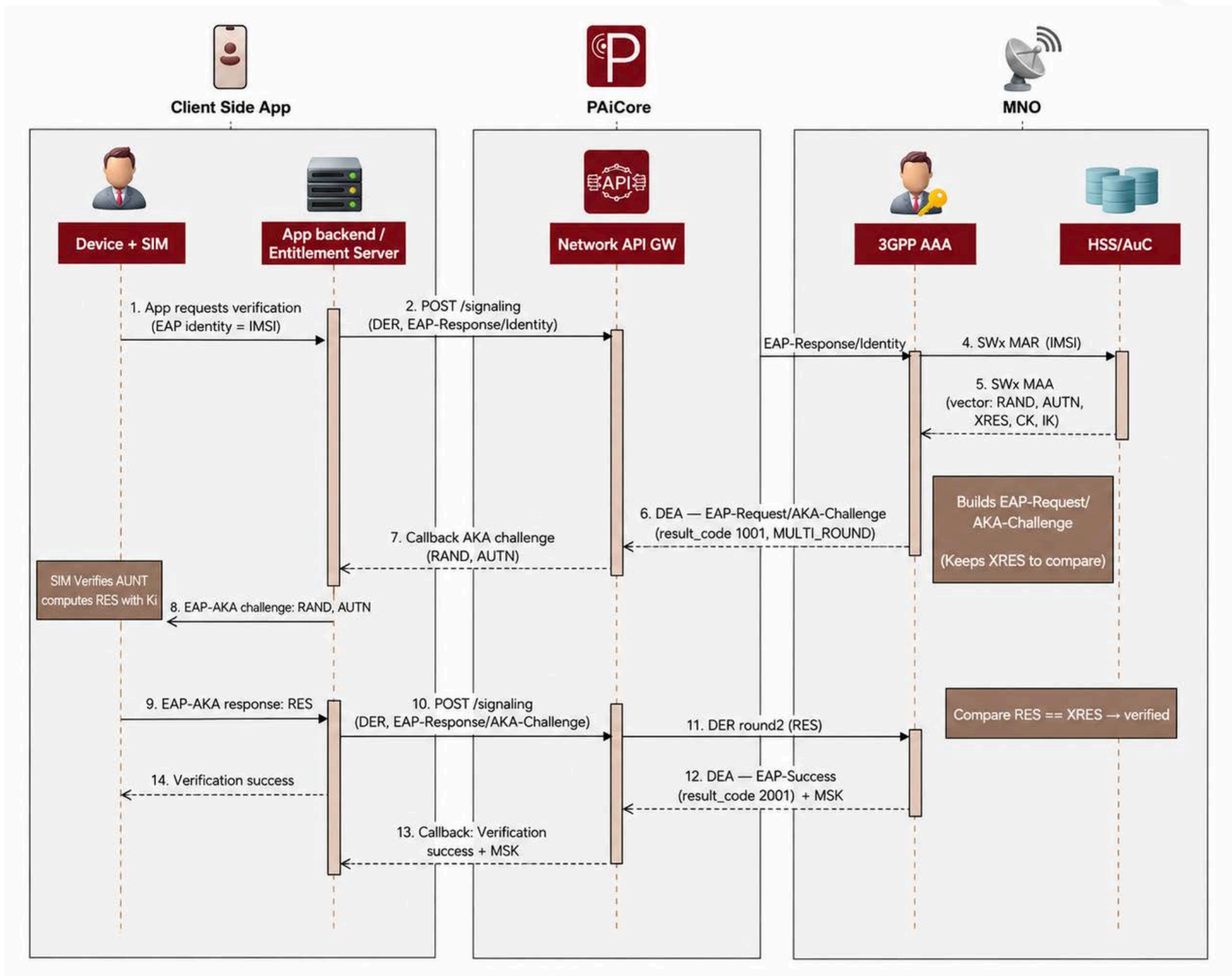
- The backend sends an HTTP request containing the IMSI-derived EAP identity and callback URL.
- Network API Gw maps the request to MAP SAI and sends it to the HLR/AuC.
- The HLR/AuC returns a 3G quintet {RAND, AUTN, XRES, CK, IK} or a 2G triplet {RAND, SRES, Kc}.
- Network API Gw returns the vector through the JSON callback.
- The backend challenges the SIM with RAND/AUTN and validates RES against XRES.

Security boundary

Ki never leaves the SIM or AuC. In this mode, the authentication vector reaches the trusted Entitlement Server/backend, which is responsible for the cryptographic comparison and protection of the returned key material

Silent Authentication over Diameter STa

In Diameter STa mode, Network API Gw relays the EAP-AKA exchange to the operator's 3GPP AAA. The AAA retrieves the authentication vector from the HSS/AuC over SWx and performs the $RES = XRES$ comparison inside the operator domain.



Signaling sequence

- The backend sends the EAP identity in an HTTP request to Network API Gw.
- The gateway sends a Diameter-EAP-Request (DER, command 268; STa application-id 16777250) to the 3GPP AAA.
- The AAA requests the vector from the HSS/AuC over SWx using MAR/MAA.
- The AAA returns an EAP-Request/AKA-Challenge with RAND and AUTN.
- The SIM produces RES; the gateway relays the second EAP round to the AAA.
- The AAA validates RES against XRES and returns EAP-Success and the MSK.

Security boundary

XRES, CK and IK remain inside the operator domain. Network API Gw transports EAP payloads but does not expose the authentication vector to the application backend, providing the strongest security posture of the two supported modes.

Silent Authentication Use Case

Architecture Event-Driven Network API Gw

Silent Network Authentication (SNA) Solution

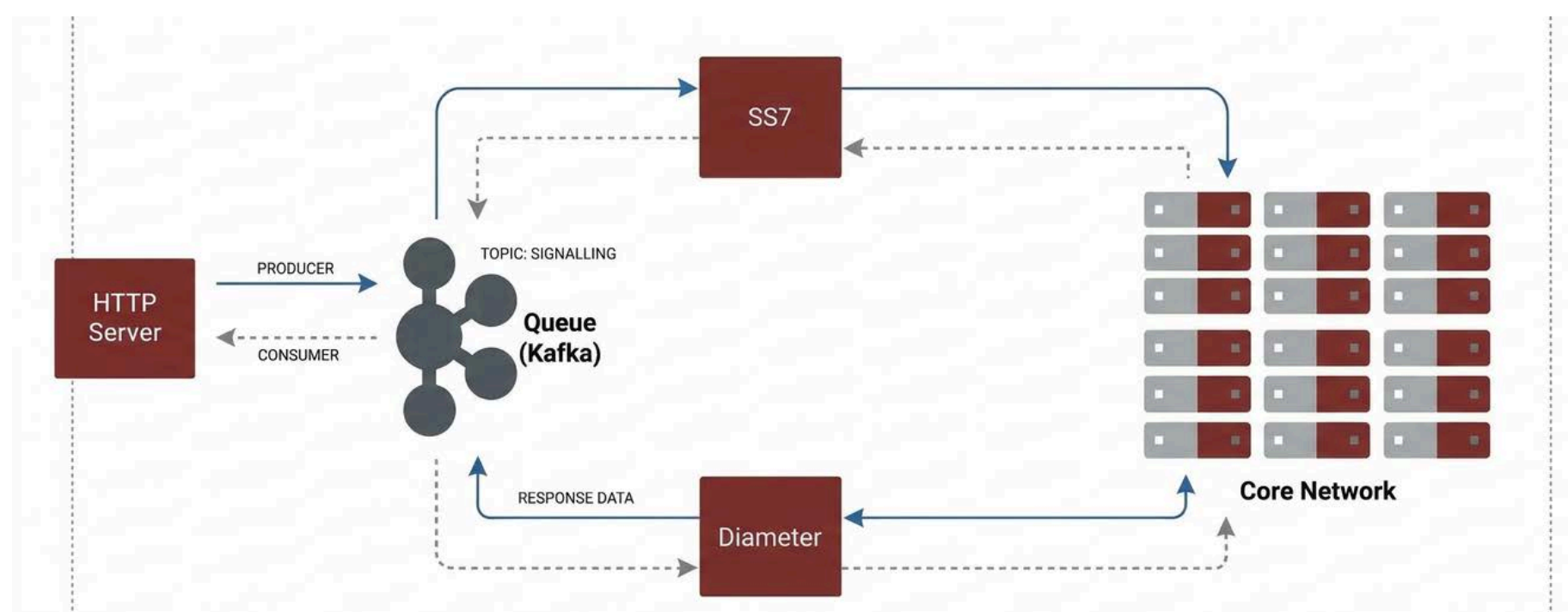
This documentation details the implementation of a Silent Network Authentication (SNA) solution designed to enable instant user verification by leveraging native telecommunications network data.

At the core of this system is a high-performance Network API Gw that serves as an intelligent translation bridge between modern HTTP/REST-based requests and essential mobile signaling protocols, such as SS7 and Diameter.

1. System Overview

The Network API Gw is an application designed to act as a translation bridge between HTTP/REST, and telecommunications signaling networks based on SS7 and Diameter protocols.

To manage the disparity in response times between IP networks and mobile networks, the Network API Gw implements an **Event-Driven Architecture** using Apache Kafka as the central message bus. HTTP requests are processed in a non-blocking manner, assigned a unique Transaction ID, and the mobile network's responses are delivered to the client asynchronously via a Webhook / Callback.



2. Core Components

The system consists of a central event bus and three main modules that interact exclusively through it, ensuring complete decoupling.

2.1. HTTP Module (API Gateway & Webhook Dispatcher)

- **Inbound (Reception):**
 - Exposes REST endpoints to receive signaling requests.
 - Validates the payload structure (which must include the `callback_url`).
 - Generates a unique Transaction ID for operation tracking.
 - Publishes the event to the corresponding Kafka topic (**e.g., `sgw.requests.ss7` or `sgw.requests.diameter`**).
 - Responds immediately and synchronously to the client with an HTTP status code (e.g., 202 Accepted) and the generated Transaction ID.
- **Outbound (Dispatcher):**
 - Acts as a Kafka consumer on the response topic (**`sgw.responses.callback`**).
 - Upon receiving a response event from SS7 or Diameter, it extracts the `callback_url` and the Transaction ID.
 - Executes an HTTP POST request to the original client system, delivering the final response object and its associated Transaction ID.

2.2. SS7 Module

Responsible for native interaction with the common channel signaling network.

- **Transmitter Mode:** Consumes events from the ***sgw.requests.ss7*** topic. It translates the JSON payload into SS7 primitives, initiates the TCAP dialogue, and sends the message to the mobile network.
- **Receiver Mode:** Features asynchronous listeners connected to the SS7 stack. When it receives a network response (or an incoming network-originated message), it correlates the transaction, packages the result into a standard object, and publishes it to the ***sgw.responses.callback*** Kafka topic.

2.3. Diameter Module

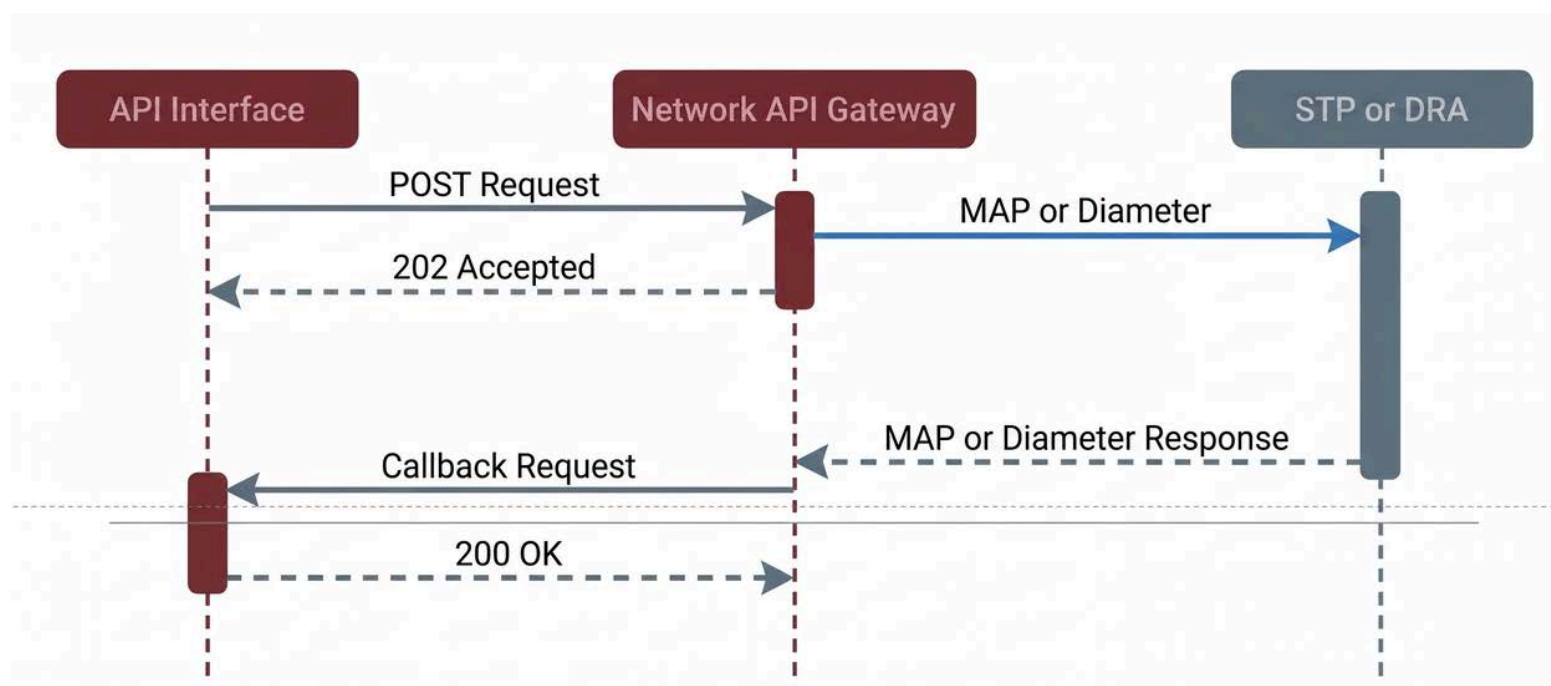
Responsible for native interaction with LTE/IMS nodes and OCS platforms.

- **Transmitter Mode:** Consumes events from the ***sgw.requests.diameter*** topic. It builds the required commands and Attribute-Value Pairs (AVPs), and initiates the Diameter session toward the destination node.
- **Receiver Mode:** Through its event listeners, it captures asynchronous network responses. It packages the response along with the original Transaction ID and publishes the event to the ***sgw.responses.callback*** topic.

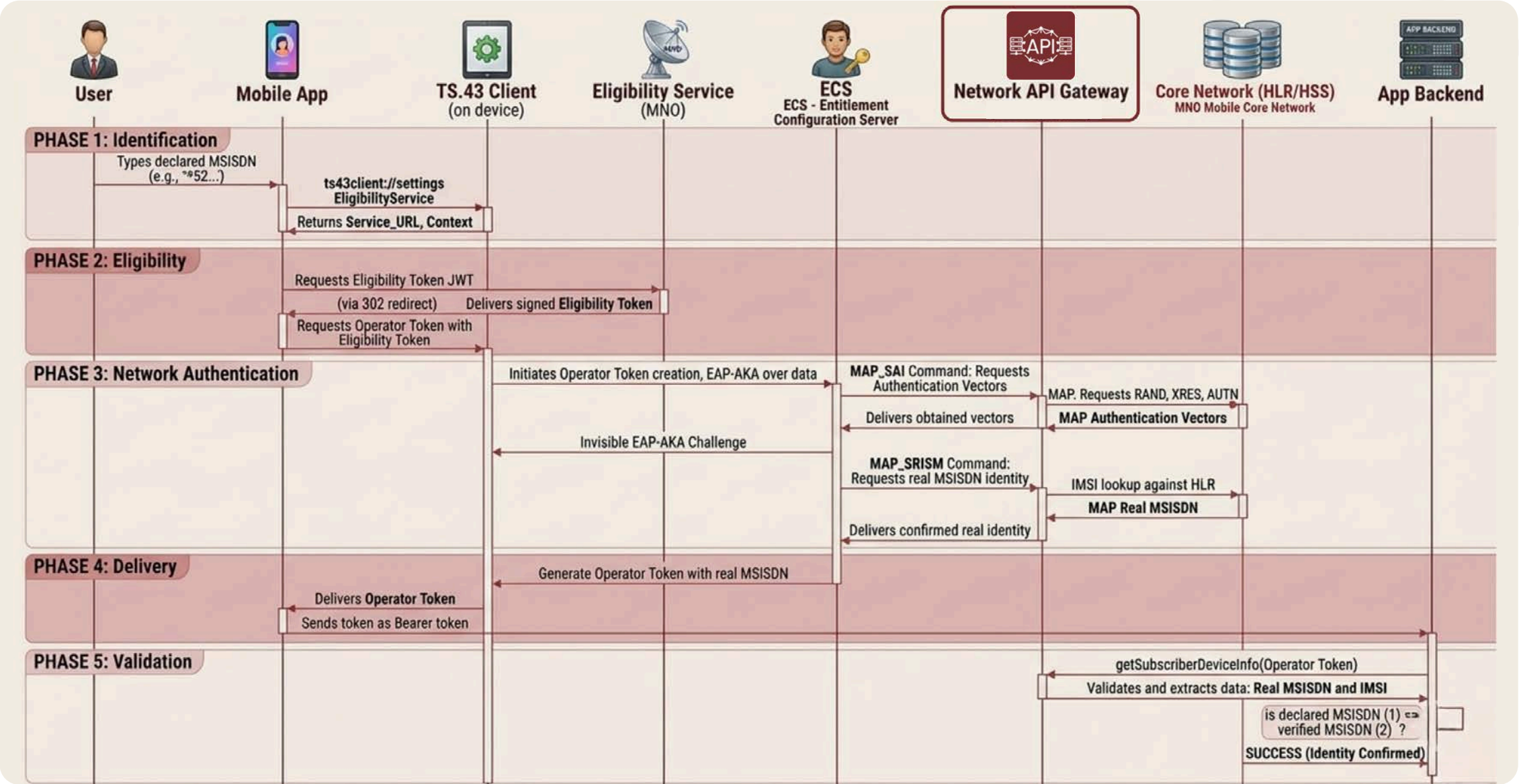
3. End-to-End Transactional Flow

The lifecycle of a message follows a strict deferred request-response pattern:

1. **Client Request:** An external system sends an HTTP POST to the Network API Gw indicating the desired operation and its `callback_url`.
2. **Admission:** The HTTP Module receives the request, generates a Transaction ID (e.g., Tx-8899), and queues the message in Kafka.
3. **Immediate Response:** The HTTP Module returns an HTTP code to the client indicating successful reception, attaching Tx-8899. The HTTP connection is closed.
4. **Consumption & Signaling:** The SS7 or Diameter Module reads the message from Kafka and triggers the query to the telecommunications network.
5. **Network Response:** The mobile network processes the request and returns the answer to the corresponding module (SS7 or Diameter).
6. **Response Queuing:** The signaling module takes the technical response, formats it into a standard JSON object, attaches Tx-8899 and the `callback_url`, and sends it to the Kafka response topic.
7. **Final Delivery (Callback):** The HTTP Module reads the response from Kafka and performs an HTTP POST to the client's `callback_url`, delivering the response object alongside the Transaction ID.



Silent Authentication Flow



Connectivity Requirements

TPS & Recommended HW

TPS Range	RAM	Core	Disk	Network	Sample AWS Instance
Up to 9,000 TPS	16 GB	8	250 GB SSD	1Gbps	m5.2xlarge

Note: Our software products are fully tested and optimized for deployment in AWS EC2 instances, ensuring peak performance and reliability. While they are compatible with any Linux-based virtual environment, performance may vary depending on the underlying infrastructure and configuration.

For the following interconnection tables, the displayed values must be taken as examples.

For SS7 interconnection

Description	Value	Comments
OPC	1	Originating Point Code, assigned to the platform (SMSC, GMLC, USSD Gw, etc.) as part of its MTP3 routing label.
DPC	2	Destination Point Code, usually belonging to a core network Signaling Transfer Point (STP) at MTP3 layer.
Global Title	919800123456	Signalling Connection Control Part (SCCP) Global Title digits assigned to the platform.
SSN	6	Signalling Connection Control Part (SCCP) Subsystem Number assigned to the platform.
Network Indicator	National	International (0/1) or National network (2/3) determination for Signaling Point Codes (OPC/DPC).

For SIGTRAN (SCTP/M3UA) interconnection

Description	Value	Comments
IP and Port at peer side	1.1.1.1	Peer (usually SGW) SCTP endpoint IP address(es) and port number list.
IP and Port at host	2.2.2.2	Host (usually AS) SCTP endpoint IP address(es) and port number list.
SCTP mode	peer	Side that shall initiate the SCTP association and act as client: Host or Peer?
M3UA type	AS	Process that shall run on the platform host at M3UA: AS or SGW or IPSP?
Routing Context	1	The Routing Context assigned to the platform identifying its MTP3 routing key.
Network Appearance	102	Identifier of the SS7 network context for the routing key. Not required if SGW operates in the context of a single SS7 network, or if individual SCTP associations are dedicated to each SS7 network context.
Traffic mode	Override	Traffic mode type: Override or Loadshare. Override: ASP takes over all traffic overriding any currently active ASPs in the AS. Loadshare: the ASP will share in the traffic distribution with any other currently active ASPs.
Message Exchange	SE	IPSP model defining if each exchange of ASPTM or ASPSM messages can be initiated by either IPSP (Double Exchange, DE) or not (Single Exchange). SE if unknown.
MUA ASP ID	1	Optional ASP ID (provide it only if it has been set by the operator).

For HTTP/1.1 and HTTP/2 interconnection

No	Comments
1	The required API documentation is available.



For SIP interconnection

No	Description	Value	Comments
1	SIP Proxy IP	1.1.1.1	IP address of the SIP proxy.
2	SIP Proxy Port	5060	Port number of the SIP proxy.
3	Username	sip_user	Username for SIP authentication.
4	Password	*****	Password for SIP authentication.
5	Domain	sip.customer.com	SIP domain for user registration and authentication.

For Diameter interconnection

No	Description	Value	Comments
1	Host IP	1.1.1.1	Diameter IP address(es) assigned to the PAiCore Technology platform (SMSC, GMLC, DRA, etc.).
2	Host Port	3868	Diameter port number(s) assigned to the PAiCore Technology platform (SMSC, GMLC, DRA, etc.).
3	Host Realm	epc.mcc748.mnc012.3gppnetwork.org	The string(s) in the Network Access Identifier (NAI) that immediately follows the "@" character assigned to the PAiCore Technology platform (SMSC, GMLC, DRA, etc.).
4	Host User Name	host_user_name	The string(s) which contains the User-Name in a format consistent with the NAI specification assigned to the PAiCore Technology platform (SMSC, GMLC, DRA, etc.).
5	Host Number	861684684	ISDN (E.164) number assigned to the PAiCore Technology platform (SMSC, GMLC, DRA, etc.).
6	Peer IP(s)	2.2.2.2	Diameter IP address(es) of the peer side (DSC or DRA or DEA or IWF).
7	Peer Port(s)	3869	Diameter port number(s) of the peer side (DSC or DRA or DEA or IWF).
8	Peer Port Range	3868-3869	Port range to use for establishing SCTP associations.
9	Peer Realm	epc.mcc748.mnc01..3gppnetwork.org	The string in the DSC/DRA/DEA/IWF Network Access Identifier (NAI) that immediately follows the "@" character.
10	Peer User Name	peer_user_name	The string which contains the DSC/DRA/DEA/IWF User-Name in a format consistent with the NAI specification.

For SMPP interconnection

No	Description	Value	Comments
1	System ID	smpp_account_1	Identifies the External Short Message Entity (ESME) requesting to bind with the SMSC.
2	Password	*****	The password used by the SMSC to authenticate the ESME requesting to bind.
3	Peer IP	2.2.2.2	IP address used by the ESME to send or accept bind procedures with the SMSC.
4	Peer Port	2775	Port number used by the ESME to send or accept bind procedures to the SMSC.
5	Host IP	1.1.1.1	IP address used on bind procedures when the SMSC is acting as ESME or Service Centre (SC).
6	Host Port	2776	Port number used on bind procedures when the SMSC is acting as ESME or Service Centre (SC).
7	SMPP Bind Type	Transceiver	Binding type value: Transmitter, Receiver or Transceiver.
8	SMPP Session Type	Client	SERVER (SMSC will act as a server, listening for incoming SMPP binds) or CLIENT (SMSC will act as a client, initiating SMPP binds to remote servers).
9	System Type	smpp_account	Used to categorize the type of ESME that is binding to the SMSC.