

OmniSEP - Service Endpoint Platform

OmniSEP is a unified service endpoint platform providing carrier-grade implementations of mobile device provisioning protocols. It handles TS.43 Entitlement Configuration, XCAP Supplementary Services, and Visual Voicemail from a single platform.

Quick Links

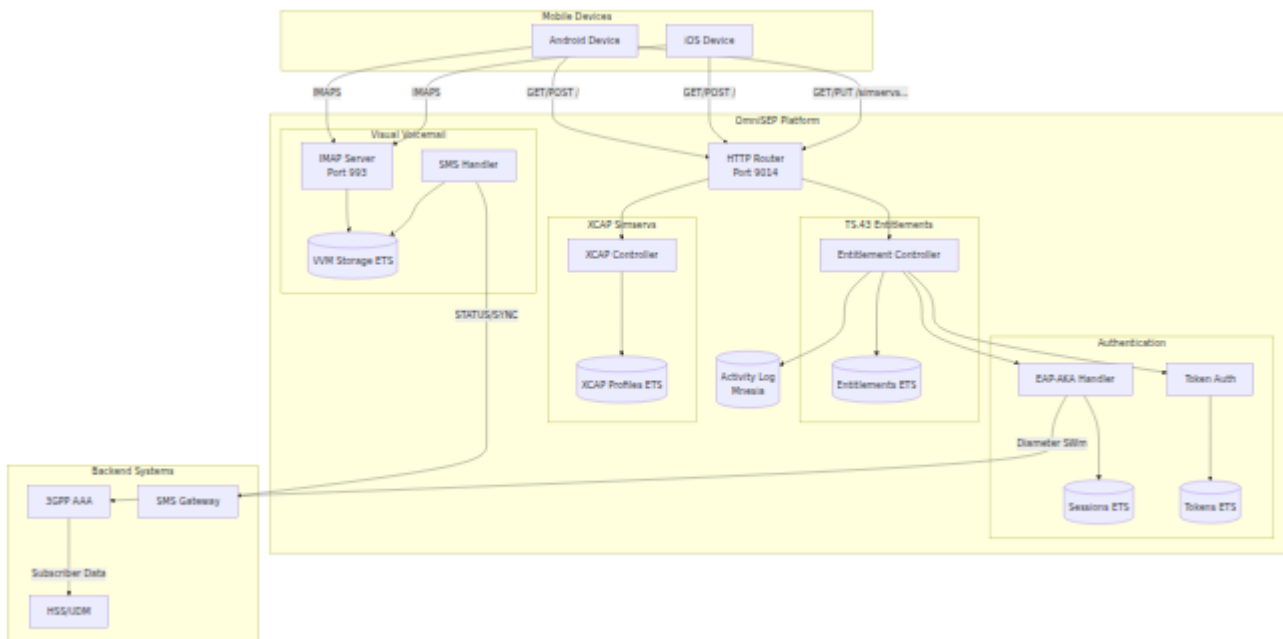
Operations & Monitoring

- **Configuration Reference** - Complete parameter documentation for all services
- **Management API** - REST API reference for monitoring and administration (port 8443)
- **Troubleshooting** - Common issues and resolutions

Service Documentation

- **TS.43 Entitlements** - GSMA TS.43 Service Entitlement Configuration
- **XCAP Sirmservs** - ETSI TS 183 023 Supplementary Services (Call Forwarding, Call Barring, Caller ID)
- **Visual Voicemail** - OMTP VVM v1.3 / GSMA TS.46 Visual Voicemail Server

Architecture Overview



Supported Services

TS.43 Entitlement Configuration

Implements GSMA TS.43 for device service entitlement queries:

App ID	Service	Description
ap2003	Voice-over-Cellular	VoLTE/VoNR entitlement status
ap2004	VoWiFi	Voice over WiFi calling entitlement
ap2005	SMSoIP	SMS over IP entitlement
ap2006	ODSA Companion	On-Device Service Activation (companion)
ap2009	ODSA Primary	On-Device Service Activation (primary)
ap2010	Data Plan Boost	Data plan information
ap2012	Direct Carrier Billing	DCB service status
ap2016	Satellite Mode	Satellite connectivity entitlement

XCAP Supplementary Services

Implements ETSI TS 183 023 for IMS supplementary service configuration:

Service	Description	Reference
Communication Diversion	Call forwarding (CFU, CFB, CFNA, CFNRC)	ETSI TS 183 004
Incoming Call Barring	Block incoming calls by type	ETSI TS 183 023
Outgoing Call Barring	Block outgoing calls by type	ETSI TS 183 023
OIP/OIR	Caller ID presentation and restriction	ETSI TS 183 023

Visual Voicemail

Implements OMTP VVM Specification v1.3 and GSMA TS.46:

Component	Description	Reference
IMAP Server	Message retrieval and management	RFC 3501
SMS Protocol	SYNC/STATUS provisioning messages	OMTP VVM v1.3
Greeting Management	Personal and extended absence greetings	GSMA TS.46
Transcription	Voicemail-to-text support	OMTP VVM v1.3

Endpoints Overview



Endpoint	Method
/	GET/POST
/simservs.ngn.etsi.org/users/{sip}/simservs.xml	GET/PUT
/simservs.ngn.etsi.org/users/{sip}/simservs.xml/~/ {xpath}	GET/PUT
/api/activity	GET
/api/entitlements/{imsi}	GET/POST
/api/xcap/{msisdn}	GET/POST
/health	GET

Web UI

OmniSEP includes a real-time web interface for monitoring and management, built with Phoenix LiveView. Access the UI at `http://<host>:9014/`.

XCAP Simservs Profile management showing subscriber supplementary services configuration

Dashboards

Dashboard	Path	Description
Status	/	System health, service status, memory usage, storage statistics
Entitlements	/entitlements	View and manage TS.43 custom entitlements by IMSI
Sessions	/sessions	Monitor EAP-AKA sessions and active authentication tokens
Activity	/activity	Browse TS.43 activity logs with filtering and pagination
XCAP Profiles	/xcap	Manage XCAP Sirmservs profiles, sync with HSS
Diameter	/diameter	Monitor Diameter peer connections and status
Logs	/logs	Real-time system log viewer with level filtering

Status Dashboard

The main status dashboard provides at-a-glance system health:

- **Service Status:** HTTP server, EAP-AKA, VVM service indicators
- **Storage Counters:** Entitlements, XCAP profiles, sessions, tokens, activity records
- **Memory Usage:** Total, process, and ETS memory breakdown
- **Uptime:** System uptime display

All metrics auto-refresh every 5 seconds.

XCAP Profile Management

The XCAP dashboard allows operators to:

- Search and view subscriber profiles by MSISDN
- Edit supplementary service settings (OIP/OIR, call forwarding, call barring)
- Pull profiles from HSS via Diameter Sh interface
- Push local changes back to HSS
- View modification metadata (timestamp, client IP, User-Agent)

Activity Monitoring

The activity dashboard provides real-time visibility into all device requests:

Activity log showing XCAP and entitlement requests with request/response details

Features:

- Filter by request type (XCAP, Entitlement Query, EAP Challenge, etc.)
- Search by IMSI, MSISDN, Terminal ID, or Client IP
- View detailed request/response data including headers, body, and path
- Track HTTP method and response status codes

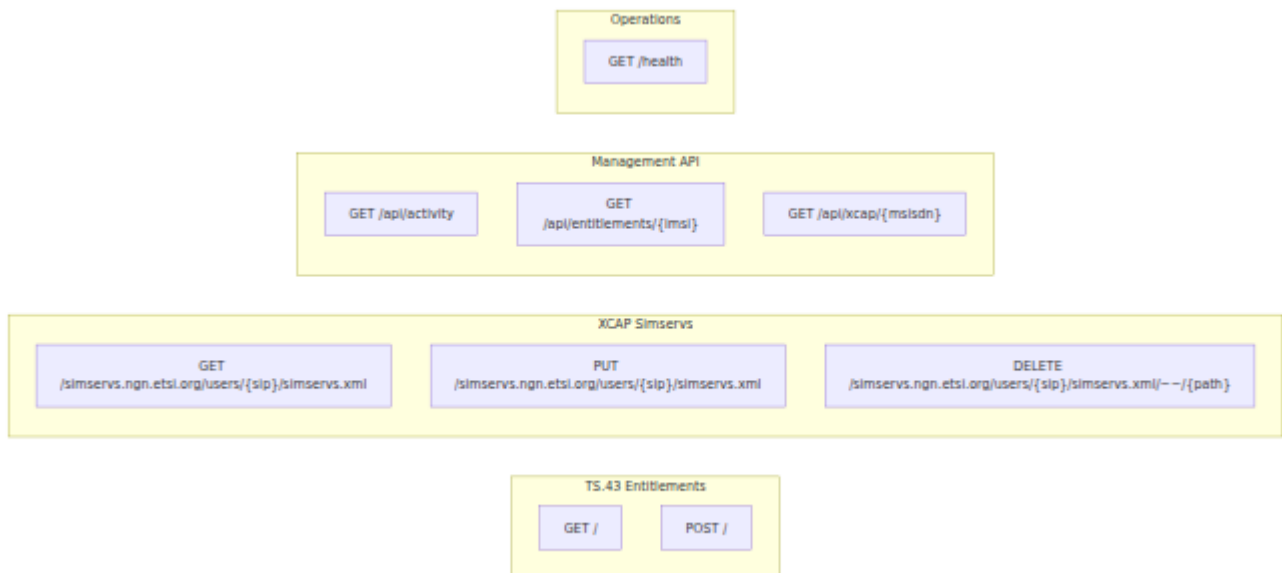
Session Monitoring

The sessions dashboard displays:

- **Sessions Tab:** Active EAP-AKA sessions with state (pending, challenge_sent, authenticated, failed)
- **Tokens Tab:** Active authentication tokens with time remaining until expiry

Both views support manual session/token revocation.

Authentication Flow



Quick Start Configuration

Minimal Configuration

```
# config/config.exs
import Config

config :omni_sep,
  http_port: 9014,
  http_ip: {0, 0, 0, 0},
  server_fqdn: "sep.mnc001.mcc001.pub.3gppnetwork.org",
  entitlement_version: "2.0"

# Default entitlements for all subscribers
config :omni_sep, :default_entitlements,
  vowifi: %{
    entitlement_status: 1,
    addr_status: 2,
    tc_status: 2,
    prov_status: 1
  },
  volte: %{
    entries: [
      %{access_type: 1, home_roaming_nw_type: 1,
        entitlement_status: 1}
    ]
  }

# Token settings
config :omni_sep, :token,
  validity_seconds: 86400,
  signing_secret: "your-production-secret-here"

# EAP-AKA settings
config :omni_sep, :eap_aka,
  enabled: true,
  session_timeout_ms: 30_000
```

See [Configuration Reference](#) for complete parameter documentation.

Documentation Structure

By Role

Network Operators:

1. Start with this overview
2. Review [Configuration Reference](#) for deployment settings
3. Review [Troubleshooting](#) for common issues

Service Configuration:

1. [TS.43 Entitlements](#) for VoWiFi/VoLTE provisioning
2. [XCAP Sirmservs](#) for call forwarding/barring

Troubleshooting:

1. [Troubleshooting Guide](#) for common issues
2. Check [Activity Logging](#) for request tracking

By Protocol

GSMA TS.43:

- [TS.43 Entitlements](#) - Complete service entitlement documentation
- Specification: [GSMA TS.43](#)

ETSI XCAP:

- [XCAP Sirmservs](#) - Supplementary services documentation
- Specifications:
 - [ETSI TS 183 023](#) - XCAP framework
 - [ETSI TS 183 004](#) - Communication Diversion
 - [RFC 4825](#) - XCAP protocol

OmniSEP Configuration Reference

Complete configuration reference for OmniSEP Service Endpoint Platform.

Table of Contents

- [HTTP Server Configuration](#)
- [Storage Configuration](#)
- [Default Entitlements](#)
- [Token Configuration](#)
- [EAP-AKA Configuration](#)
- [Diameter Configuration](#)
- [Activity Logging](#)
- [Environment-Specific Configuration](#)

Configuration Structure



HTTP Server Configuration

Controls the HTTP endpoint where devices connect.

```
config :omni_sep,
  http_port: 9014,
  http_ip: {0, 0, 0, 0},
  server_fqdn: "sep.mnc001.mcc001.pub.3gppnetwork.org",
  entitlement_version: "2.0"
```

Parameters

Parameter	Type	Required	Default	Description
<code>http_port</code>	Integer	No	9014	TCP port for HTTP and TLS terminator in the server.
<code>http_ip</code>	Tuple	No	<code>{0, 0, 0, 0}</code>	IP address to bind the server on. Use special addresses for all interfaces. Use special addresses for all hosts.
<code>server_fqdn</code>	String	Yes	-	Server's fully qualified domain name (FQDN) in the format <code>aes.mnc<MNC>.mcc<MCC>.pub.3gppnetwork.org</code> per GSMA TS.43 S .
<code>entitlement_version</code>	String	No	"2.0"	TS.43 protocol version. Default is "2.0".

FQDN Format

The server FQDN follows the 3GPP naming convention:

```
aes.mnc<MNC>.mcc<MCC>.pub.3gppnetwork.org
```

Component	Description	Example
<code>aes</code>	Application Entitlement Server prefix	<code>aes</code>
<code>mnc<MNC></code>	Mobile Network Code (3 digits, zero-padded)	<code>mnc001</code>
<code>mcc<MCC></code>	Mobile Country Code (3 digits)	<code>mcc310</code>
<code>pub.3gppnetwork.org</code>	Standard 3GPP domain suffix	-

Example: For MCC 310 (USA), MNC 410 (AT&T):

```
aes.mnc410.mcc310.pub.3gppnetwork.org
```

Storage Configuration

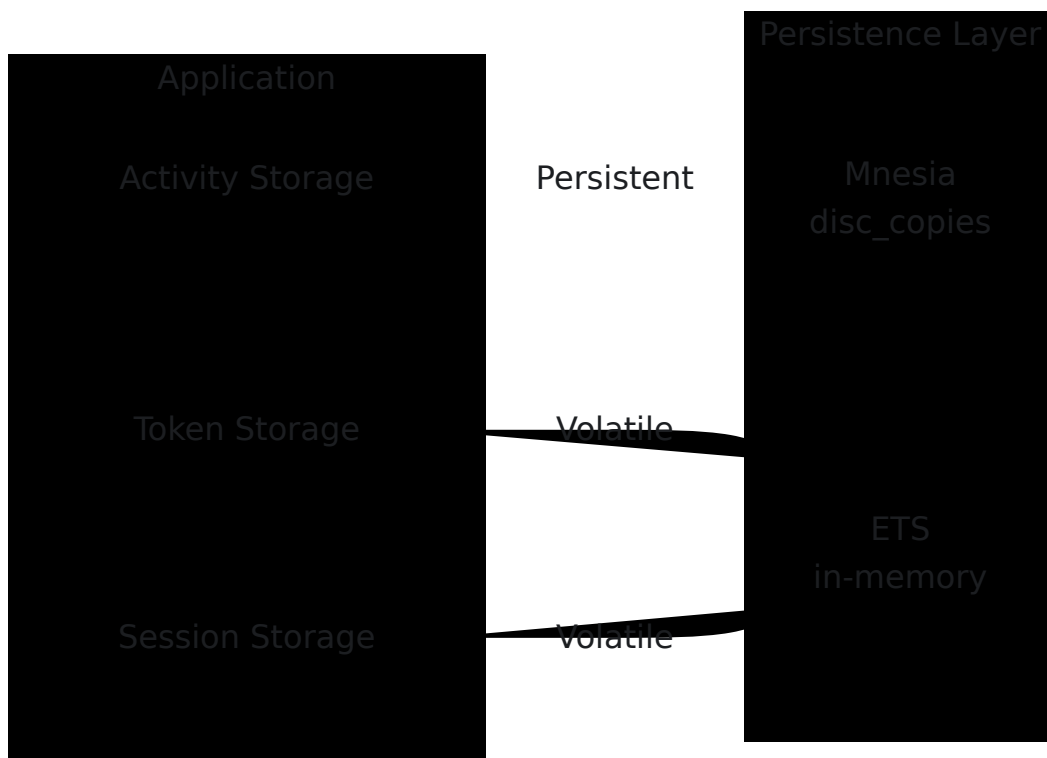
Controls persistent data storage using Mnesia for audit-compliant activity logging.

```
config :omni_sep, :storage,  
      data_dir: "priv/data"
```

Parameters

Parameter	Type	Required	Default	Description
<code>data_dir</code>	String	No	"priv/data"	Directory for Mnesia database files. Activity logs are persisted here and survive application restarts.

Storage Architecture



Mnesia Tables

Activity logging uses Mnesia `disc_copies` tables for persistence:

Table	Purpose	Persistence
<code>activity</code>	Activity records (audit log)	Mnesia <code>disc_copies</code>
<code>activity_by_imsi</code>	IMSI index for fast lookup	Mnesia <code>disc_copies</code>
<code>activity_by_terminal</code>	Terminal ID index	Mnesia <code>disc_copies</code>

Data Directory Structure

```
priv/data/
└─ mnesia/
    ├── schema.DAT          # Mnesia schema
    ├── activity.DCD        # Activity table data
    ├── activity.DCL        # Activity transaction log
    ├── activity_by_imsi.DCD
    ├── activity_by_imsi.DCL
    ├── activity_by_terminal.DCD
    ├── activity_by_terminal.DCL
    ├── DECISION_TAB.LOG
    └─ LATEST.LOG
```

Production Considerations

For production deployments:

1. **Data Directory Permissions:** Ensure the data directory is writable by the application user
2. **Disk Space:** Monitor disk usage as activity logs grow
3. **Backup:** Include the Mnesia directory in backup procedures
4. **Recovery:** Mnesia automatically recovers from transaction logs on startup

```
# config/prod.exs
config :omni_sep, :storage,
  data_dir: "/var/lib/omni_sep/data"
```

Default Entitlements

Defines the entitlement status returned to subscribers who have no custom configuration.


```
config :omni_sep, :default_entitlements,
  # VoWiFi (ap2004) defaults
  vowifi: %{
    entitlement_status: 1,          # ENABLED
    addr_status: 2,                # NOT_REQUIRED
    tc_status: 2,                  # NOT_REQUIRED
    prov_status: 1,                # PROVISIONED
    service_flow_url: "",
    service_flow_user_data: "",
    message_for_incompatible: "VoWiFi service is not available for
your subscription."
  },

  # Voice-over-Cellular (ap2003) defaults
  volte: %{
    entries: [
      %{
        access_type: 1,            # 4G/LTE
        home_roaming_nw_type: 1,   # Home & Roaming
        entitlement_status: 1      # ENABLED
      },
      %{
        access_type: 2,            # 5G/NR
        home_roaming_nw_type: 1,   # Home & Roaming
        entitlement_status: 1,      # ENABLED
        network_voice_irat_capability: "EPS-Fallback"
      }
    ]
  },

  # SMS over IP (ap2005) defaults
  smsoip: %{
    entitlement_status: 1          # ENABLED
  },

  # Data Plan (ap2010) defaults
  data_plan: %{
    entitlement_status: 1
  },

  # Direct Carrier Billing (ap2012) defaults
  dcb: %{
    entitlement_status: 0,          # DISABLED by default
```

```
    tc_status: 2                                # NOT_REQUIRED
},

# Satelllite Mode (ap2016) defaults
satmode: %{
    entitlement_status: 0                        # DISABLED by default
}
```

VoWiFi Parameters (ap2004)

Parameter	Type	Values	Description
entitlement_status	Integer	0=Disabled, 1=Enabled, 2=Incompatible, 3=Provisioning	Service availability status
addr_status	Integer	0=Not Available, 1=Available, 2=Not Required, 3=In Progress	Address verification status
tc_status	Integer	0=Not Available, 1=Available, 2=Not Required, 3=In Progress	Terms & Conditions acceptance status
prov_status	Integer	0=Not Provisioned, 1=Provisioned, 2=Not Required, 3=In Progress	Provisioning status
service_flow_url	String	URL	URL for service flow (address verification T&C)
service_flow_user_data	String	-	User data passed service flow

Parameter	Type	Values	Description
<code>message_for_incompatible</code>	String	-	Message shown w <code>entitlement_sta</code>

VoLTE Parameters (ap2003)

VoLTE configuration uses an array of entries, one per access technology:

Parameter	Type	Values	Description
<code>access_type</code>	Integer	1=4G/LTE, 2=5G/NR	Radio access technology
<code>home_roaming_nw_type</code>	Integer	1=Home & Roaming, 2=Home Only, 3=Roaming Only	Network scope
<code>entitlement_status</code>	Integer	0=Disabled, 1=Enabled	Service availability
<code>network_voice_irat_capability</code>	String	"EPS-Fallback", "VoNR"	5G voice capability

Entitlement Status Values

Value	Name	Description
0	DISABLED	Service not available
1	ENABLED	Service available and ready
2	INCOMPATIBLE	Device or subscription incompatible
3	PROVISIONING	Provisioning in progress

Token Configuration

Controls authentication token generation and validation.

```
config :omni_sep, :token,  
  validity_seconds: 86400,  
  signing_secret: "change_me_in_production"
```

Parameters

Parameter	Type	Required	Default	Description
<code>validity_seconds</code>	Integer	No	86400	Token lifetime in seconds. Default is 24 hours.
<code>signing_secret</code>	String	Yes	-	Secret key for token signing. Must be changed in production. Use a cryptographically random string of at least 32 characters.

Security Considerations

- Generate a unique `signing_secret` for each deployment
- Rotate secrets periodically
- Use environment variables for secrets in production:

```
# config/prod.exs
config :omni_sep, :token,
  signing_secret: System.get_env("OMNI_SEP_TOKEN_SECRET")
```

EAP-AKA Configuration

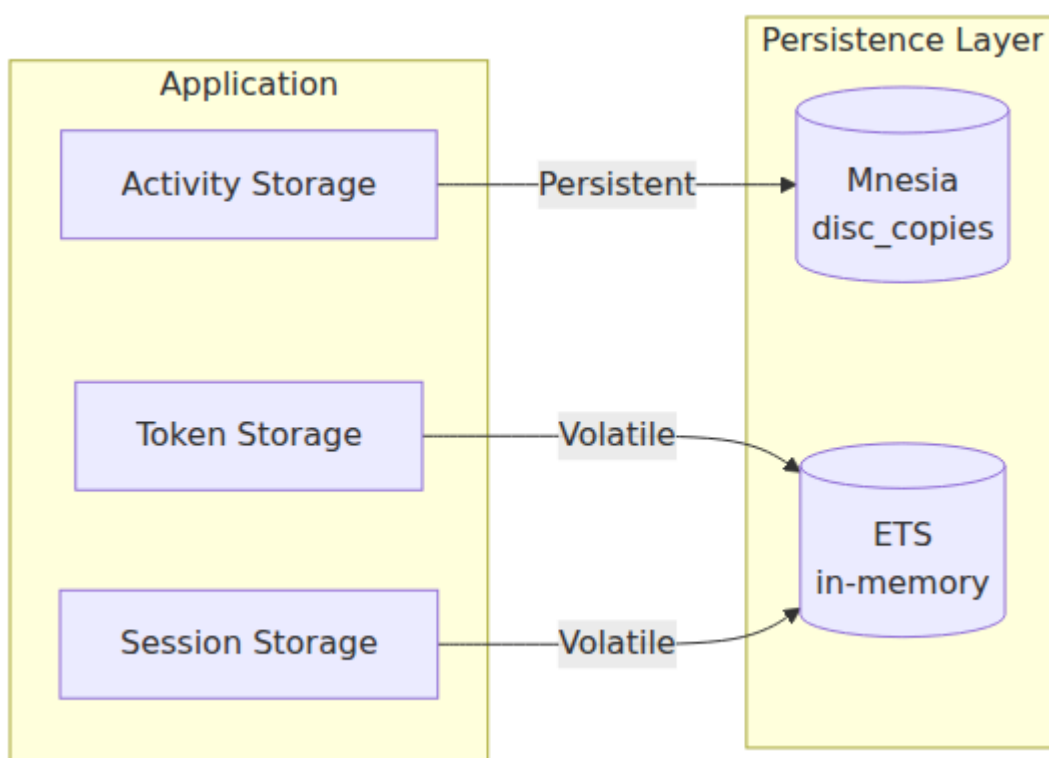
Controls EAP-AKA authentication behavior for initial device authentication.

```
config :omni_sep, :eap_aka,
  enabled: true,
  session_timeout_ms: 30_000
```

Parameters

Parameter	Type	Required	Default	Description
<code>enabled</code>	Boolean	No	true	Enable EAP-AKA authentication. When false, only token-based auth is accepted.
<code>session_timeout_ms</code>	Integer	No	30000	EAP session timeout in milliseconds. Session expires if the client does not respond within this time.

EAP-AKA Flow



Diameter Configuration

Configures the Diameter client for EAP-AKA authentication via SWm interface.

```
config :diameter_ex, :diameter,
  service_name: :omni_sep_aaa,
  listen_ip: {127, 0, 0, 1},
  listen_port: 3868,
  host: "omnisep.example.com",
  realm: "example.com",
  product_name: "OmniSep",
  vendor_id: 10415,
  auth_application_ids: [16777265],
  acct_application_ids: [],
  supported_vendor_ids: [10415],
  request_timeout: 5000,
  allow_undefined_peers_to_connect: true,
  peer_selection_algorithm: :round_robin,
  control_module: OmniSep.Diameter.Control,
  processor_module: OmniSep.Diameter.Processor,
  applications: [
    %{
      alias: :swm,
      dictionary: :diameter_gen_base_rfc6733,
      module: OmniSep.Diameter.Swm
    }
  ],
  peers: [
    %{
      host: "aaa01.example.com",
      ip: "192.168.1.10",
      port: 3868,
      transport: :tcp
    }
  ]
]
```


Core Parameters

Parameter	Type	Required	Default	Description
<code>service_name</code>	Atom	Yes	-	Internal service identifier
<code>host</code>	String	Yes	-	Diameter Origin-Host. Must be resolvable FQDN. RFC 6733 .
<code>realm</code>	String	Yes	-	Diameter Origin-Realm. Used for routing. RFC 6733 .
<code>vendor_id</code>	Integer	Yes	10415	3GPP Vendor ID per TS 29.061 .
<code>auth_application_ids</code>	List	Yes	-	List of supported authentication application IDs. See RFC 6733 .
<code>request_timeout</code>	Integer	No	5000	Request timeout in milliseconds.

Parameter	Type	Required	Default	Description
<code>peer_selection_algorithm</code>	Atom	No	<code>:round_robin</code>	Peer selection algorithm: <code>:round_robin</code> or <code>:fast</code>

Network Parameters

Parameter	Type	Required	Default	Description
<code>listen_ip</code>	Tuple	No	<code>{127, 0, 0, 1}</code>	IP to bind for incoming Diameter connections
<code>listen_port</code>	Integer	No	3868	Diameter port per RFC 6733

Peer Parameters

Each peer in the `peers` list:

Parameter	Type	Required	Default	Description
<code>host</code>	String	Yes	-	Peer's Diameter Identity (must match peer's Origin-Host exactly)
<code>ip</code>	String	Yes	-	Peer's IP address for TCP/SCTP connection
<code>port</code>	Integer	No	3868	Peer's Diameter port
<code>transport</code>	Atom	No	<code>:tcp</code>	Transport: <code>:tcp</code> or <code>:sctp</code>

SWm Application

The SWm interface (Application ID 16777265) is used for EAP-AKA authentication per [3GPP TS 29.273](#).

Message	Code	Description
DER	268	Diameter-EAP-Request - Carries EAP payload to AAA
DEA	268	Diameter-EAP-Answer - Returns EAP response from AAA

Mock Mode

If no Diameter peers are configured, OmniSEP operates in mock mode:

- EAP-AKA challenges are simulated
- IMSI is extracted from EAP_ID
- Authentication always succeeds
- Useful for development and testing

Activity Logging

Controls subscriber activity logging for audit and troubleshooting.

```
config :omni_sep, :activity,  
  max_records_per_subscriber: 1000,  
  retention_seconds: 2_592_000
```

Parameters

Parameter	Type	Required	Default	Description
<code>max_records_per_subscriber</code>	Integer	No	1000	Maximum activity records kept per subscriber. Oldest records are deleted when limit is exceeded.
<code>retention_seconds</code>	Integer	No	2592000	Activity retention period in seconds. Default is 30 days. Records older than this are automatically cleaned up.

Activity Record Contents

Each activity record contains:

Field	Description
imsi	Subscriber IMSI
terminal_id	Device IMEI/terminal ID
timestamp	Request timestamp
client_ip	Client IP address
user_agent	HTTP User-Agent header
app_ids	Requested application IDs
auth_method	Authentication method used (EAP-AKA, TOKEN)
response_code	HTTP response status code

Environment-Specific Configuration

Development

```
# config/dev.exs
import Config

config :omni_sep,
  http_port: 9014

config :omni_sep, :eap_aka,
  enabled: true # Mock mode - no AAA peers

config :logger, :console,
  level: :debug
```

Production

```
# config/prod.exs
import Config

config :omni_sep,
  http_port: 9014,
  http_ip: {0, 0, 0, 0},
  server_fqdn: System.get_env("OMNI_SEP_FQDN")

config :omni_sep, :token,
  validity_seconds: 86400,
  signing_secret: System.get_env("OMNI_SEP_TOKEN_SECRET")

config :diameter_ex, :diameter,
  host: System.get_env("DIAMETER_HOST"),
  realm: System.get_env("DIAMETER_REALM"),
  peers: [
    %{
      host: System.get_env("AAA_PEER_HOST"),
      ip: System.get_env("AAA_PEER_IP"),
      port: 3868,
      transport: :tcp
    }
  ]

config :logger, :console,
  level: :info
```

Environment Variables

Variable	Description
OMNI_SEP_FQDN	Server FQDN for TS.43
OMNI_SEP_TOKEN_SECRET	Token signing secret
DIAMETER_HOST	Diameter Origin-Host
DIAMETER_REALM	Diameter Origin-Realm
AAA_PEER_HOST	AAA peer hostname
AAA_PEER_IP	AAA peer IP address

Logger Configuration

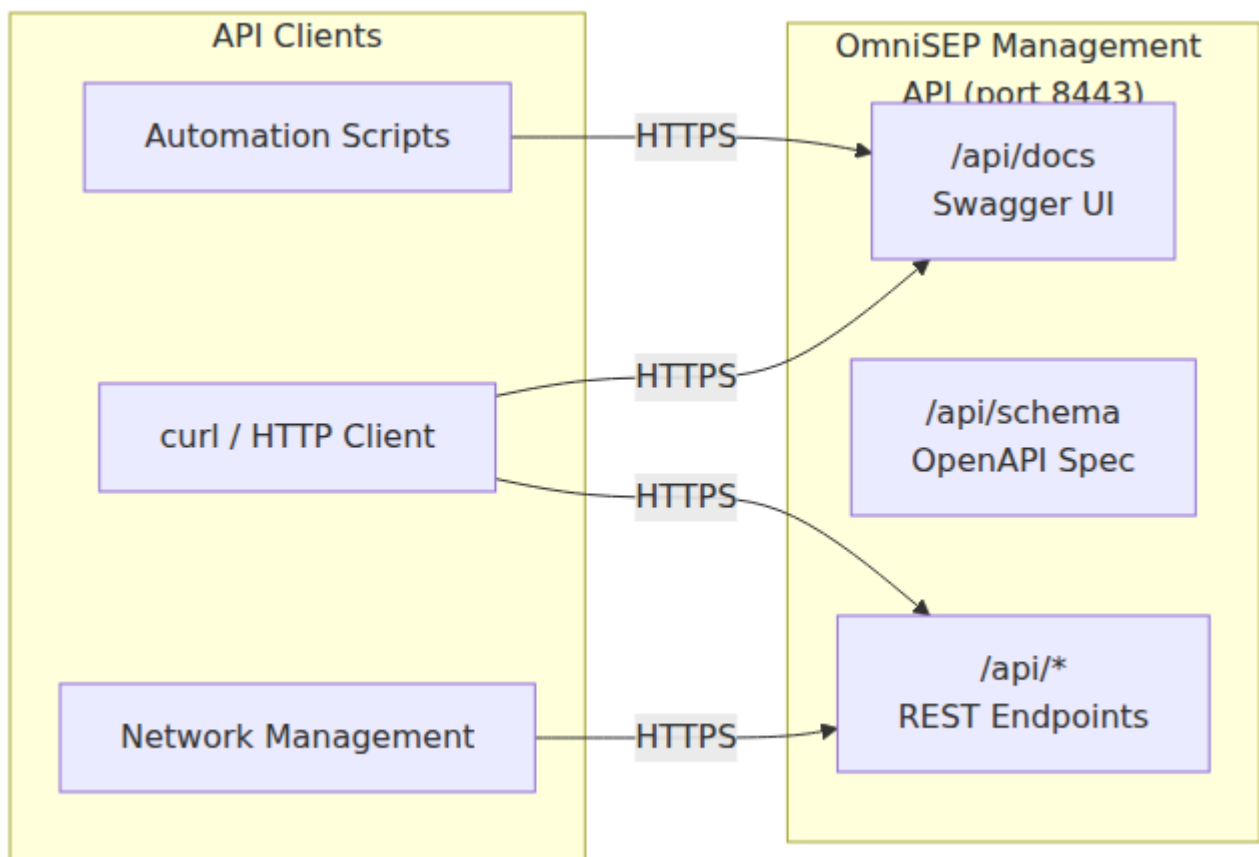
```
config :logger, :console,  
  format: "$time $metadata[$level] $message\n",  
  metadata: [:request_id, :imsi, :terminal_id]
```

Metadata	Description
request_id	Unique request identifier (UUID)
imsi	Subscriber IMSI (when authenticated)
terminal_id	Device terminal ID from request

Management API

OmniSEP provides a RESTful management API for monitoring and administering the platform. The API serves JSON responses over HTTPS on port 8443 and includes auto-generated Swagger/OpenAPI documentation.

Overview



Endpoints Summary

Endpoint	Methods	Description
/api/docs	GET	Swagger UI (interactive API documentation)
/api/schema	GET	OpenAPI 3.0 specification (JSON)
/api/status	GET	System status, uptime, memory, storage counts
/api/activity	GET	Activity log with filtering and pagination
/api/activity/:id	GET	Single activity record detail
/api/entitlements	GET	List subscribers with custom entitlements
/api/entitlements/:id	GET, DELETE	Get or delete entitlements by IMSI
/api/sessions	GET	List active EAP-AKA sessions
/api/sessions/:id	GET, DELETE	Get or expire a session
/api/tokens	GET	List active authentication tokens
/api/tokens/:id	GET, DELETE	Get or revoke a token
/api/diameter	GET	Diameter peer connection status
/api/xcap	GET	List XCAP Simservs profiles

Endpoint	Methods	Description
<code>/api/xcap/:id</code>	GET, DELETE	Get or delete a profile by MSISDN

Authentication

The management API listens on a separate TLS port (8443) from the subscriber-facing HTTP port. Access control is managed at the network level - restrict access to the management port using firewall rules or bind to a management interface only.

TLS Configuration

The API uses the same TLS certificate as other management interfaces:

```
priv/cert/omnitouch.crt  # TLS certificate
priv/cert/omnitouch.pem  # TLS private key
```

All examples in this document use `curl -k` to skip certificate verification for self-signed certificates. In production, use proper CA-signed certificates and remove the `-k` flag.

Status

Returns system health information including uptime, memory usage, storage counts, and service status.

```
GET /api/status
```

```
curl -sk https://localhost:8443/api/status
```

Response

```
{
  "status": "ok",
  "application": "OmniSEP",
  "version": "1.0.0",
  "uptime": "2d 5h 32m",
  "uptime_seconds": 193920,
  "memory": {
    "total_mb": 118.3,
    "processes_mb": 42.9,
    "ets_mb": 3.4
  },
  "storage": {
    "custom_entitlements": 12,
    "xcap_profiles": 45,
    "eap_aka_sessions": 3,
    "active_tokens": 8,
    "activity_records": 1267
  },
  "services": {
    "http_port": 80,
    "eap_aka_enabled": true,
    "vvm_enabled": false
  },
  "timestamp": "2026-04-11T10:35:24.794331Z"
}
```

Response Fields

Field	Type	Description
<code>status</code>	String	Always <code>"ok"</code> when the service is running
<code>application</code>	String	Application name
<code>version</code>	String	Application version
<code>uptime</code>	String	Human-readable uptime (e.g. <code>"2d 5h 32m"</code>)
<code>uptime_seconds</code>	Integer	Uptime in seconds
<code>memory.total_mb</code>	Float	Total BEAM memory usage in megabytes
<code>memory.processes_mb</code>	Float	Memory used by Erlang processes
<code>memory.ets_mb</code>	Float	Memory used by ETS tables
<code>storage.custom_entitlements</code>	Integer	Number of subscribers with custom entitlements
<code>storage.xcap_profiles</code>	Integer	Number of stored XCAP Simservs profiles
<code>storage.eap_aka_sessions</code>	Integer	Number of active EAP-AKA authentication sessions
<code>storage.active_tokens</code>	Integer	Number of active authentication tokens
<code>storage.activity_records</code>	Integer	Total activity log records

Field	Type	Description
<code>services.http_port</code>	Integer	Subscriber-facing HTTP port
<code>services.eap_aka_enabled</code>	Boolean	Whether EAP-AKA authentication is enabled
<code>services.vvm_enabled</code>	Boolean	Whether Visual Voicemail is enabled
<code>timestamp</code>	String	Current server time (ISO 8601)

Activity

Query the activity log for subscriber request history. The activity log records all TS.43 entitlement queries, XCAP operations, and EAP-AKA authentication events.

List Activity Records

```
GET /api/activity
```

```
# Default: first 50 records
```

```
curl -sk https://localhost:8443/api/activity
```

```
# With pagination
```

```
curl -sk "https://localhost:8443/api/activity?page=2&limit=25"
```

```
# Filter by request type
```

```
curl -sk "https://localhost:8443/api/activity?  
type=entitlement_query"
```

```
# Search by IMSI
```

```
curl -sk "https://localhost:8443/api/activity?query=001010555"
```

Query Parameters

Parameter	Type	Default	Description
<code>limit</code>	Integer	50	Number of records per page (max 1000)
<code>page</code>	Integer	1	Page number
<code>type</code>	String	"all"	Filter by request type
<code>query</code>	String	-	Search across IMSI, MSISDN, terminal ID, and client IP (partial match)

Type Filter Values

Value	Description
<code>all</code>	All activity types (default)
<code>entitlement_query</code>	TS.43 entitlement queries
<code>xcap</code>	XCAP Simservs operations
<code>eap_challenge</code>	EAP-AKA challenge requests
<code>eap_response</code>	EAP-AKA response submissions
<code>token_auth</code>	Token-authenticated requests

Response

```

{
  "count": 25,
  "total": 1267,
  "page": 1,
  "page_size": 25,
  "activities": [
    {
      "id": "72420db8-c328-46bd-9878-2a9f3f0d2b88",
      "timestamp": "2026-03-09T04:19:19.461325Z",
      "request_type": "entitlement_query",
      "request_method": "GET",
      "request_path": "/",
      "imsi": "00101055555558",
      "msisdn": null,
      "terminal_id": "terminal_001",
      "terminal_vendor": "Samsung",
      "terminal_model": "SM-S928B",
      "terminal_sw_version": "1.0",
      "client_ip": "10.0.0.1",
      "user_agent": "SEC TS.43/2.0",
      "auth_method": "EAP-AKA",
      "response_status": 200,
      "response_code": 200,
      "app_ids": ["ap2004"],
      "request_headers": {},
      "request_body": null,
      "response_body": null,
      "response_summary": {}
    }
  ]
}

```

Response Fields

Field	Type	Description
count	Integer	Number of records in this page
total	Integer	Total matching records
page	Integer	Current page number
page_size	Integer	Records per page
activities	Array	Activity records (see fields below)

Activity Record Fields

Field	Type	Description
id	String	Unique activity record ID (UUID)
timestamp	String	Request timestamp (ISO 8601)
request_type	String	Type of request (see type filter values)
request_method	String	HTTP method (GET, POST, PUT, DELETE)
request_path	String	HTTP request path
imsi	String	Subscriber IMSI
msisdn	String	Subscriber MSISDN (if available)
terminal_id	String	Device terminal/IMEI identifier
terminal_vendor	String	Device manufacturer
terminal_model	String	Device model
terminal_sw_version	String	Device software version
client_ip	String	Client IP address
user_agent	String	HTTP User-Agent header
auth_method	String	Authentication method used (EAP-AKA, TOKEN)
response_status	Integer	HTTP response status code
response_code	Integer	Application response code
app_ids	Array	Requested application IDs (e.g. ["ap2004"])

Field	Type	Description
request_headers	Object	HTTP request headers
request_body	String	HTTP request body
response_body	String	HTTP response body
response_summary	Object	Summarized response data

Get Activity Record

```
GET /api/activity/:id
```

```
curl -sk https://localhost:8443/api/activity/72420db8-c328-46bd-9878-2a9f3f0d2b88
```

Returns a single activity record with all fields. Returns **404** if the record does not exist.

Entitlements

Manage custom TS.43 entitlements per subscriber. Subscribers without custom entitlements receive the default entitlement configuration.

List Subscribers with Custom Entitlements

```
GET /api/entitlements
```

```
# List all
curl -sk https://localhost:8443/api/entitlements

# Search by IMSI
curl -sk "https://localhost:8443/api/entitlements?query=31038"
```

Query Parameters

Parameter	Type	Default	Description
<code>query</code>	String	-	Filter by IMSI (partial match, case-insensitive)

Response

```
{
  "count": 2,
  "entitlements": [
    {
      "imsi": "310380100000001",
      "app_ids": ["ap2003", "ap2004"],
      "services": ["VoLTE/VoNR", "VoWiFi"]
    },
    {
      "imsi": "310380100000042",
      "app_ids": ["ap2004", "ap2005"],
      "services": ["VoWiFi", "SMSoIP"]
    }
  ]
}
```

Application ID Reference

App ID	Service Name
ap2003	VoLTE/VoNR
ap2004	VoWiFi
ap2005	SMSoIP
ap2006	ODSA Companion
ap2009	ODSA Primary
ap2010	Data Plan
ap2012	DCB (Direct Carrier Billing)
ap2016	Satellite

Get Entitlements for a Subscriber

```
GET /api/entitlements/:imsi
```

```
curl -sk https://localhost:8443/api/entitlements/310380100000001
```

Response

```
{
  "imsi": "3103801000000001",
  "entitlements": {
    "ap2003": {
      "service_name": "VoLTE/VoNR",
      "data": {
        "entries": [
          {
            "access_type": 1,
            "home_roaming_nw_type": 1,
            "entitlement_status": 1
          }
        ]
      }
    },
    "ap2004": {
      "service_name": "VoWiFi",
      "data": {
        "entitlement_status": 1,
        "addr_status": 2,
        "tc_status": 2,
        "prov_status": 1
      }
    }
  }
}
```

Returns `404` if no custom entitlements exist for the given IMSI.

Delete Entitlements for a Subscriber

```
DELETE /api/entitlements/:imsi
```

```
curl -sk -X DELETE
https://localhost:8443/api/entitlements/3103801000000001
```

Removes all custom entitlements for the subscriber. The subscriber will revert to receiving default entitlements.

Response

```
{
  "status": "ok",
  "imsi": "310380100000001"
}
```

Sessions

View and manage active EAP-AKA authentication sessions. Sessions are created when a device initiates EAP-AKA authentication and are short-lived (typically 30 seconds).

List Sessions

```
GET /api/sessions
```

```
curl -sk https://localhost:8443/api/sessions
```

Response

```
{
  "count": 1,
  "sessions": [
    {
      "session_id": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
      "imsi": "310380100000001",
      "state": "challenge_sent",
      "created_at": 1712834400000,
      "eap_id": 1,
      "aaa_session_id": "dra01-sess-001",
      "completed_at": null
    }
  ]
}
```

Session State Values

State	Description
pending	Session created, awaiting Diameter response
challenge_sent	EAP-AKA challenge sent to device
authenticated	Device successfully authenticated
failed	Authentication failed
completed	Session completed and token issued

Get Session

```
GET /api/sessions/:id
```

```
curl -sk https://localhost:8443/api/sessions/a1b2c3d4-e5f6-7890-abcd-ef1234567890
```

Returns `404` if the session does not exist.

Expire Session

```
DELETE /api/sessions/:id
```

```
curl -sk -X DELETE https://localhost:8443/api/sessions/a1b2c3d4-e5f6-7890-abcd-ef1234567890
```

Immediately expires the session. The device will need to restart authentication.

Response

```
{
  "status": "ok",
  "session_id": "a1b2c3d4-e5f6-7890-abcd-ef1234567890"
}
```

Tokens

View and manage active authentication tokens. Tokens are issued after successful EAP-AKA authentication and are used for subsequent entitlement queries.

List Tokens

```
GET /api/tokens
```

```
curl -sk https://localhost:8443/api/tokens
```

Response

```
{
  "count": 1,
  "tokens": [
    {
      "token_id": "f8e7d6c5-b4a3-2190-fedc-ba0987654321",
      "imsi": "310380100000001",
      "created_at": 1712834400,
      "expires_at": 1712920800,
      "expires_in_seconds": 72340,
      "validity": 86400,
      "terminal_id": "terminal_001",
      "app_name": null,
      "auth_method": "EAP-AKA"
    }
  ]
}
```


Token Fields

Field	Type	Description
token_id	String	Unique token identifier
imsi	String	Subscriber IMSI
created_at	Integer	Token creation time (Unix epoch seconds)
expires_at	Integer	Token expiry time (Unix epoch seconds)
expires_in_seconds	Integer	Seconds remaining until expiry
validity	Integer	Total token validity period in seconds
terminal_id	String	Device terminal ID that owns this token
app_name	String	Application name (if scoped)
auth_method	String	How the token was issued (e.g. "EAP-AKA")

Get Token

```
GET /api/tokens/:id
```

```
curl -sk https://localhost:8443/api/tokens/f8e7d6c5-b4a3-2190-fedc-ba0987654321
```

Returns `404` if the token does not exist or has expired.

Revoke Token

```
DELETE /api/tokens/:id
```

```
curl -sk -X DELETE https://localhost:8443/api/tokens/f8e7d6c5-b4a3-2190-fedc-ba0987654321
```

Immediately revokes the token. The device will need to re-authenticate via EAP-AKA.

Response

```
{
  "status": "ok",
  "token_id": "f8e7d6c5-b4a3-2190-fedc-ba0987654321"
}
```

Diameter

Returns Diameter peer connection status for monitoring 3GPP AAA connectivity.

```
GET /api/diameter
```

```
curl -sk https://localhost:8443/api/diameter
```

Response

```
{
  "status": "configured",
  "connected_count": 1,
  "disconnected_count": 0,
  "peers": [
    {
      "host": "dra01.epc.mnc001.mcc001.3gppnetwork.org",
      "status": "Connected",
      "ip": "10.179.2.233",
      "port": 3868,
      "transport": "tcp",
      "connection_initiation": "OmniSEP -> Peer",
      "realm": "epc.mnc001.mcc001.3gppnetwork.org",
      "product_name": "OmniDRA",
      "application_ids": {}
    }
  ]
}
```

Response Fields

Field	Type	Description
status	String	Diameter status: "configured", "not_configured", or "no_peers"
connected_count	Integer	Number of connected peers
disconnected_count	Integer	Number of disconnected peers
peers	Array	Peer details (see below)

Peer Fields

Field	Type	Description
host	String	Peer's Diameter Origin-Host
status	String	Connection status: "Connected", "Disconnected", or "Unknown"
ip	String	Peer IP address
port	Integer	Peer Diameter port
transport	String	Transport protocol ("tcp" or "sctp")
connection_initiation	String	Who initiates the connection (e.g. "OmniSEP -> Peer")
realm	String	Peer's Diameter realm
product_name	String	Peer's advertised product name
application_ids	Object	Diameter application IDs advertised by the peer

Diameter Status Values

Status	Description
configured	Diameter is configured with one or more peers
not_configured	No Diameter configuration present. EAP-AKA operates in mock mode.
no_peers	Diameter is configured but no peers are defined

XCAP

Manage XCAP Supplementary Services (Simservs) profiles. Profiles contain call forwarding, call barring, and caller ID settings per subscriber.

List Profiles

```
GET /api/xcap
```

```
# List all
curl -sk https://localhost:8443/api/xcap

# Search by MSISDN
curl -sk "https://localhost:8443/api/xcap?query=61400"
```

Query Parameters

Parameter	Type	Default	Description
<code>query</code>	String	-	Filter by MSISDN (partial match, case-insensitive)

Response

```
{
  "count": 1,
  "profiles": [
    {
      "msisdn": "61400000001",
      "profile": {
        "originating_identity_presentation": {"active": true},
        "originating_identity_presentation_restriction":
{"active": false},
        "communication_waiting": {"active": true},
        "communication_diversion": {
          "active": true,
          "rules": []
        },
        "incoming_communication_barring": {"active": false},
        "outgoing_communication_barring": {"active": false}
      }
    }
  ]
}
```

Get Profile

```
GET /api/xcap/:msisdn
```

```
curl -sk https://localhost:8443/api/xcap/61400000001
```

Returns `404` if no profile exists for the given MSISDN.

Delete Profile

```
DELETE /api/xcap/:msisdn
```

```
curl -sk -X DELETE https://localhost:8443/api/xcap/61400000001
```

Removes the XCAP profile from local storage. This does not affect the profile stored in the HSS.

Response

```
{
  "status": "ok",
  "msisdn": "61400000001"
}
```

Error Responses

All endpoints return errors in a consistent format:

```
{
  "error": "Activity record not found"
}
```

HTTP Status Codes

Code	Meaning
200	Success
404	Resource not found
422	Unprocessable entity (validation error)

Configuration

The management API is configured in `config.exs` at compile time for route definitions, and can be overridden at runtime for network settings.

Compile-Time Configuration (config.exs)

```
config :api_ex,  
  api: %{  
    port: 8443,  
    listen_ip: "0.0.0.0",  
    product_name: "OmniSEP",  
    title: "API - OmniSEP",  
    hostname: "localhost",  
    enable_tls: true,  
    tls_cert_path: "priv/cert/omnitouch.crt",  
    tls_key_path: "priv/cert/omnitouch.pem",  
    routes: [...]  
  }
```

Runtime Overrides (runtime.exs)

Port, listen IP, and TLS paths can be overridden at runtime:

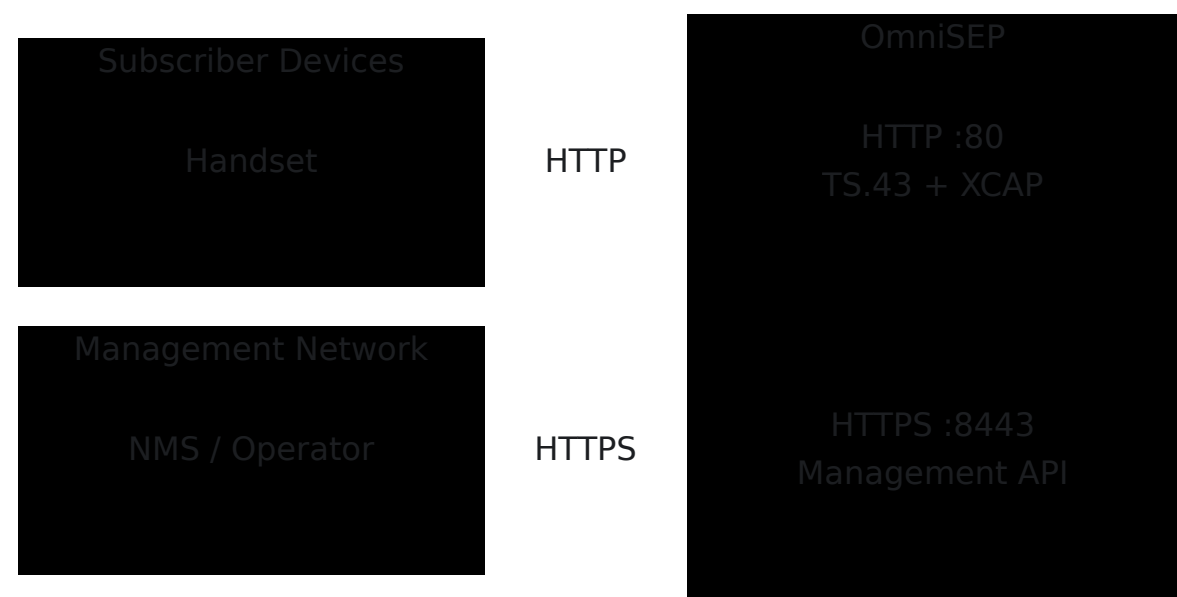
```
config :api_ex,  
  api: %{  
    port: 8443,  
    listen_ip: "0.0.0.0",  
    enable_tls: true,  
    tls_cert_path: "priv/cert/omnitouch.crt",  
    tls_key_path: "priv/cert/omnitouch.pem"  
  }
```


Configuration Parameters

Parameter	Type	Default	Description
<code>port</code>	Integer	8443	HTTPS port for the management API
<code>listen_ip</code>	String	<code>"0.0.0.0"</code>	IP address to bind. Use a management interface IP to restrict access.
<code>product_name</code>	String	<code>"OmniSEP"</code>	Product name shown in API responses and Swagger UI
<code>title</code>	String	<code>"API - OmniSEP"</code>	Title shown in Swagger UI
<code>hostname</code>	String	<code>"localhost"</code>	Hostname for OpenAPI spec
<code>enable_tls</code>	Boolean	<code>true</code>	Enable TLS. Should always be <code>true</code> in production.
<code>tls_cert_path</code>	String	<code>"priv/cert/omnitouch.crt"</code>	Path to TLS certificate file
<code>tls_key_path</code>	String	<code>"priv/cert/omnitouch.pem"</code>	Path to TLS private key file

Parameter	Type	Default	Description
<code>routes</code>	List	-	API route definitions (compile-time only)

Network Ports



Port	Protocol	Purpose
80 (configurable)	HTTP	Subscriber-facing TS.43 entitlements and XCAP
8443 (configurable)	HTTPS	Management API (this document)
3868	TCP	Diameter (SWm interface to AAA/DRA)

OmniSEP

Troubleshooting Guide

Common issues and resolutions for OmniSEP Service Endpoint Platform.

Table of Contents

- [Activity Log](#)
- [TS.43 Entitlement Issues](#)
- [XCAP Simservs Issues](#)
- [Authentication Issues](#)
- [Connectivity Issues](#)
- [Storage Issues](#)

Activity Log

The Activity Log provides a real-time view of all requests to OmniSEP, including TS.43 entitlement queries and XCAP operations.

Features:

- Filter by request type (XCAP, Entitlement Query, EAP Challenge, etc.)
- Search by IMSI, MSISDN, Terminal ID, or Client IP
- View HTTP method (GET, PUT, POST, DELETE) and response status
- Click any row to see detailed information

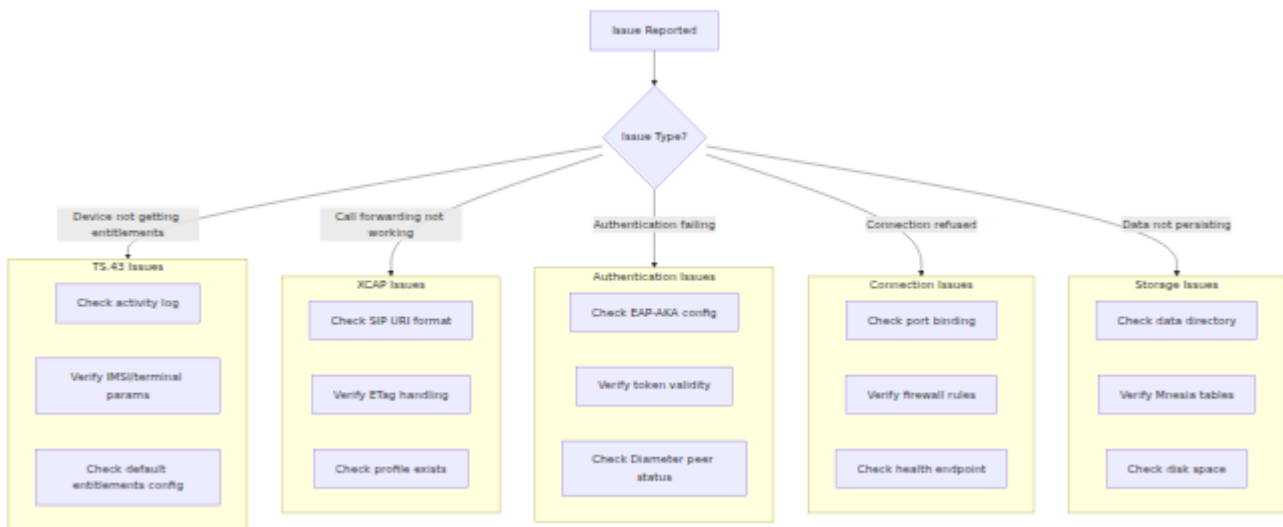
Activity Details Panel

Selecting an activity record shows comprehensive request/response details:

Captured Information:

- **Timestamp and Client IP:** When and where the request originated
- **Request Type and HTTP Method:** XCAP, entitlement query, etc.
- **Subscriber Info:** IMSI, MSISDN (when available)
- **Terminal Info:** Device ID, Vendor, Model (extracted from User-Agent)
- **Request Path:** Full XCAP path or entitlement endpoint
- **User-Agent:** Raw User-Agent header
- **Request Headers:** Content-Type, If-Match, 3GPP headers, etc.
- **Request/Response Body:** XML content for XCAP operations
- **Response Status:** Success, Server Error, Client Error with HTTP code

Diagnostic Workflow



TS.43 Entitlement Issues

Device Reports "Service Not Available"

Symptoms: Device shows VoWiFi/VoLTE as unavailable despite correct subscription

Possible causes:

- Default entitlements configured with `entitlement_status: 0`
- Custom entitlement set for subscriber with disabled status
- Device sending incorrect IMSI or terminal parameters

Resolution:

1. Check the activity log for the subscriber's requests:

```
GET /api/activity?imsi=<subscriber_imsi>
```

2. Verify the entitlement response in activity records
3. Check default entitlements configuration:

```
config :omni_sep, :default_entitlements,  
  vowifi: %{  
    entitlement_status: 1, # Should be 1 for enabled  
    ...  
  }
```

4. Check for custom entitlements overriding defaults:

```
GET /api/entitlements/<imsi>
```

Device Gets Wrong Entitlement Status

Symptoms: Device receives different entitlement values than expected

Possible causes:

- Custom entitlement configured for subscriber
- Wrong app_id being queried
- Configuration mismatch between environments

Resolution:

1. Verify which app_id the device is requesting (check activity log)
2. Common app IDs:

App ID	Service
ap2003	VoLTE/VoNR
ap2004	VoWiFi
ap2005	SMSoIP

3. Check custom entitlements:

```
GET /api/entitlements/<imsi>
```

4. Remove unwanted custom entitlement:

```
DELETE /api/entitlements/<imsi>/<app_id>
```

Missing Parameters in Request

Symptoms: HTTP 400 Bad Request with "Missing parameters" error

Possible causes:

- Device not sending required TS.43 parameters
- Parameters in wrong format
- URL encoding issues

Required parameters:

Parameter	Description
<code>terminal_id</code>	Device IMEI (15 digits)
<code>terminal_vendor</code>	Manufacturer (max 4 chars)
<code>terminal_model</code>	Model name (max 10 chars)
<code>terminal_sw_version</code>	Software version
<code>entitlement_version</code>	Protocol version (typically "2.0")
<code>app</code>	Application ID(s) to query

Resolution:

1. Check activity log for raw request details
2. Verify device is sending all required parameters

3. For Android, ensure User-Agent follows format:

```
PRD-TS43 term-<vendor>/<model> client-IMS-Entitlement/1.0 OS-Android/<version>
```

Version Mismatch (HTTP 406)

Symptoms: Device receives HTTP 406 Not Acceptable

Possible causes:

- Device sending unsupported `entitlement_version`
- Server configured with incompatible version

Resolution:

1. Check server's configured version:

```
config :omni_sep,  
    entitlement_version: "2.0"
```

2. Android devices typically use version "2.0"

3. Ensure server version matches expected device version

XCAP Simservs Issues

Profile Not Found (HTTP 404)

Symptoms: XCAP GET request returns 404

Possible causes:

- Profile never created for subscriber
- Wrong SIP URI format
- MSISDN not linked to profile

Resolution:

1. Check SIP URI format in request:

```
/simservs.ngn.etsi.org/users/sip:+  
<msisdn>@<domain>/simservs.xml
```

2. Verify profile exists via management API:

```
GET /api/xcap/<msisdn>
```

3. Create profile if missing:

```
POST /api/xcap/<msisdn>  
Content-Type: application/json  
  
{  
  "oip": {"active": true},  
  "oir": {"active": true, "default_behaviour": "presentation-  
not-restricted"},  
  "no_reply_timer": 20,  
  "call_forwarding": {},  
  "call_barring_incoming": {},  
  "call_barring_outgoing": {}  
}
```

ETag Mismatch (HTTP 412)

Symptoms: PUT or DELETE request returns HTTP 412 Precondition Failed

Possible causes:

- Client using stale ETag
- Concurrent modification by another client
- ETag not sent with conditional request

Resolution:

1. Fetch current document to get fresh ETag:

```
GET /simservs.ngn.etsi.org/users/<sip_uri>/simservs.xml
```

2. Use returned ETag in `If-Match` header:

```
PUT /simservs.ngn.etsi.org/users/<sip_uri>/simservs.xml
If-Match: "<etag_value>"
Content-Type: application/xcap-el+xml

<simservs>...</simservs>
```

3. For unconditional updates (testing only), omit `If-Match` header

Invalid XML (HTTP 400)

Symptoms: PUT request returns HTTP 400 Bad Request

Possible causes:

- Malformed XML in request body
- Missing required namespaces
- Invalid element structure

Required namespaces:

Prefix	Namespace
(default)	<code>http://uri.etsi.org/ngn/params/xml/simservs/xcap</code>
cp	<code>urn:ietf:params:xml:ns:common-policy</code>

Resolution:

1. Validate XML structure
2. Ensure root element includes required namespaces:

```
<simservs
xmlns="http://uri.etsi.org/ngn/params/xml/simservs/xcap"
xmlns:cp="urn:ietf:params:xml:ns:common-policy">
```

3. Verify element names match ETSI TS 183 023 specification

Call Forwarding Not Activating

Symptoms: Call forwarding settings saved but calls not forwarded

Possible causes:

- Rule disabled in profile
- Wrong condition type
- Target number format incorrect

Resolution:

1. Verify `communication-diversion` is active:

```
<communication-diversion active="true">
```

2. Check rule is correctly structured:

```
<cp:rule id="cfb">
  <cp:conditions>
    <busy/>
  </cp:conditions>
  <cp:actions>
    <forward-to>
      <target>tel:+15557654321</target>
    </forward-to>
  </cp:actions>
</cp:rule>
```

3. Valid rule IDs and conditions:

Rule ID	Condition	Description
cfu	(none)	Unconditional
cfb	busy	On Busy
cfna	no-answer	On No Answer
cfnrc	not-reachable	Not Reachable
cfnl	not-logged-in	Not Logged In

4. Target must use `tel:` URI format with E.164 number

Authentication Issues

EAP-AKA Challenge Not Returned

Symptoms: Initial request with EAP_ID returns error instead of challenge

Possible causes:

- EAP-AKA disabled in configuration
- Invalid EAP_ID format
- Diameter peer connectivity issues

Resolution:

1. Verify EAP-AKA is enabled:

```
config :omni_sep, :eap_aka,
  enabled: true
```

2. Check EAP_ID format (Root NAI):

```
0<IMSI>@nai.epc.mnc<MNC>.mcc<MCC>.3gppnetwork.org
```

Example: `0310410123456789@nai.epc.mnc410.mcc310.3gppnetwork.org`

3. If Diameter peers configured, check connectivity
4. If no peers configured, mock mode should accept any valid format

Token Invalid (HTTP 511)

Symptoms: Request with token returns HTTP 511 Network Authentication Required

Possible causes:

- Token expired
- Token signing secret changed
- Token was generated by different instance

Resolution:

1. Check token validity period:

```
config :omni_sep, :token,  
  validity_seconds: 86400 # 24 hours default
```

2. If secret was rotated, all existing tokens are invalidated
3. Device should fall back to EAP-AKA authentication to get new token
4. Verify all instances use same `signing_secret`

Token Signature Mismatch

Symptoms: Valid-looking token rejected

Possible causes:

- Token from different environment
- Signing secret mismatch between instances
- Token tampered

Resolution:

1. Ensure consistent `signing_secret` across all instances:

```
config :omni_sep, :token,  
      signing_secret: System.get_env("OMNI_SEP_TOKEN_SECRET")
```

2. Use environment variable for secret in production
3. Rotate secrets across all instances simultaneously

Connectivity Issues

Service Not Reachable

Symptoms: Connection refused or timeout

Possible causes:

- Service not running
- Wrong port configuration
- Firewall blocking traffic

Resolution:

1. Check service health:

```
curl http://<host>:9014/health
```

2. Verify port configuration:

```
config :omni_sep,  
  http_port: 9014,  
  http_ip: {0, 0, 0, 0}
```

3. Check firewall rules allow traffic on configured port

4. Verify service is listening:

```
netstat -tlnp | grep 9014
```

Health Check Returns Unhealthy

Symptoms: `/health` endpoint returns non-200 status

Possible causes:

- Dependent service unavailable
- Storage tables not initialized
- Application startup incomplete
- Mnesia data directory not writable

Resolution:

1. Check application logs for startup errors

2. Verify storage initialization:

- Mnesia tables (activity logging)
- ETS tables (entitlements, tokens, sessions, XCAP profiles)

3. Check Mnesia data directory permissions:

```
ls -la priv/data/mnesia/
```

4. If Mnesia fails to start, check for corrupted files and consider clearing the data directory

5. Restart application if tables missing

Slow Response Times

Symptoms: Requests taking longer than expected

Possible causes:

- Diameter peer timeouts
- Activity logging backlog
- High concurrent load

Resolution:

1. Check Diameter peer configuration:

```
config :diameter_ex, :diameter,  
      request_timeout: 5000 # 5 seconds default
```

2. Monitor activity log size per subscriber
3. Check ETS table memory usage
4. Consider horizontal scaling for high load

Storage Issues

Activity Log Not Persisting

Symptoms: Activity records lost after restart

Possible causes:

- Mnesia data directory not writable
- Application terminated before Mnesia could flush to disk
- Disk full

Resolution:

1. Verify data directory exists and is writable:

```
ls -la priv/data/mnesia/
```

2. Check for Mnesia transaction logs (.DCL files) - these contain recent writes:

```
ls -la priv/data/mnesia/*.DCL
```

3. Ensure graceful shutdown to allow Mnesia to flush:

```
# Don't use kill -9, use SIGTERM instead  
kill -TERM <pid>
```

4. Check disk space:

```
df -h priv/data/
```

Mnesia Schema Corrupted

Symptoms: Application fails to start with Mnesia errors

Possible causes:

- Power loss or crash during write
- Disk corruption
- Mixed Erlang versions

Resolution:

1. Check logs for specific Mnesia error
2. If data can be lost, clear and reinitialize:

```
rm -rf priv/data/mnesia/*  
# Restart application - tables will be recreated
```

3. If data must be preserved, try Mnesia repair:

```
:mnesia.stop()  
:mnesia.start()  
:mnesia.wait_for_tables([:activity, :activity_by_imsi,  
:activity_by_terminal], 30000)
```

Activity Log Growing Too Large

Symptoms: Disk usage increasing, slow queries

Possible causes:

- High request volume
- Retention period too long
- Per-subscriber limit too high

Resolution:

1. Check current retention settings:

```
config :omni_sep, :activity,  
  max_records_per_subscriber: 1000, # Reduce if needed  
  retention_seconds: 2_592_000      # 30 days, reduce if  
  needed
```

2. Monitor table size:

```
ls -lh priv/data/mnesia/activity.*
```

3. Consider reducing retention period for high-volume deployments

Diagnostic Commands

Health Check

```
curl -s http://localhost:9014/health | jq
```

Expected response:

```
{
  "status": "ok",
  "service": "omni-sep",
  "version": "0.1.0",
  "services": ["entitlements", "xcap"]
}
```

Activity Log Query

```
# Recent activity for subscriber
curl "http://localhost:9014/api/activity?imsi=<imsi>&limit=10"
```

```
# Activity by terminal
curl "http://localhost:9014/api/activity?terminal_id=
<imei>&limit=10"
```

```
# Activity in time range
curl "http://localhost:9014/api/activity?from=<unix_ts>&to=
<unix_ts>"
```

Entitlements Check

```
# Get custom entitlements
curl http://localhost:9014/api/entitlements/<imsi>

# Set custom entitlement
curl -X POST http://localhost:9014/api/entitlements/<imsi> \
  -H "Content-Type: application/json" \
  -d '{"app_id": "ap2004", "entitlement": {"entitlement_status": 1}}'
```

XCAP Profile Check

```
# Get profile
curl http://localhost:9014/api/xcap/<msisdn>

# Get full simservs document
curl "http://localhost:9014/simservs.ngn.etsi.org/users/sip:+<msisdn>@domain/simservs.xml"
```

Log Analysis

Key Log Patterns

Pattern	Meaning
<code>[info] GET / ...</code>	Incoming entitlement request
<code>[info] POST / ...</code>	Incoming POST request (EAP or entitlement)
<code>[warning] Missing parameters</code>	Request validation failed
<code>[error] EAP session timeout</code>	EAP-AKA not completed in time
<code>[debug] Token validated</code>	Successful token authentication

Enabling Debug Logging

```
# config/dev.exs or runtime
config :logger, :console,
  level: :debug,
  metadata: [:request_id, :imsi, :terminal_id]
```

Reference Specifications

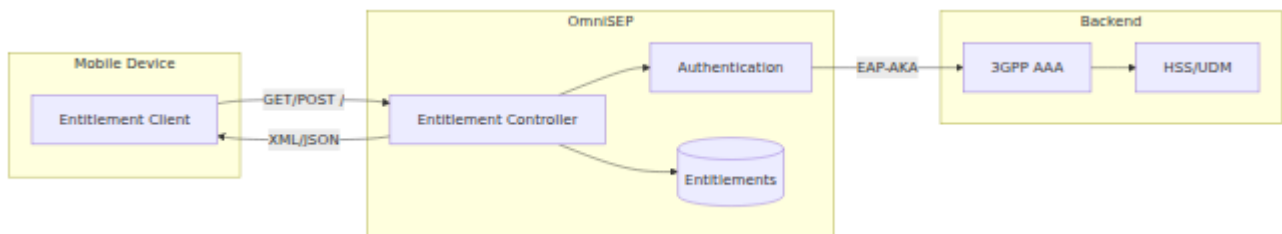
For protocol-specific troubleshooting, refer to:

Protocol	Specification
TS.43	GSMA TS.43
XCAP	RFC 4825
Simservs	ETSI TS 183 023
EAP-AKA	RFC 4187
Diameter	RFC 6733

TS.43 Entitlement Configuration

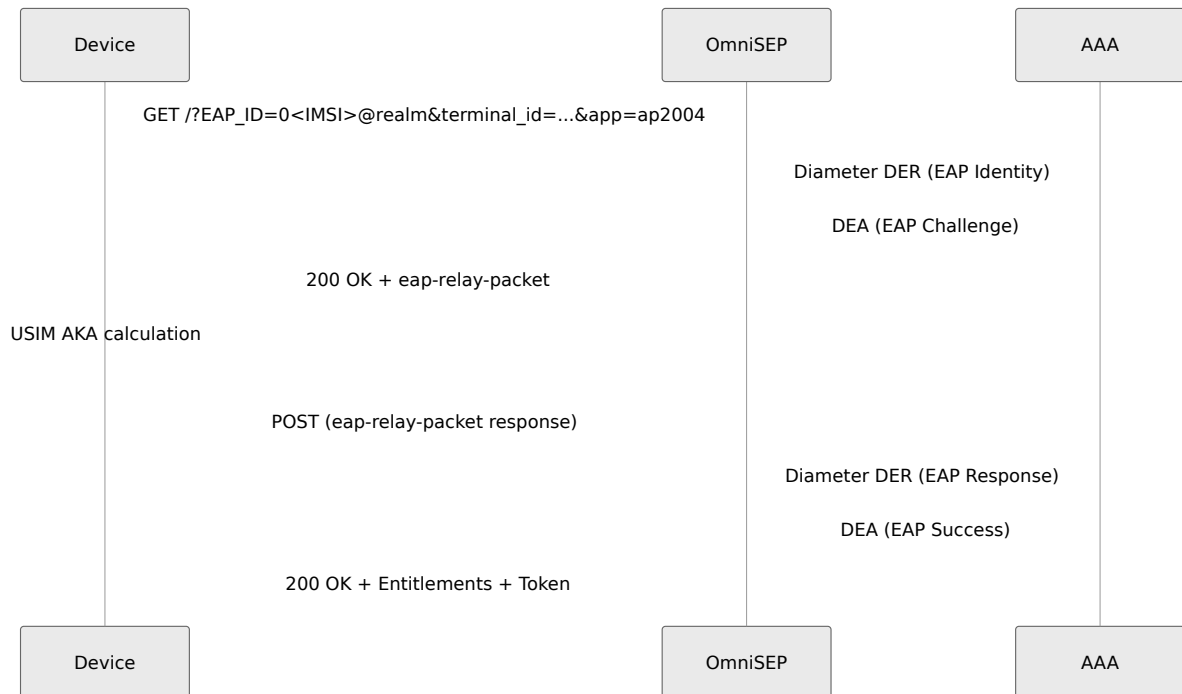
OmniSEP implements GSMA TS.43 Service Entitlement Configuration, enabling mobile devices to query their service entitlements (VoWiFi, VoLTE, SMS, etc.) from the carrier network.

Overview

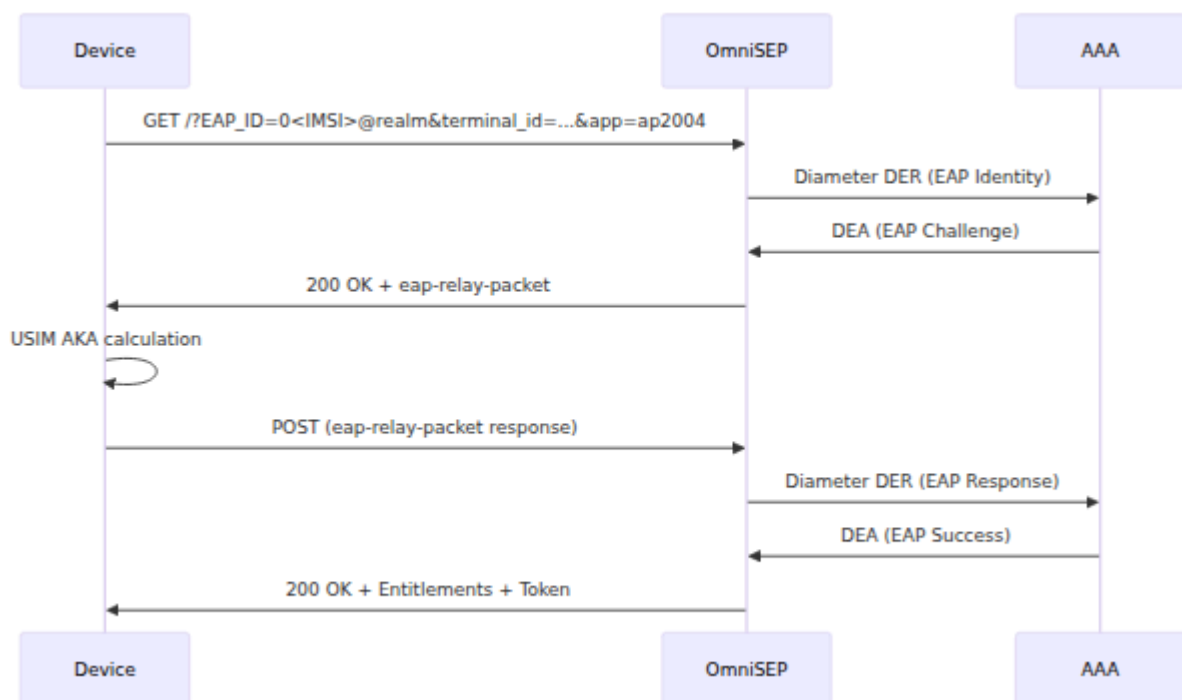


Request Flow

Initial Authentication (EAP-AKA)



Fast Re-Authentication (Token)



HTTP Interface

Endpoint

Method	Path	Content-Type
GET	/	Query parameters
POST	/	application/json

Required Parameters

Parameter	Type	Description
terminal_id	String	Device IMEI (15 digits)
terminal_vendor	String	Device manufacturer (max 4 chars per RCC.14)
terminal_model	String	Device model (max 10 chars per RCC.14)
terminal_sw_version	String	Software version (max 20 chars per RCC.14)
entitlement_version	String	Protocol version (e.g., "2.0")
app	String/List	Application ID(s) to query

Authentication Parameters

One of these authentication methods is required:

Parameter	Description
EAP_ID	Root NAI for EAP-AKA: 0<IMSI>@nai.epc.mnc<MNC>.mcc<MCC>.3gppnetwork.org
token	Authentication token from previous response
temporary_token	Temporary token
operator_token	Operator-issued token

Optional Parameters

Parameter	Type	Description
IMSI	String	Required with token for fast re-auth
app_name	String	Application name
app_version	String	Application version
notif_action	Integer	Notification action (0=disable, 1=GCM, 2=FCM)
notif_token	String	Push notification token
vers	Integer	Configuration version for change detection

Request Examples

GET with Token:

```
GET /?
terminal_id=123456789012345&terminal_vendor=Goog&terminal_model=Pixel
```

POST with EAP-AKA:

```
{
  "terminal_id": "123456789012345",
  "terminal_vendor": "Goog",
  "terminal_model": "Pixel8",
  "terminal_sw_version": "14.0",
  "entitlement_version": "2.0",
  "app": "[ap2003,ap2004]",
  "EAP_ID":
  "0310410123456789@nai.epc.mnc410.mcc310.3gppnetwork.org"
}
```

Response Formats

Accept Header

Accept Header	Response Format
text/vnd.wap.connectivity+xml	XML (default)
application/json	JSON
application/vnd.gsma.eap-relay.v1.0+json	EAP relay JSON

XML Response

```
<?xml version="1.0"?>
<wap-provisioningdoc version="1.1">
  <characteristic type="VERS">
    <parm name="version" value="2"/>
    <parm name="validity" value="86400"/>
  </characteristic>
  <characteristic type="TOKEN">
    <parm name="token" value="eyJ0eXAi..."/>
    <parm name="validity" value="86400"/>
  </characteristic>
  <characteristic type="APPLICATION">
    <parm name="AppID" value="ap2004"/>
    <characteristic type="ap2004">
      <parm name="EntitlementStatus" value="1"/>
      <parm name="AddrStatus" value="2"/>
      <parm name="TC_Status" value="2"/>
      <parm name="ProvStatus" value="1"/>
    </characteristic>
  </characteristic>
</wap-provisioningdoc>
```

JSON Response

```
{
  "Vers": {
    "version": "2",
    "validity": "86400"
  },
  "Token": {
    "token": "eyJ0eXAi...",
    "validity": "86400"
  },
  "ap2004": {
    "EntitlementStatus": "1",
    "AddrStatus": "2",
    "TC_Status": "2",
    "ProvStatus": "1"
  }
}
```

EAP Challenge Response

When EAP-AKA authentication is in progress:

```
{
  "eap-relay-packet": "AQEALBcBAAAn..."
}
```

Application IDs

Supported Applications

App ID	Service	Reference
ap2003	Voice-over-Cellular (VoLTE/VoNR)	TS.43 Section 4
ap2004	Voice-over-WiFi (VoWiFi)	TS.43 Section 3
ap2005	SMS-over-IP	TS.43 Section 5
ap2006	ODSA Companion	TS.43 Section 6
ap2009	ODSA Primary	TS.43 Section 6
ap2010	Data Plan Boost	TS.43 Section 7
ap2011	Server Initiated Requests	TS.43 Section 8
ap2012	Direct Carrier Billing	TS.43 Section 9
ap2013	Private User Identity	TS.43 Section 10
ap2014	Phone Number Information	TS.43 Section 11
ap2016	Satellite Entitlement	TS.43 Section 12

VoWiFi (ap2004) Response Fields

Field	Type	Values	Description
EntitlementStatus	Integer	0-3	Service availability
AddrStatus	Integer	0-3	E911 address verification status
TC_Status	Integer	0-3	Terms & Conditions status
ProvStatus	Integer	0-3	Provisioning status
ServiceFlow_URL	String	URL	Service flow URL
ServiceFlow_UserData	String	-	Data for service flow

Status Values:

Value	EntitlementStatus	AddrStatus/TC_Status/ProvStatus
0	Disabled	Not Available
1	Enabled	Available
2	Incompatible	Not Required
3	Provisioning	In Progress

VoLTE (ap2003) Response Fields

VoLTE uses an array of entries per access technology:

Field	Type	Values	Description
AccessType	Integer	1=LTE, 2=NR	Radio access technology
HomeRoamingNWType	Integer	1-3	Network scope
EntitlementStatus	Integer	0-1	Service availability
NetworkVoiceIRATCapability	String	-	Voice capability (5G only)

HomeRoamingNWType Values:

Value	Meaning
1	Home and Roaming
2	Home Only
3	Roaming Only

Error Responses

HTTP Status	Meaning	Description
400	Bad Request	Missing required parameters
403	Forbidden	Authentication failed
406	Not Acceptable	Unsupported protocol version
511	Network Authentication Required	Token invalid or EAP-AKA required

Error Response Format

```
HTTP/1.1 400 Bad Request
Content-Type: text/plain
```

```
Bad Request: Missing parameters ["terminal_id"]
```

Custom Entitlements

Setting Custom Entitlements

Use the management API to set custom entitlements for specific subscribers:

```
POST /api/entitlements/{imsi}
Content-Type: application/json
```

```
{
  "app_id": "ap2004",
  "entitlement": {
    "entitlement_status": 0,
    "addr_status": 1,
    "tc_status": 1,
    "prov_status": 0,
    "service_flow_url": "https://activate.example.com/vowifi",
    "message_for_incompatible": "VoWiFi requires address
verification"
  }
}
```

Retrieving Entitlements

```
GET /api/entitlements/{imsi}
```

Returns all custom entitlements for the subscriber:

```
{
  "imsi": "310410123456789",
  "entitlements": {
    "ap2004": {
      "entitlement_status": 1,
      "addr_status": 2,
      "tc_status": 2,
      "prov_status": 1
    }
  }
}
```

Activity Logging

All entitlement requests are logged for audit purposes.

Searching Activity

GET /api/activity?
imsi=310410123456789&from=1704067200&to=1704153600&limit=100

Parameter	Type	Description
imsi	String	Filter by subscriber IMSI
terminal_id	String	Filter by device terminal ID
from	Integer	Start timestamp (Unix epoch)
to	Integer	End timestamp (Unix epoch)
limit	Integer	Maximum records to return
offset	Integer	Pagination offset

Activity Record

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "timestamp": "2024-01-15T10:30:00Z",
  "imsi": "310410123456789",
  "terminal_id": "123456789012345",
  "terminal_vendor": "Google",
  "terminal_model": "Pixel8",
  "app_ids": ["ap2003", "ap2004"],
  "client_ip": "192.168.1.100",
  "user_agent": "PRD-TS43 Goog/Pixel8 client-IMS-Entitlement/1.0
OS-Android/14.0",
  "auth_method": "TOKEN",
  "response_code": 200
}
```

Android Client Compatibility

OmniSEP is tested against the Android AOSP `service_entitlement` library.

User-Agent Format

Android devices use this User-Agent format:

```
PRD-TS43 term-<vendor>/<model> <client_ts43>/<app_version> OS-  
Android/<sw_version>
```

Example:

```
PRD-TS43 term-Google/Pixel8 client-IMS-Entitlement/1.0 OS-  
Android/14.0
```

Multiple App IDs

Android sends multiple app IDs in POST requests as a bracket-enclosed string:

```
{  
  "app": "[ap2003,ap2004]"  
}
```

OmniSEP parses both formats:

- `"ap2003,ap2004"` - Comma-separated
- `"[ap2003,ap2004]"` - Bracket-enclosed (Android format)
- `["ap2003", "ap2004"]` - JSON array

Reference Specifications

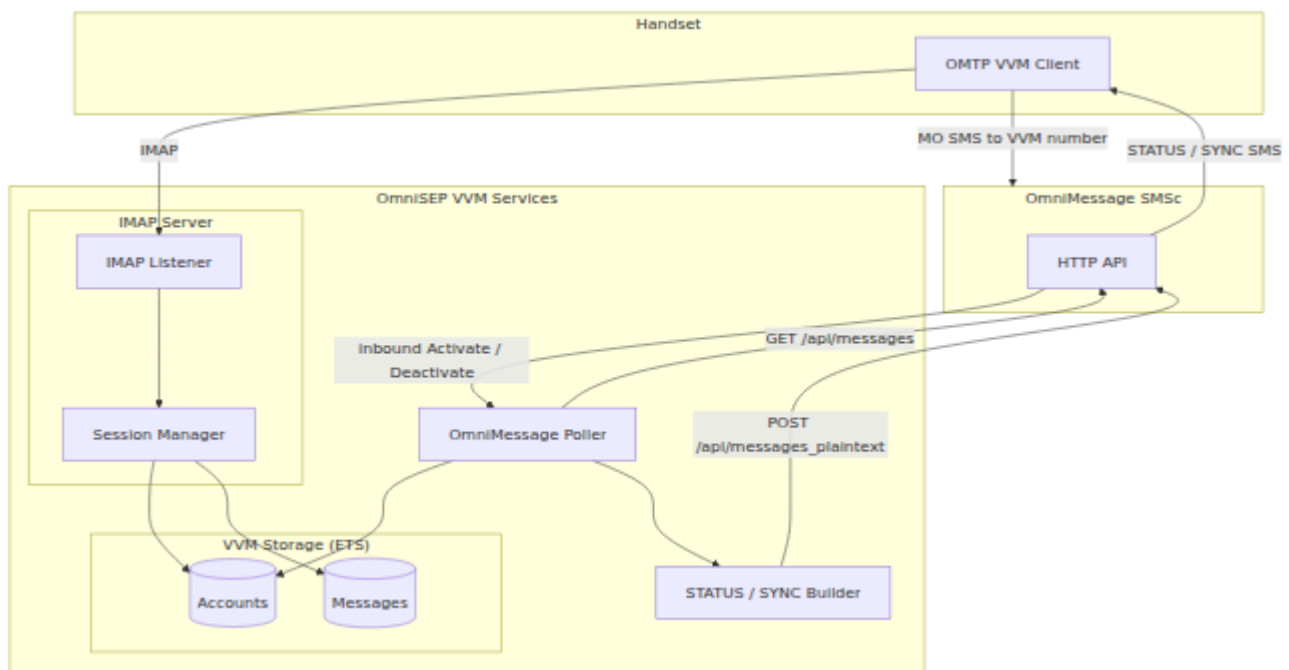
Specification	Description
GSMA TS.43	Service Entitlement Configuration
GSMA RCC.14	IMS Device Configuration Guidelines
3GPP TS 33.220	Generic Bootstrapping Architecture (GBA)
3GPP TS 29.273	EPS AAA Interfaces (SWm)

Visual Voicemail (VVM)

OmniSEP provides a Visual Voicemail server. It implements the OMTF VVM specification and GSMA TS.46. The service lets a handset manage voicemail through an IMAP interface. OmniSEP provisions each handset over SMS and delivers the voicemail audio and transcript over IMAP.

OmniSEP does not connect to the SMSC directly. It uses the OmniMessage HTTP API for all SMS. OmniSEP polls OmniMessage for inbound VVM SMS from handsets. OmniSEP sends **STATUS** and **SYNC** SMS to handsets through the same API.

Architecture Overview

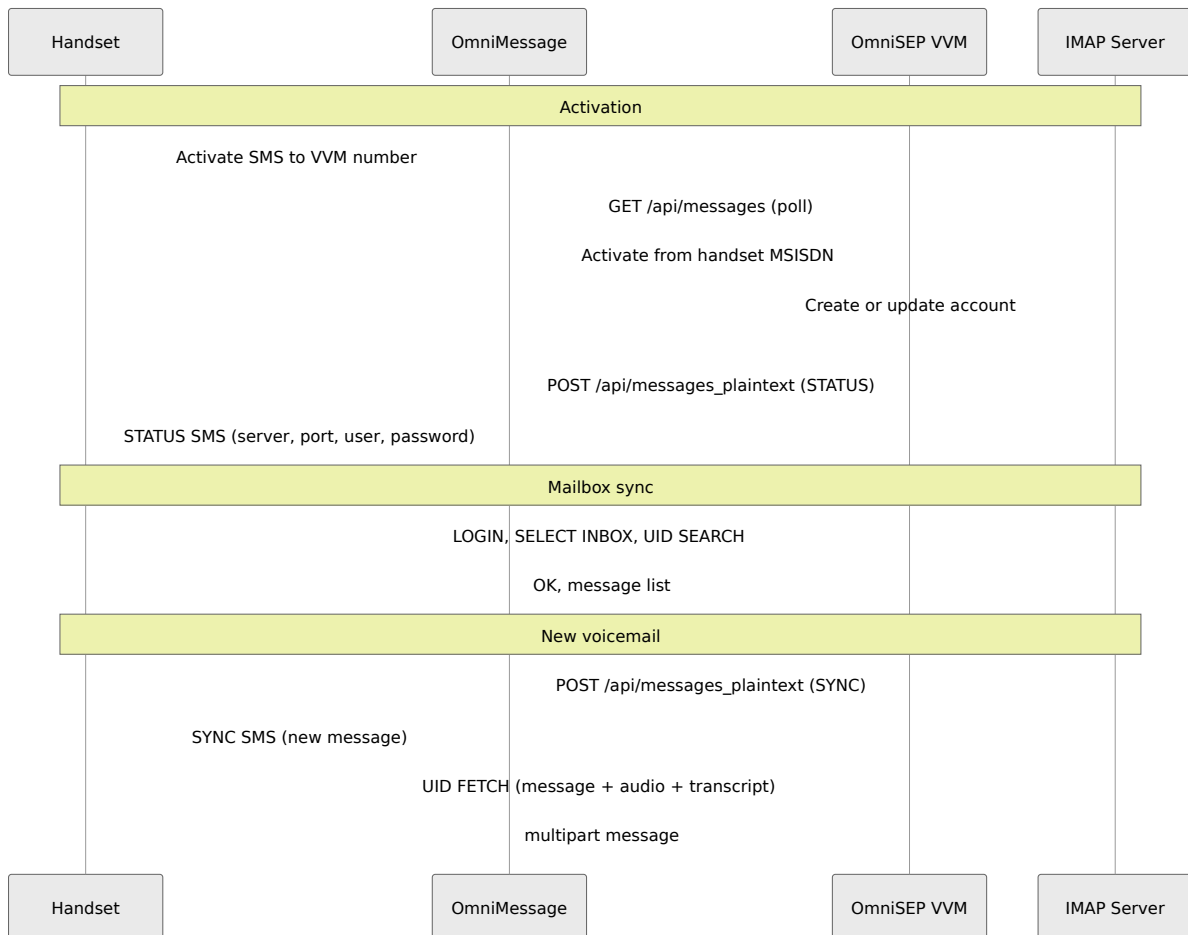


The service has three parts:

- **OmniMessage Poller** reads inbound VVM SMS from OmniMessage and provisions accounts.
- **STATUS / SYNC Builder** sends OMTF SMS to the handset through OmniMessage.
- **IMAP Server** serves the mailbox, the voicemail audio, and the transcript.

Provisioning Flow

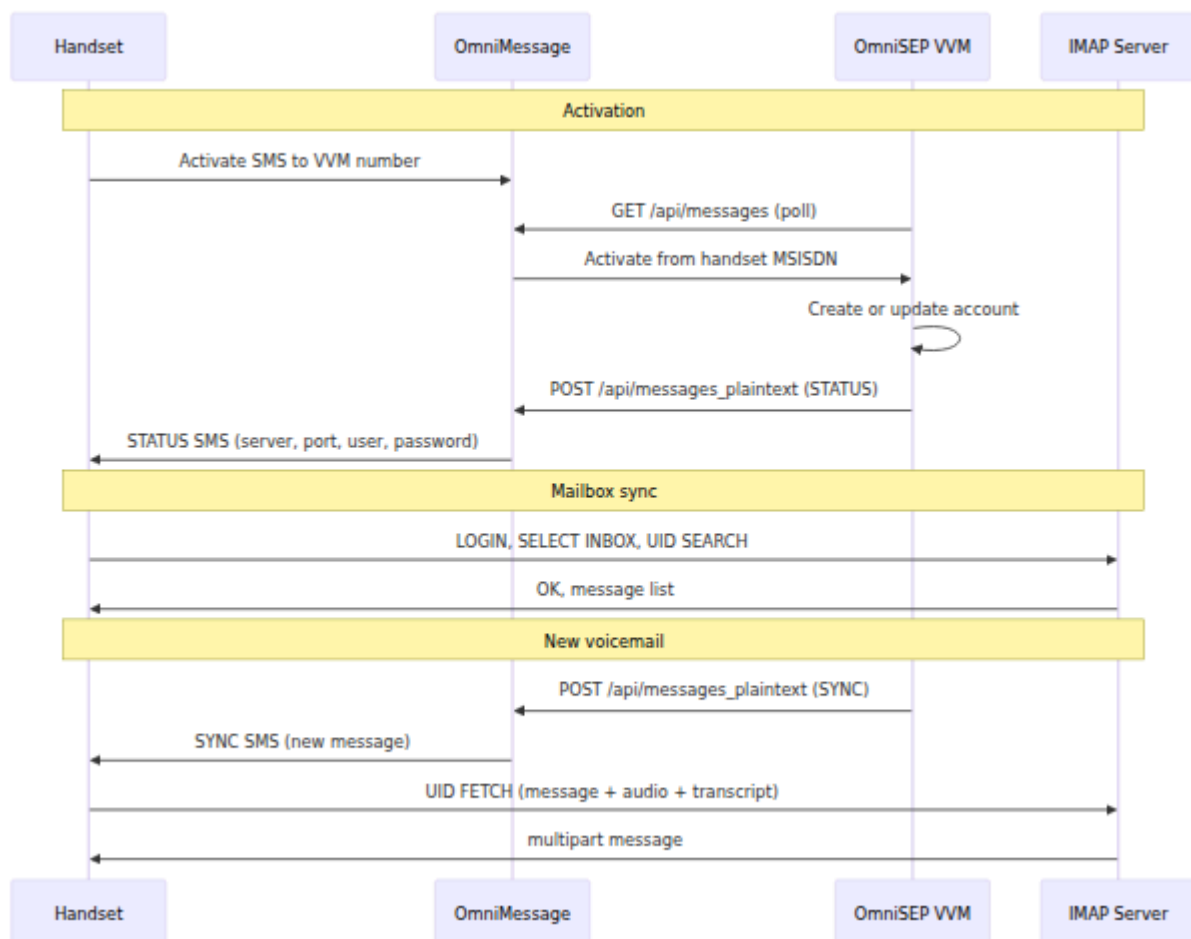
The handset sends an **Activate** SMS to the VVM number. OmniSEP reads that SMS from OmniMessage, creates the account, and sends a **STATUS** SMS with the IMAP credentials. The handset then connects over IMAP.



Provisioning States

The service tracks a provisioning state for each subscriber. The state follows the OMTP specification.

State	Code	Description
Unknown	U	Initial state. No provisioning attempted.
New	N	Account created. The service waits for the first client connection.
Ready	R	Fully provisioned and operational.
Provisioned	P	Credentials sent. The service waits for client verification.
Blocked	B	Service deactivated or suspended.



OmniMessage Integration

OmniSEP uses the OmniMessage HTTP API for all VVM SMS. See the [OmniMessage API](#) for the full schema.

Inbound SMS (poll)

OmniSEP polls `GET /api/messages` on a fixed interval. It selects messages whose `destination_msisdn` is the configured VVM number. It processes messages whose text starts with `Activate:`, `Deactivate:`, or `//VVM`. It records each processed message id, so it does not process a message twice. If a message fails to process, OmniSEP retries it on the next poll.

An inbound `Activate` message from the handset looks like this:

```
Activate:pv=11;ct=samsung.SM-A536E.13
```

Outbound SMS (send)

OmniSEP sends `STATUS` and `SYNC` SMS with `POST /api/messages_plaintext`. The request body carries the source number (the VVM number), the destination number (the handset MSISDN), and the message text.

```
{
  "source_msisdn": "123",
  "destination_msisdn": "15551234212",
  "message":
    "//VVM:STATUS:st=R;rc=0;srv=10.179.4.165;ipt=9143;..."
}
```

SMS Protocol

STATUS Message (server to handset)

OmniSEP sends a `STATUS` message after activation. The message carries the IMAP credentials.

```
//VVM:STATUS:st=R;rc=0;srv=10.179.4.165;ipt=9143;spt=8025;u=15551234215;pm=N;gm=N;vtc=A
```

Field	Description
st	Provisioning state. R=Ready, B=Blocked, N=New, P=Provisioned, U=Unknown.
rc	Return code. 0 is success.
srv	IMAP server address. The handset connects to this address.
ipt	IMAP port.
spt	SMTP port.
u	IMAP username.
pw	IMAP password.
tui	Telephone user interface number.
dn	Destination number for client SMS replies.
lang	Language code.
g_len	Maximum greeting length in seconds.
vs_len	Maximum voice signature length in seconds.
pw_len	PIN length range.
pm	PIN required. Y or N.
gm	Greeting reset mode. G=greeting, V=voice signature, B=both, N=none.
vtc	Transcription capability. A=automatic, D=on-demand, B=both, N=none.

SYNC Message (server to handset)

OmniSEP sends a **SYNC** message when the mailbox content changes.

```
//VVM:SYNC:ev=NM;id=1;c=1;t=v;s=15551234567;dt=19/08/2026 01:39+0000;l=8
```

Field	Description
ev	Event type. NM=new message, MBU=mailbox update, GU=greeting update.
id	Message id.
c	Unread message count.
t	Message type. v=voice, o=video, f=fax, i=infotainment, e=ECC.
s	Sender number.
dt	Deposit timestamp.
l	Message length in seconds.

Activate Message (handset to server)

The handset sends an **Activate** message to enable VVM.

```
Activate:pv=11;ct=samsung.SM-A536E.13
```

Field	Description
<code>pv</code>	Protocol version.
<code>ct</code>	Client type. The format is vendor.model.os_version.

Deactivate Message (handset to server)

The handset sends a `Deactivate` message to disable VVM.

```
Deactivate:pv=11
```

IMAP Server

The IMAP server implements a subset of IMAP4rev1 ([RFC 3501](#)) for voicemail. It accepts plain IMAP or IMAPS.

Capabilities

```
IMAP4rev1 AUTH=PLAIN AUTH=LOGIN UIDPLUS MOVE QUOTA
```

Supported Commands

Command	Description
CAPABILITY	List server capabilities.
LOGIN	Authenticate with username and password.
LOGOUT	End the session.
SELECT	Open a mailbox.
EXAMINE	Open a mailbox read-only.
LIST	List the available mailboxes.
STATUS	Get mailbox status.
FETCH	Retrieve message content.
UID FETCH	Retrieve by UID.
STORE	Update message flags.
UID STORE	Update flags by UID.
SEARCH	Search messages.
UID SEARCH	Search by UID.
COPY	Copy messages between mailboxes.
EXPUNGE	Delete flagged messages.
CLOSE	Close the mailbox and expunge.
GETQUOTAROOT	Get the storage quota.

Command	Description
NOOP	Keep-alive.

Message Format

OmniSEP presents each voicemail as a MIME `multipart/mixed` message. Part 1 is the audio. Part 2 is the transcript. The handset reads the top-level `Content-Type` header to confirm the message is multipart. The handset then extracts the `audio/*` part.

```
Content-Type: multipart/mixed; boundary="-----=_Part_0"

-----=_Part_0
Content-Type: audio/amr
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="voicemail.amr"

[base64 audio]
-----=_Part_0
Content-Type: text/plain; charset="UTF-8"

[transcript text]
-----=_Part_0--
```

Voicemail Fetch

The handset fetches a voicemail in more than one step. The exact commands depend on the client. The OMTP client on Android uses this sequence:

1. `UID SEARCH UID <id>` to confirm the message exists.
2. `UID FETCH <id> (UID BODY.PEEK[])` to read the full message.
3. `UID FETCH <id> (... BODY.PEEK[HEADER.FIELDS (...)] BODYSTRUCTURE)` to read the headers and structure.
4. `UID FETCH <id> (UID BODY.PEEK[<n>])` to read one part, for example the transcript.

The client requires the `Content-Type` header to start with `multipart/`. If the header is absent, the client logs "Ignored non multi-part message" and drops the voicemail. OmniSEP therefore returns the full multipart message for the header-fields fetch, so the client sees the correct `Content-Type`.

Configuration

Configuration Example

```
config :omni_sep, :vvm,  
  # Master switch  
  enabled: true,  
  
  # IMAP server  
  imap_port: 9143,  
  imap_ssl: false,  
  imap_server: "10.179.4.165",  
  imap_cert: "priv/cert/server.crt",  
  imap_key: "priv/cert/server.key",  
  
  # SMTP (voicemail deposit, future use)  
  smtp_port: 8025,  
  smtp_server: nil,  
  
  # Telephone user interface  
  tui_number: "*86",  
  
  # SMS via OmniMessage  
  sms_source_number: "123",  
  vvm_number: "123",  
  omnimessage_url: "https://10.179.3.219:8443",  
  sms_gateway: "https://10.179.3.219:8443/api/messages_plaintext",  
  poll_interval_ms: 5000,  
  
  # PIN  
  min_pin_length: 4,  
  max_pin_length: 15,  
  
  # Subscriber limits  
  default_max_messages: 100,  
  default_storage_limit_kb: 50_000,  
  default_max_greeting_seconds: 60
```

Configuration Parameters

Parameter	Type	Required	Default
<code>enabled</code>	Boolean	No	false
<code>imap_port</code>	Integer	No	993
<code>imap_ssl</code>	Boolean	No	true
<code>imap_server</code>	String	Yes	-
<code>imap_cert</code>	String	No	priv/cert/server.crt
<code>imap_key</code>	String	No	priv/cert/server.key
<code>smtp_port</code>	Integer	No	587
<code>smtp_server</code>	String	No	nil
<code>tui_number</code>	String	No	*86
<code>sms_source_number</code>	String	Yes	-

Parameter	Type	Required	Default
vvm_number	String	No	123
omnimessage_url	String	Yes	nil
sms_gateway	String	Yes	nil
poll_interval_ms	Integer	No	5000
min_pin_length	Integer	No	4
max_pin_length	Integer	No	15
default_max_messages	Integer	No	100
default_storage_limit_kb	Integer	No	50000
default_max_greeting_seconds	Integer	No	60

Reachability

The handset connects to the IMAP server over the mobile data bearer. Set `imap_server` to an address that the UE data network can reach. In the lab, the OmniSEP host on the P-CSCF is reachable from the UE pools.

Handset Activation

The handset activates VVM from its carrier configuration. Set the OMTP keys in CarrierConfig. The handset then sends the `Activate` SMS to the destination number.

CarrierConfig key	Value	Description
<code>vvm_type_string</code>	<code>vvm_type_omtp</code>	Select the OMTP protocol.
<code>vvm_destination_number_string</code>	<code>123</code>	The VVM short number. Match this value to <code>vvm_number</code> .
<code>vvm_ssl_enabled_bool</code>	<code>false</code>	Use plain IMAP. Match this value to <code>imap_ssl</code> .

When the CarrierConfig changes, the handset starts VVM activation. It sends the `Activate` SMS to the destination number. OmniSEP reads the SMS from OmniMessage, provisions the account, and returns the `STATUS` SMS.

Handset Experience

After activation, the handset receives the `SYNC` SMS for each new voicemail. The native voicemail client syncs over IMAP, downloads the audio, and shows the voicemail in the Phone app. The following screenshots show a Samsung handset that runs the native OMTP client against OmniSEP.

The voicemail inbox lists each message with the caller number and the deposit time.

The voicemail detail view plays the downloaded audio. The handset streams the audio part that OmniSEP serves over IMAP.

The caller numbers in these screenshots use the North American fictional range (555-0100 to 555-0199).

Metrics

IMAP Session Metrics

Metric: `vvm_imap_sessions_total` **Type:** Counter **Description:** Total number of VVM IMAP sessions. **Labels:**

- `result` - Session outcome: `success`, `auth_failed`, `timeout`.

Metric: `vvm_imap_active_sessions` **Type:** Gauge **Description:** Number of active IMAP sessions.

Metric: `vvm_imap_commands_total` **Type:** Counter **Description:** Total IMAP commands processed. **Labels:**

- `command` - IMAP command, for example `LOGIN`, `SELECT`, `FETCH`.

- `result` - Command result: `ok`, `no`, `bad`.

Message Metrics

Metric: `vvm_messages_total` **Type:** Counter **Description:** Total VVM message operations. **Labels:**

- `operation` - Operation type: `deposit`, `read`, `delete`, `move`.

Metric: `vvm_messages_stored` **Type:** Gauge **Description:** Number of voicemail messages in storage.

Metric: `vvm_message_duration_seconds` **Type:** Histogram **Description:** Duration of voicemail messages in seconds. **Buckets:** 5, 10, 15, 30, 60, 120, 180, 300.

SMS Metrics

Metric: `vvm_sms_total` **Type:** Counter **Description:** Total VVM SMS messages. **Labels:**

- `type` - SMS type: `status`, `sync`, `activate`, `deactivate`.
- `result` - Delivery result: `success`, `failed`, `no_gateway`.

Example Prometheus Queries

```
# IMAP session rate
rate(vvm_imap_sessions_total[5m])

# Authentication failure ratio
sum(rate(vvm_imap_sessions_total{result="auth_failed"}[5m]))
  / sum(rate(vvm_imap_sessions_total[5m]))

# SMS delivery success ratio
sum(rate(vvm_sms_total{result="success"}[5m]))
  / sum(rate(vvm_sms_total[5m]))
```

Troubleshooting

Handset Cannot Activate VVM

Symptoms: The handset shows "Visual Voicemail unavailable", or activation does not complete.

Possible causes:

- The VVM service is disabled.
- `omnimessage_url` is wrong or the API is unreachable.
- The handset does not send the `Activate` SMS.

Resolution:

1. Confirm `enabled: true` in the VVM configuration.
2. Confirm the OmniMessage API answers `GET /api/messages`.
3. Confirm the CarrierConfig keys are set on the handset. See [Handset Activation](#).
4. Check OmniMessage for an inbound `Activate` message to the VVM number.

STATUS SMS Not Delivered

Symptoms: The handset sends `Activate`, but the handset does not receive the `STATUS` SMS.

Possible causes:

- `sms_gateway` is wrong or the API is unreachable.
- The handset is not registered for SMS-over-IMS.

Resolution:

1. Confirm `sms_gateway` points at `<omnimessage_url>/api/messages_plaintext`.

2. Check OmniMessage for an outbound message from the VVM number to the handset.
3. Confirm the handset is registered for SMS delivery.

IMAP Authentication Fails

Symptoms: The handset receives the `STATUS` SMS, but the IMAP login fails.

Possible causes:

- The password in the cached `STATUS` differs from the account password.
- The account no longer exists, for example after an OmniSEP restart.
- A firewall blocks the IMAP port.

Resolution:

1. Re-provision the handset so it receives a fresh `STATUS` with the current password.
2. Confirm the account exists in storage.
3. Confirm the firewall allows the IMAP port from the UE data network.

Handset Ignores the Voicemail

Symptoms: The handset syncs, but the voicemail does not appear, or the audio does not play.

Possible causes:

- The message `Content-Type` is not `multipart/`.
- The audio part is absent or has the wrong `Content-Type`.

Resolution:

1. Confirm the message is `multipart/mixed` with an `audio/amr` part.
2. Confirm the handset can reach the IMAP server on the configured port.
3. Confirm the `SYNC` SMS reports the correct message id and count.

References

- [OMTP VVM Specification v1.3](#) - Visual Voicemail Interface Specification.
- [GSMA TS.46](#) - Visual Voicemail Interface Specification.
- [RFC 3501](#) - IMAP4rev1 Protocol.
- [RFC 5322](#) - Internet Message Format.

XCAP Supplementary Services (Simservs)

OmniSEP implements ETSI TS 183 023 XCAP (XML Configuration Access Protocol) for managing IMS supplementary services including call forwarding, call barring, and caller ID settings.

Control Panel UI

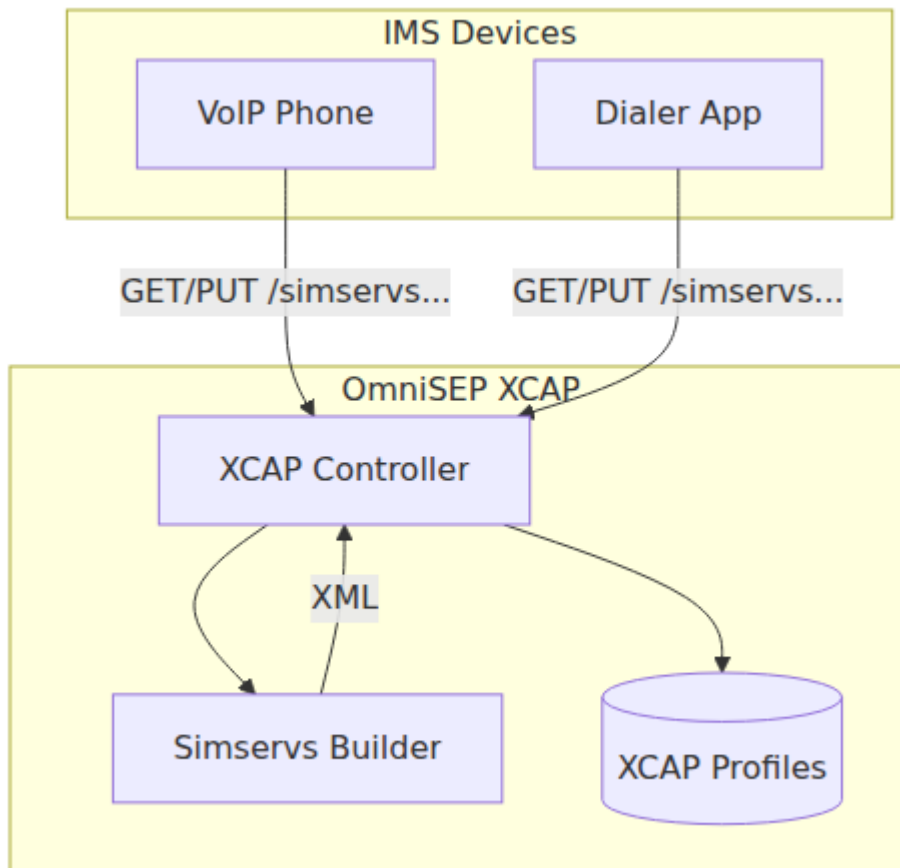
The XCAP Simservs Profiles page provides a web-based interface for viewing and managing subscriber profiles:

Features:

- Search and filter local profiles by MSISDN
- Fetch profiles from HSS using Diameter Sh interface
- View profile details in a 2-column layout:
 - **Left column:** Identity Services (OIP/OIR), Other Services (Call Waiting, Call Hold), Call Forwarding rules

- **Right column:** Incoming and Outgoing Call Barring rules with conditions
- Push/Pull synchronization with HSS
- Edit mode for modifying profile settings

Overview



Simservs Document Structure

The simservs document follows ETSI TS 183 023 structure:



HTTP Interface

XCAP URL Format

```
/simservs.ngn.etsi.org/users/{sip_uri}/simservs.xml[/~/~/{xpath}]
```

Component	Description	Example
{sip_uri}	SIP URI with MSISDN	sip:+15551234567@ims.example.com
{xpath}	XPath selector (optional)	simservs/communication-diversion

Operations

Method	Path
GET	/simservs.ngn.etsi.org/users/{sip}/simservs.xml
PUT	/simservs.ngn.etsi.org/users/{sip}/simservs.xml
GET	/simservs.ngn.etsi.org/users/{sip}/simservs.xml/~/simservs
PUT	/simservs.ngn.etsi.org/users/{sip}/simservs.xml/~/simservs
DELETE	/simservs.ngn.etsi.org/users/{sip}/simservs.xml/~/simservs

Content Types

Content-Type	Usage
application/xcap-el+xml	XCAP element operations
application/xml	Standard XML

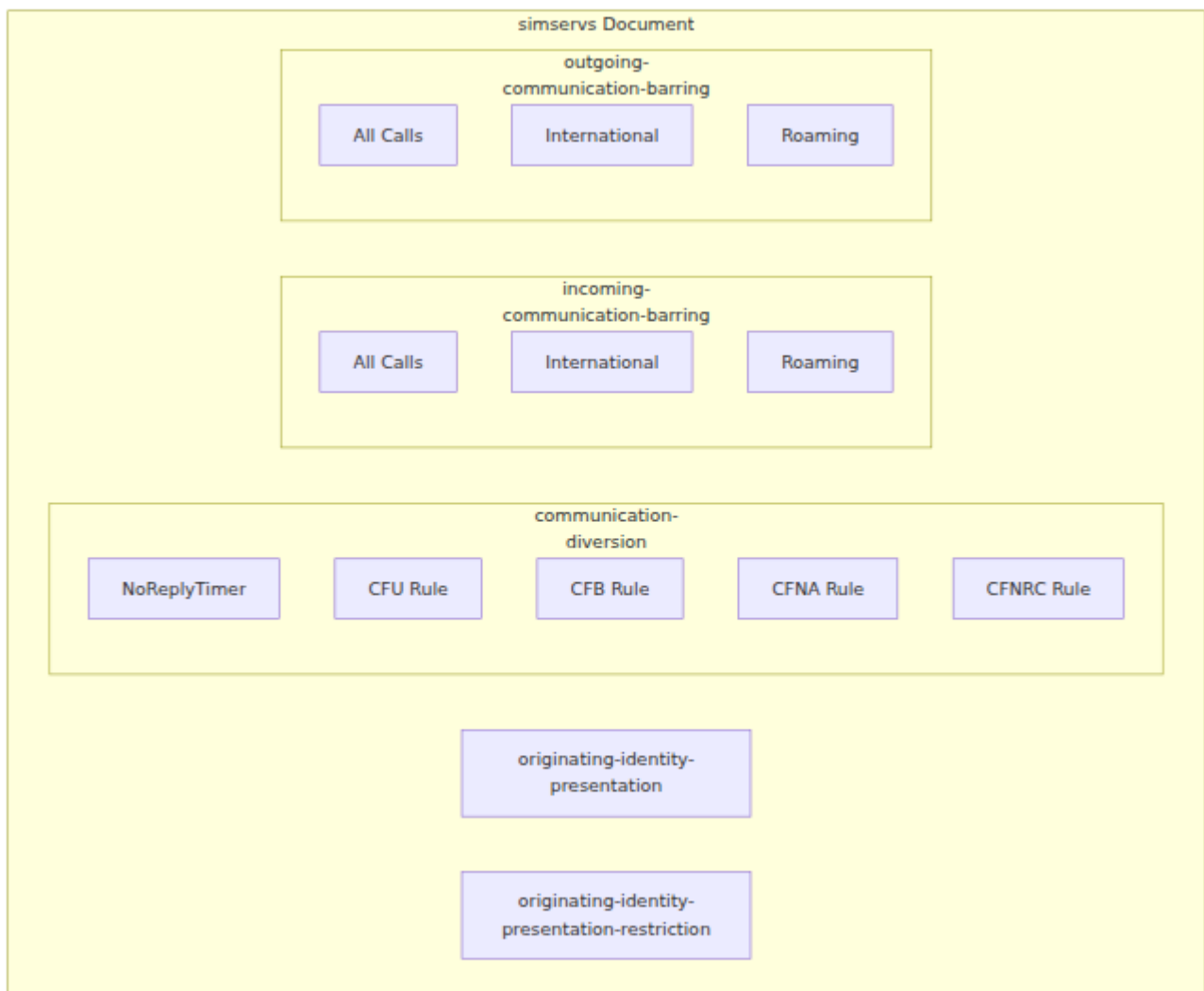
ETag Support

OmniSEP implements [RFC 4825](#) ETag-based concurrency control:

Header	Usage
ETag	Response header with document version
If-Match	Conditional update (PUT/DELETE)
If-None-Match	Conditional GET (304 Not Modified)

Caching Architecture

OmniSEP uses a local-first caching strategy with ETS-based storage and asynchronous HSS synchronization.



Storage Tables

Table	Key	Value	Purpose
xcap_profiles	IMSI	{profile_map, etag}	Primary profile storage
xcap_by_msisdn	MSISDN	IMSI	Lookup index for XCAP requests

Caching Strategy

Read Operations (GET)

1. Query local ETS cache first
2. On cache miss, fetch from HSS via Diameter Sh
3. Seed local cache with HSS response
4. Return default profile if HSS unavailable or no profile exists

Write Operations (PUT)

1. Validate ETag precondition against cached value
2. Update local cache immediately
3. Return response to client with new ETag
4. Asynchronously push changes to HSS (non-blocking)

ETag Generation

ETags are computed as MD5 hashes of the profile data:

```
ETag = MD5(erlang:term_to_binary(profile))[0:16]
```

Example: "a1b2c3d4e5f6g7h8"

The 16-character hex string changes whenever any profile field is modified, enabling precise cache invalidation.

Concurrency Control

Mechanism	Purpose
ETS read_concurrency	Fast parallel reads without locking
GenServer writes	Atomic write operations
ETag validation	Prevents lost updates from concurrent clients

Default Profile

Unknown subscribers automatically receive a default profile per 3GPP TS 24.623:

- All supplementary services active
- OIR default: presentation-not-restricted
- NoReplyTimer: 20 seconds
- Empty call forwarding/barring rulesets

This ensures devices always receive a valid response without requiring pre-provisioning.

HSS Integration

Operation	Diameter Command	Timing
Fetch profile	Sh UDR (User-Data-Request)	Synchronous on cache miss
Push changes	Sh PUR (Profile-Update-Request)	Asynchronous after PUT

The asynchronous push means HSS updates do not block client responses, improving latency for device requests.

Simservs XML Document

Full Document Example

```
<?xml version="1.0" encoding="UTF-8"?>
<simservs xmlns="http://uri.etsi.org/ngn/params/xml/simservs/xcap"
  xmlns:cp="urn:ietf:params:xml:ns:common-policy">

  <originating-identity-presentation active="true"/>

  <originating-identity-presentation-restriction active="true">
    <default-behaviour>presentation-not-restricted</default-
behaviour>
  </originating-identity-presentation-restriction>

  <communication-diversion active="true">
    <NoReplyTimer>20</NoReplyTimer>
    <cp:ruleset>
      <cp:rule id="cfb">
        <cp:conditions>
          <busy/>
        </cp:conditions>
        <cp:actions>
          <forward-to>
            <target>tel:+15557654321</target>
            <notify-caller>false</notify-caller>
          </forward-to>
        </cp:actions>
      </cp:rule>
    </cp:ruleset>
  </communication-diversion>

  <incoming-communication-barring active="false">
    <cp:ruleset/>
  </incoming-communication-barring>

  <outgoing-communication-barring active="false">
    <cp:ruleset/>
  </outgoing-communication-barring>

</simservs>
```

Namespaces

Prefix	Namespace	Descrip
(default)	<code>http://uri.etsi.org/ngn/params/xml/simservs/xcap</code>	ETSI Simservs
cp	<code>urn:ietf:params:xml:ns:common-policy</code>	RFC 474 Commor Policy

Services

Originating Identity Presentation (OIP)

Controls whether caller ID is displayed to called party.

```
<originating-identity-presentation active="true"/>
```

Attribute	Type	Description
<code>active</code>	Boolean	Service enabled/disabled

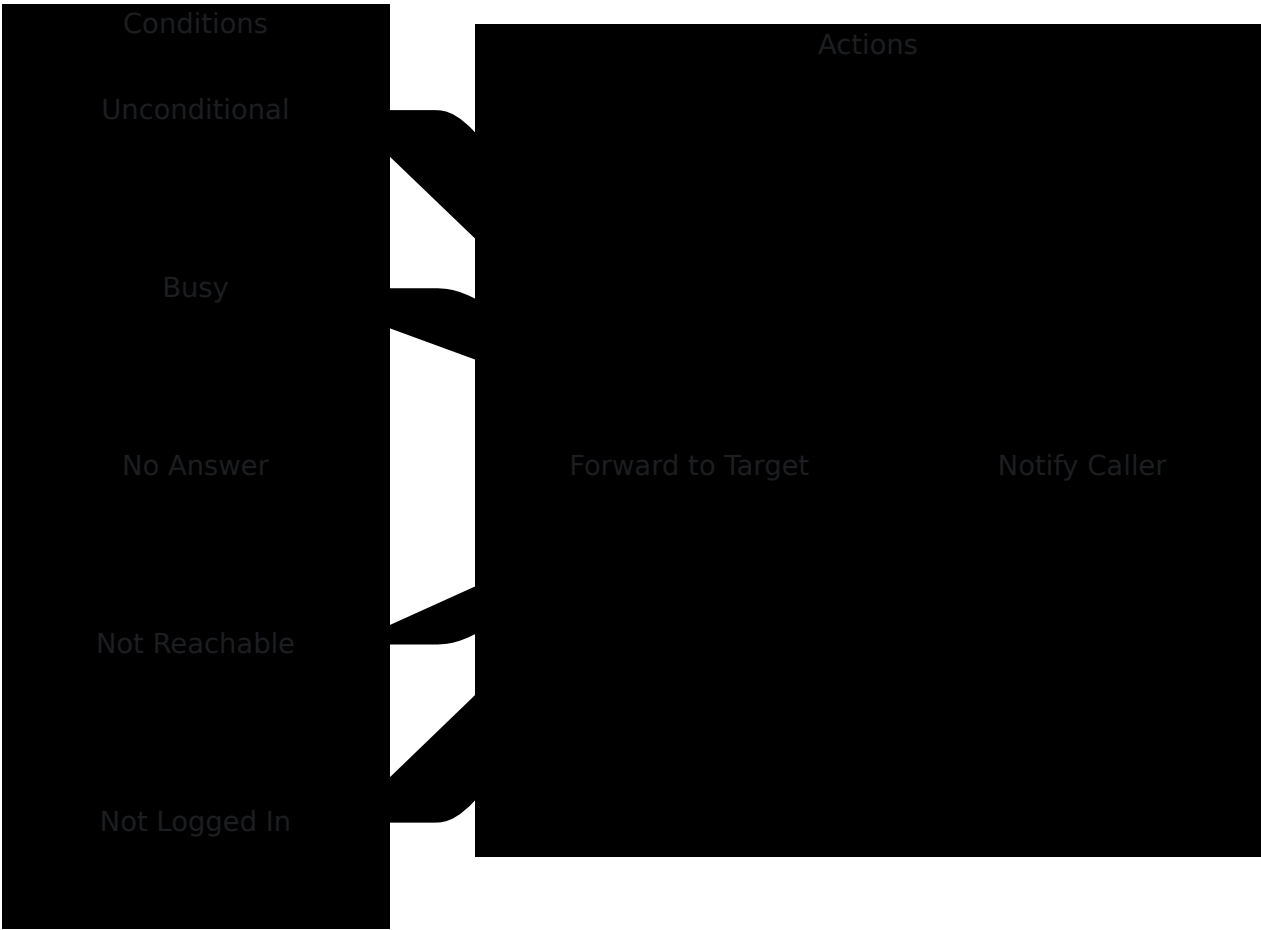
Originating Identity Presentation Restriction (OIR)

Controls caller ID hiding.

```
<originating-identity-presentation-restriction active="true">  
  <default-behaviour>presentation-not-restricted</default-behaviour>  
</originating-identity-presentation-restriction>
```

Element	Values	Description
default-behaviour	presentation-restricted, presentation-not-restricted	Default caller ID behavior

Communication Diversion (Call Forwarding)



Rule Types

Rule ID	Condition	Description
cfu	(none)	Call Forwarding Unconditional - forwards all calls immediately
cfb	busy	Call Forwarding on Busy
cfna	no-answer	Call Forwarding on No Answer (uses NoReplyTimer)
cfnrc	not-reachable	Call Forwarding on Not Reachable
cfnl	not-logged-in	Call Forwarding on Not Logged In

Call Forwarding Rule Structure

```

<communication-diversion active="true">
  <NoReplyTimer>20</NoReplyTimer>
  <cp:ruleset>
    <cp:rule id="cfna">
      <cp:conditions>
        <no-answer/>
      </cp:conditions>
      <cp:actions>
        <forward-to>
          <target>tel:+15557654321</target>
          <notify-caller>false</notify-caller>
        </forward-to>
      </cp:actions>
    </cp:rule>
  </cp:ruleset>
</communication-diversion>

```

Element	Type	Description
NoReplyTimer	Integer	Seconds to wait before CFNA (1-300)
cp:rule/@id	String	Rule identifier (cfu, cfb, cfna, cfnc, cfnl)
target	tel: URI	Forward destination number
notify-caller	Boolean	Play announcement to caller

Target URI Format

Forward targets use tel: URI format per [RFC 3966](#):

```
tel:+15557654321;phone-context=ims.mnc001.mcc310.3gppnetwork.org
```

Component	Description
tel:	URI scheme
+15557654321	E.164 number with country code
phone-context	IMS domain (optional)

Call Barring

Incoming Communication Barring


```
<incoming-communication-barring active="true">
  <cp:ruleset>
    <cp:rule id="international">
      <cp:conditions>
        <international/>
      </cp:conditions>
      <cp:actions>
        <allow>false</allow>
      </cp:actions>
    </cp:rule>
  </cp:ruleset>
</incoming-communication-barring>
```

Outgoing Communication Barring

```
<outgoing-communication-barring active="true">
  <cp:ruleset>
    <cp:rule id="international">
      <cp:conditions>
        <international/>
      </cp:conditions>
      <cp:actions>
        <allow>false</allow>
      </cp:actions>
    </cp:rule>
  </cp:ruleset>
</outgoing-communication-barring>
```

Barring Rule Types

Rule ID	Condition	Description
all	(none)	Bar all calls
international	international	Bar international calls
international-exHC	international-exHC	Bar international except home country
roaming	roaming	Bar calls while roaming

API Examples

Get Full Simservs Document

GET

/simservs.ngn.etsi.org/users/sip:+15551234567@ims.example.com/simservs
Accept: application/xcap-el+xml

Response:

HTTP/1.1 200 OK

Content-Type: application/xcap-el+xml

ETag: "a1b2c3d4e5f6g7h8"

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<simservs
```

```
xmlns="http://uri.etsi.org/ngn/params/xml/simservs/xcap"...>
```

```
...
```

```
</simservs>
```

Get Call Forwarding Settings

GET

/simserve.ngn.etsi.org/users/sip:+15551234567@ims.example.com/simserve/diversion

Enable Call Forwarding on Busy

PUT

/simserve.ngn.etsi.org/users/sip:+15551234567@ims.example.com/simserve/diversion/cp:ruleset/cfb

Content-Type: application/xcap-el+xml

If-Match: "a1b2c3d4e5f6g7h8"

```
<cp:rule id="cfb">
  <cp:conditions>
    <busy/>
  </cp:conditions>
  <cp:actions>
    <forward-to>
      <target>tel:+15557654321</target>
      <notify-caller>false</notify-caller>
    </forward-to>
  </cp:actions>
</cp:rule>
```

Disable Call Forwarding Rule

DELETE

/simserve.ngn.etsi.org/users/sip:+15551234567@ims.example.com/simserve/diversion/cp:ruleset/cfb

If-Match: "a1b2c3d4e5f6g7h8"

Update NoReplyTimer

```
PUT
/simservs.ngn.etsi.org/users/sip:+15551234567@ims.example.com/simserv
diversion/NoReplyTimer
Content-Type: application/xcap-el+xml

25
```

Management API

For administrative access, use the JSON management API:

Get Profile

```
GET /api/xcap/15551234567
```

Response:

```
{
  "oip": { "active": true },
  "oir": {
    "active": true,
    "default_behaviour": "presentation-not-restricted"
  },
  "no_reply_timer": 20,
  "call_forwarding": {
    "cfb": {
      "enabled": true,
      "target": "tel:+15557654321",
      "condition": "busy"
    }
  },
  "call_barring_incoming": {},
  "call_barring_outgoing": {}
}
```

Set Profile

```
POST /api/xcap/15551234567
Content-Type: application/json
```

```
{
  "no_reply_timer": 25,
  "call_forwarding": {
    "cfna": {
      "enabled": true,
      "target": "tel:+15559876543",
      "condition": "no-answer"
    }
  }
}
```

Default Profile

New subscribers receive this default profile:

Setting	Default
OIP	Active
OIR	Active, presentation-not-restricted
NoReplyTimer	20 seconds
Call Forwarding	All rules disabled
Call Barring (Incoming)	All calls allowed
Call Barring (Outgoing)	All calls allowed

Error Responses

HTTP Status	Description
200	Success
304	Not Modified (If-None-Match matched)
400	Bad Request - Invalid XML or path
404	Not Found - Document or element not found
405	Method Not Allowed
409	Conflict - Constraint violation
412	Precondition Failed - ETag mismatch

Error Response Format

```
<?xml version="1.0" encoding="UTF-8"?>
<xcap-error xmlns="urn:ietf:params:xml:ns:xcap-error">
  <error-element>Element not found: simservs/unknown</error-
element>
</xcap-error>
```

Reference Specifications

Specification	Description
ETSI TS 183 023	XCAP framework for NGN SImservs
ETSI TS 183 004	Communication Diversion (CDIV)
RFC 4825	XCAP Protocol
RFC 4745	Common Policy
RFC 3966	tel: URI
3GPP TS 24.623	XCAP over Ut interface