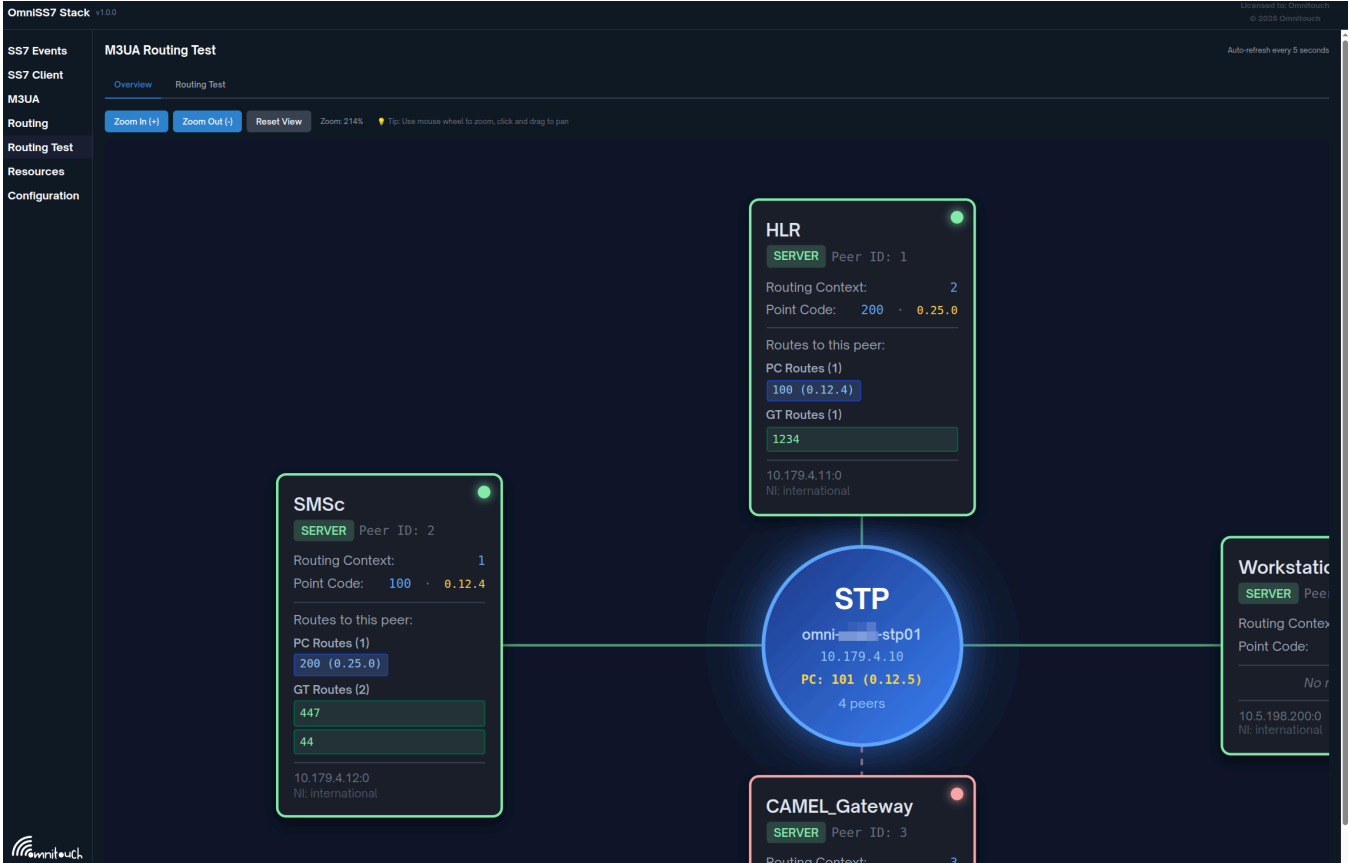




OmniSS7 - Guia do Usuário

OmniSS7 da Omnitouch Network Services é uma pilha de sinalização SS7 abrangente e de uso geral que fornece funcionalidade flexível para elementos de rede.



Visão Geral da Documentação

Esta documentação está organizada por função do elemento de rede. Escolha o guia que corresponde à sua implantação:

◆ Guias de Configuração

- [Guia STP](#) - Configuração do Ponto de Transferência de Sinal
 - Roteie o tráfego SS7 entre pares de rede
 - Roteamento de Código de Ponto e Título Global
 - Balançamento de carga e ocultação de topologia
 - **Use isso se você estiver roteando tráfego SS7 entre redes**
- [Guia do Cliente MAP](#) - Configuração do Cliente MAP
 - Conecte-se como cliente M3UA para enviar solicitações MAP
 - Consultas HLR, autenticação, informações de roteamento
 - Suporte ao protocolo MAP genérico
 - **Use isso se você estiver enviando solicitações MAP para elementos de rede**
- [Guia do Centro de SMS](#) - Configuração do Centro de SMS (SMSc)
 - Roteamento e entrega de mensagens SMS
 - Filas de mensagens com suporte a banco de dados
 - Auto-limpeza e relatórios de entrega
 - **Use isso se você estiver operando um Centro de SMS**
- [Guia HLR](#) - Configuração do Registro de Localização do Assinante
 - Gerenciamento de banco de dados de assinantes
 - Geração de vetores de autenticação
 - Atualizações de localização e informações de roteamento
 - **Use isso se você estiver operando um HLR/HSS**
- [Guia do Gateway CAMEL](#) - Configuração do Gateway CAMEL
 - Serviços de rede inteligente (CAP/CAMEL)
 - Controle de chamadas em tempo real e cobrança
 - Integração OCS para faturamento
 - Construtor de solicitações interativas e monitoramento de sessões
 - **Use isso se você estiver fornecendo serviços IN ou cobrança em tempo real**

◆ Recursos Comuns

- [Guia de Recursos Comuns](#) - Componentes Compartilhados
 - Visão geral e configuração da interface da web
 - Documentação da API
 - Monitoramento e métricas (Prometheus)
 - Melhores práticas e solução de problemas

◆ Documentação de Referência

- [Apêndice](#) - Referência Técnica
 - Especificações do protocolo SS7
 - Códigos de operação MAP
 - Fluxos de transação TCAP
 - Codificações e formatos de caracteres

Início Rápido

1. Visão Geral do Sistema

OmniSS7 pode operar em diferentes modos, dependendo das suas necessidades de rede:

3. Configuração

OmniSS7 pode ser executado em 5 modos operacionais diferentes. O arquivo de configuração `config/runtime.exe` contém exemplos completos e prontos para uso.

Para mudar de modo:

1. Abra `config/runtime.exe`
2. Descomente a seção de configuração desejada (STP, HLR, SMSc ou CAMEL GW)
3. Comente as outras seções
4. Atualize os endereços IP e URLs da API conforme necessário
5. Reinicie o aplicativo

→ Veja os guias específicos de modo abaixo para instruções completas de configuração

Exemplos de configurações em runtime. exs:

Modo STP:

```
config :omniss7,
  map_client_enabled: true,
  hlr_mode_enabled: false,
  smsc_mode_enabled: false,
  camelgw_mode_enabled: false,
  map_client_m3ua: %{...}
```

Modo HLR:

```
config :omniss7,
  map_client_enabled: false,
  hlr_mode_enabled: true,
  smsc_mode_enabled: false,
  camelgw_mode_enabled: false,
  hlr_api_base_url: "...",
  map_client_m3ua: %{...}
```

Modo SMSc:

```
config :omniss7,
  map_client_enabled: true,
  hlr_mode_enabled: false,
  smsc_mode_enabled: true,
  camelgw_mode_enabled: false,
  smsc_api_base_url: "...",
  auto_flush_enabled: true,
  map_client_m3ua: %{...}
```

Modo Gateway CAMEL:

```
config :omniss7,
  cap_client_enabled: true,
  camelgw_mode_enabled: true,
  ocs_enabled: true,
  ocs_url: "http://your-ocs-server/api/charging",
  cap_version: :v2, # Versão CAP: :v1, :v2, :v3 ou :v4
  cap_client_m3ua: %{...}
```

4. Acesso à Interface da Web

Navegue até <http://localhost> (ou seu hostname configurado)

Arquitetura do Sistema

Matriz de Recursos

Recurso	Modo STP	Cliente MAP	Modo SMSc	Modo HLR	Gateway CAMEL
Roteamento de Código de Ponto	🔗	🔗	🔗	🔗	
Roteamento de Título Global	🔗	🔗	🔗	🔗	🔗
Reescrita de SSN	🔗	🔗	🔗	🔗	🔗
Suporte Multi-Peer	🔗	🔗	🔗	🔗	🔗
Solicitações MAP (Enviar)	🔗	🔗	🔗	🔗	🔗
Respostas MAP (Receber)	🔗	🔗	🔗	🔗	🔗
Gerenciamento de Filas de SMS	🔗	🔗	🔗	🔗	🔗
Auto-Limpeza de SMS	🔗	🔗	🔗	🔗	🔗
Banco de Dados de Assinantes	🔗	🔗	🔗	🔗	🔗
Vetores de Autenticação	🔗	🔗	🔗	🔗	🔗
Atualizações de Localização	🔗	🔗	🔗	🔗	🔗
Suporte CAP/CAMEL	🔗	🔗	🔗	🔗	🔗
Cobrança em Tempo Real	🔗	🔗	🔗	🔗	🔗
Controle de Chamadas (Serviços IN)	🔗	🔗	🔗	🔗	🔗
Interface da Web	🔗	🔗	🔗	🔗	🔗
API REST	🔗	🔗	🔗	🔗	🔗
Métricas Prometheus	🔗	🔗	🔗	🔗	🔗

Operações Comuns

Acesso à Interface da Web

- URL: <http://localhost> (ou hostname configurado)
- API Swagger: <http://localhost/swagger>
- Métricas: <http://localhost/metrics>

Monitoramento

```
# Verifique o status do peer M3UA
curl http://localhost/api/m3ua-status
```

```
# Veja as métricas do Prometheus
curl http://localhost/metrics
```

```
# Verifique a saúde do aplicativo
curl http://localhost/api/health
```

Logs

```
# Configure o nível de log em config/runtime.exs
config :logger,
  level: :debug # Opções: :debug, :info, :warning, :error
```

Capacidades Principais

- **Suporte Completo ao Protocolo MAP** - Operações da Fase 2/3 do MAP
- **Suporte ao Protocolo CAP/CAMEL** - CAP v1/v2/v3/v4 para serviços de rede inteligente
- **Sinalização M3UA/CTP** - Transporte SS7 baseado em IP
- **Cobrança em Tempo Real** - Integração OCS para faturamento pré-pago/pós-pago
- **Fila de Mensagens em Tempo Real** - Entrega de SMS com suporte a banco de dados
- **Construtor de Solicitações Interativa** - Interface da Web para testes CAMEL/CAP
- **Monitoramento de Sessões** - Rastreamento de sessões de chamadas CAMEL em tempo real
- **Documentação da API Interativa** - Swagger UI para testes
- **Métricas Prometheus** - Observabilidade completa
- **Configuração Multi-função** - STP, Cliente MAP, SMSc, HLR, Gateway CAMEL

Visão Geral da Pilha de Protocolo

Exemplos de Casos de Uso

Gateway de Rede (STP)

Roteie tráfego SS7 entre diferentes redes móveis

- Conecte redes de operadores
- Gateway SS7 internacional
- Balanceamento de carga entre HLRs
- Tradução de Título Global
- NAT SCTP (Reutilização Inteligente de Título Global)
- → [Guia STP](#)

Centro de SMS (SMSc)

Entregue mensagens SMS a assinantes móveis

- Entrega de MT-SMS
- Originação de MO-SMS
- Roteamento de SMS para casa
- Ocultação de IMSI
- Firewall de SMS
- Gerenciamento de filas de mensagens
- Relatórios de entrega
- Integrado ao OmniMessage para gerenciar todos os SMS MAP
- → [Guia do Centro de SMS](#)

Cliente MAP

Interaja com qualquer elemento de rede via MAP usando uma API RESTful simples

- PRN / SRI / ATI / etc
- Construa suas próprias aplicações SS7/MAP usando APIs RESTful
- Gateways USSD
- Solicitações de vetores de autenticação
- Consultas de IMSI/MSISDN
- Consultas de informações de roteamento
- → [Guia do Cliente MAP](#)

Banco de Dados de Assinantes (HLR)

Gerencie dados de assinantes e autenticação

- Atualizações de localização
- Geração de autenticação
- Provisão de informações de roteamento
- Integra-se totalmente ao OmniHSS
- → [Guia HLR](#)

Plataforma de Rede Inteligente (Gateway CAMEL)

Controle de chamadas em tempo real e cobrança para operadores de telecomunicações

- Cobrança de chamadas pré-pagas/pós-pagas
- Controle de chamadas (conectar, liberar, roteamento)
- Gerenciamento de sessões e geração de CDR
- Construtor de solicitações interativas para testes
- → [Guia do Gateway CAMEL](#)

Suporte e Recursos

Documentação

Guias de Configuração do Core:

- [Guia de Configuração STP](#) - Roteamento do Ponto de Transferência de Sinal
- [Guia de Configuração do Cliente MAP](#) - Cliente do protocolo MAP
- [Guia de Configuração do Centro de SMS](#) - Roteamento e entrega de SMS
- [Guia de Configuração HLR](#) - Banco de dados de assinantes
- [Guia de Configuração do Gateway CAMEL](#) - Rede inteligente e cobrança

Integração & Referência:

- [Guia do Construtor de Solicitações CAMEL](#) - Ferramenta de teste interativa
- [Guia de Recursos Comuns](#) - Componentes compartilhados e Interface da Web
- [Referência Técnica](#) - Especificações de protocolo

Informações de Contato

Produto: OmniSS7
Fabricante: Omnitouch Network Services
Versão da Documentação: 2.0
Última Atualização: 2025

Para suporte técnico, assistência na implementação ou consultas de vendas, entre em contato com a Omnitouch Network Services.

Esta documentação cobre a operação em tempo de execução do OmniSS7 e a funcionalidade do usuário final. Para instalação, desenvolvimento ou configuração avançada, consulte a documentação técnica.



Guia da API REST

[← Voltar para a Documentação Principal](#)

Este guia fornece documentação abrangente para a **API REST** e **Swagger UI** do OmniSS7.

Índice

1. [Visão Geral](#)
2. [Configuração do Servidor HTTP](#)
3. [Swagger UI](#)
4. [Endpoints da API](#)
5. [Autenticação](#)
6. [Formatos de Resposta](#)
7. [Tratamento de Erros](#)
8. [Métricas \(Prometheus\)](#)
9. [Exemplos de Requisições](#)

Visão Geral

O OmniSS7 fornece uma API REST para acesso programático às operações MAP (Mobile Application Part). A API permite que você:

- Envie requisições MAP (SRI, SRI-for-SM, UpdateLocation, etc.)
- Recupere respostas MAP
- Monitore métricas do sistema via Prometheus

Arquitetura da API

Configuração do Servidor HTTP

Detalhes do Servidor

Parâmetro	Valor	Configurável
Protocolo	HTTP	Não
Endereço IP	0.0.0.0 (todas as interfaces)	Apenas via código
Porta	8080	Apenas via código
Transporte	Plug.Cowboy	Não

URL de Acesso: http://[server-ip]:8080

Habilitando/Desabilitando o Servidor HTTP

Controle se o servidor HTTP inicia:

```
config :omniss7,  
  start_http_server: true  # Defina como false para desabilitar
```

Padrão: true (habilitado)

Quando Desabilitado: O servidor HTTP não iniciará, e a API REST/Swagger UI não estará disponível.

Swagger UI

A API inclui um **Swagger UI** para documentação interativa da API e testes.

Acessando o Swagger UI

URL: http://[server-ip]:8080/swagger

Recursos:

- Documentação interativa da API
- Funcionalidade de teste para endpoints
- Esquemas de requisição/resposta
- Exemplos de payloads

Swagger JSON

A especificação OpenAPI está disponível em:

URL: http://[server-ip]:8080/swagger.json

Casos de Uso:

- Importar para Postman ou outros clientes de API
 - Gerar bibliotecas de cliente
 - Automação da documentação da API
-

Endpoints da API

Todos os endpoints de operação MAP seguem o padrão: POST /api/{operation}

Resumo dos Endpoints

Endpoint	Método	Propósito	Timeout
/api/sri	POST	Enviar Informações de Roteamento	10s
/api/sri-for-sm	POST	Enviar Informações de Roteamento para SM	10s
/api/send-auth-info	POST	Enviar Informações de Autenticação	10s
/api/MT-forwardSM	POST	Encaminhar SM Terminado para Móvel	10s
/api/forwardSM	POST	Encaminhar SM	10s
/api/updateLocation	POST	Atualizar Localização	10s
/api/prn	POST	Fornecer Número de Roaming	10s
/metrics	GET	Métricas do Prometheus	N/A
/swagger	GET	Swagger UI	N/A
/swagger.json	GET	Especificação OpenAPI	N/A

Nota: Todas as requisições MAP têm um **timeout de 10 segundos codificado**.

SendRoutingInfo (SRI)

Recupere informações de roteamento para estabelecer uma chamada a um assinante móvel.

Endpoint: POST /api/sri

Corpo da Requisição:

```
{
  "msisdn": "1234567890",
  "gmsc": "5551234567"
}
```

Parâmetros:

Campo	Tipo	Obrigatório	Descrição
msisdn	String	Sim	MSISDN da parte chamada
gmsc	String	Sim	Título Global do MSC Gateway

Resposta (200 OK):

```
{
  "result": {
    "imsi": "001001234567890",
    "msrn": "5551234999",
    "vlr_number": "5551234800",
  }
}
```

```
    ...  
  }  
}
```

Erro (504 Gateway Timeout):

```
{  
  "error": "timeout"  
}
```

Exemplo cURL:

```
curl -X POST http://localhost:8080/api/sri \  
-H "Content-Type: application/json" \  
-d '{  
  "msisdn": "1234567890",  
  "gmsc": "5551234567"  
}'
```

SendRoutingInfoForSM (SRI-for-SM)

Recupere informações de roteamento para entregar um SMS a um assinante móvel.

Endpoint: POST /api/sri-for-sm

Corpo da Requisição:

```
{  
  "msisdn": "1234567890",  
  "service_center": "5551234567"  
}
```

Parâmetros:

Campo	Tipo	Obrigatório	Descrição
msisdn	String	Sim	MSISDN de destino
service_center	String	Sim	Título Global do Centro de Serviço

Resposta (200 OK):

```
{  
  "result": {  
    "imsi": "001001234567890",  
    "msc_number": "5551234800",  
    "location_info": {...},  
    ...  
  }  
}
```

```
}  
}
```

Exemplo cURL:

```
curl -X POST http://localhost:8080/api/sri-for-sm \  
-H "Content-Type: application/json" \  
-d '{  
  "msisdn": "1234567890",  
  "service_center": "5551234567"  
}'
```

SendAuthenticationInfo

Solicite vetores de autenticação para um assinante.

Endpoint: POST /api/send-auth-info

Corpo da Requisição:

```
{  
  "imsi": "001001234567890",  
  "vectors": 3  
}
```

Parâmetros:

Campo	Tipo	Obrigatório	Descrição
imsi	String	Sim	IMSI do assinante
vectors	Integer	Sim	Número de vetores de autenticação a serem gerados

Resposta (200 OK):

```
{  
  "result": {  
    "authentication_sets": [  
      {  
        "rand": "0123456789ABCDEF...",  
        "xres": "...",  
        "ck": "...",  
        "ik": "...",  
        "autn": "..."  
      }  
    ],  
    ...  
  }  
}
```


Exemplo cURL:

```
curl -X POST http://localhost:8080/api/send-auth-info \
-H "Content-Type: application/json" \
-d '{
  "imsi": "001001234567890",
  "vectors": 3
}'
```

MT-ForwardSM

Entregue um SMS Terminado para Móvel a um assinante.

Endpoint: POST /api/MT-forwardSM

Corpo da Requisição:

```
{
  "imsi": "001001234567890",
  "destination_service_centre": "5551234567",
  "originating_service_center": "5551234568",
  "smsPDU": "0001000A8121436587F900001C48656C6C6F20576F726C64"
}
```

Parâmetros:

Campo	Tipo	Obrigatório	Descrição
imsi	String	Sim	IMSI do assinante de destino
destination_service_centre	String	Sim	GT do centro de serviço de destino
originating_service_center	String	Sim	GT do centro de serviço de origem
smsPDU	String	Sim	SMS TPDU em formato hexadecimal

Nota: smsPDU deve ser uma string codificada em hex (maiúscula ou minúscula).

Resposta (200 OK):

```
{
  "result": {
    "delivery_status": "success",
    ...
  }
}
```

Exemplo cURL:

```
curl -X POST http://localhost:8080/api/MT-forwardSM \
-H "Content-Type: application/json" \
-d '{
  "imsi": "001001234567890",
  "destination_service_centre": "5551234567",
  "originating_service_center": "5551234568",
  "smsPDU": "0001000A8121436587F900001C48656C6C6F20576F726C64"
}'
```

ForwardSM

Encaminhe uma mensagem SMS (MO-SMS do assinante).

Endpoint: POST /api/forwardSM

Corpo da Requisição: Mesmo que MT-ForwardSM

Exemplo cURL:

```
curl -X POST http://localhost:8080/api/forwardSM \
-H "Content-Type: application/json" \
-d '{
  "imsi": "001001234567890",
  "destination_service_centre": "5551234567",
  "originating_service_center": "5551234568",
  "smsPDU": "0001000A8121436587F900001C48656C6C6F20576F726C64"
}'
```

UpdateLocation

Notifique o HLR sobre a mudança de localização do assinante (registro VLR).

Endpoint: POST /api/updateLocation

Corpo da Requisição:

```
{
  "imsi": "001001234567890",
  "vlr": "5551234800"
}
```

Parâmetros:

Campo	Tipo	Obrigatório	Descrição
imsi	String	Sim	IMSI do assinante
vlr	String	Sim	Endereço do Título Global do VLR

Resposta (200 OK):

```
{
  "result": {
    "hlr_number": "5551234567",
    "subscriber_data": {...},
    ...
  }
}
```

Nota: No modo HLR, isso aciona a sequência InsertSubscriberData (ISD) com um timeout de 10 segundos por ISD.

Exemplo cURL:

```
curl -X POST http://localhost:8080/api/updateLocation \
-H "Content-Type: application/json" \
-d '{
  "imsi": "001001234567890",
  "vlr": "5551234800"
}'
```

ProvideRoamingNumber (PRN)

Solicite MSRN (Número de Roaming da Estação Móvel) para roteamento de chamadas para assinantes em roaming.

Endpoint: POST /api/prn

Corpo da Requisição:

```
{
  "msisdn": "1234567890",
  "gmsc": "5551234567",
  "msc_number": "5551234800",
  "imsi": "001001234567890"
}
```

Parâmetros:

Campo	Tipo	Obrigatório	Descrição
msisdn	String	Sim	MSISDN do assinante
gmsc	String	Sim	GT do MSC Gateway
msc_number	String	Sim	Número do MSC para o assinante
imsi	String	Sim	IMSI do assinante

Resposta (200 OK):

```
{
  "result": {
    "msrn": "5551234999",
    ...
  }
}
```

Exemplo cURL:

```
curl -X POST http://localhost:8080/api/prn \
-H "Content-Type: application/json" \
-d '{
  "msisdn": "1234567890",
  "gmsc": "5551234567",
  "msc_number": "5551234800",
  "imsi": "001001234567890"
}'
```

Autenticação

Status Atual: A API **não requer autenticação**.

Considerações de Segurança:

- A API é destinada ao uso em rede interna/confiável
- Considere usar regras de firewall para restringir o acesso
- Para implantações em produção, considere implementar middleware de autenticação

Formatos de Resposta

Todas as respostas usam o formato **JSON**.

Resposta de Sucesso

Status HTTP: 200 OK

Estrutura:

```
{
  "result": {
    // Dados de resposta específicos da operação
  }
}
```

Resposta de Erro

Status HTTP:

- 400 Bad Request - Corpo da requisição inválido
- 504 Gateway Timeout - Timeout da requisição MAP (10 segundos)
- 404 Not Found - Endpoint inválido

Estrutura:

```
{  
  "error": "timeout"  
}
```

ou

```
{  
  "error": "invalid request"  
}
```

Tratamento de Erros

Erros Comuns

Erro	Código HTTP	Descrição	Solução
JSON Inválido	400	O corpo da requisição não é um JSON válido	Verifique a sintaxe do JSON
Campos Ausentes	400	Campos obrigatórios ausentes	Inclua todos os parâmetros obrigatórios
Timeout	504	A requisição MAP excedeu o timeout de 10s	Verifique a conectividade M3UA, disponibilidade do HLR/VLR
Não Encontrado	404	Endpoint inválido	Verifique a URL do endpoint

Comportamento de Timeout

Todas as requisições MAP têm um **timeout de 10 segundos codificado**:

1. Requisição enviada ao MapClient GenServer
2. Aguarda resposta por até 10 segundos
3. Se não houver resposta → retorna 504 Gateway Timeout
4. Se a resposta for recebida → retorna 200 OK com resultado

Solução de Problemas de Timeout:

- Verifique o status da conexão M3UA (Web UI → página M3UA)
- Verifique se o elemento de rede (HLR/VLR/MSR) é acessível
- Verifique a configuração de roteamento
- Revise os logs de eventos SS7 em busca de erros

Métricas (Prometheus)

A API expõe métricas do Prometheus para monitoramento.

Endpoint de Métricas

URL: `http://[server-ip]:8080/metrics`

Formato: Formato de texto do Prometheus

Exemplo de Saída:

```
# HELP map_requests_total Total de requisições MAP
# TYPE map_requests_total counter
map_requests_total{operation="sri"} 42
map_requests_total{operation="sri_for_sm"} 158
map_requests_total{operation="updateLocation"} 23

# HELP cap_requests_total Total de requisições CAP
# TYPE cap_requests_total counter
cap_requests_total{operation="initialDP"} 87
cap_requests_total{operation="requestReportBCSMEEvent"} 91

# HELP map_request_duration_milliseconds Duração das requisições/
respostas MAP em ms
# TYPE map_request_duration_milliseconds histogram
map_request_duration_milliseconds_bucket{operation="sri",le="10"} 5
map_request_duration_milliseconds_bucket{operation="sri",le="50"} 12
map_request_duration_milliseconds_bucket{operation="sri",le="100"} 35
...

# HELP map_pending_requests Número de requisições MAP TID pendentes
# TYPE map_pending_requests gauge
map_pending_requests 3
```

Métricas Disponíveis

Métrica	Tipo	Rótulos	Descrição
map_requests_total	Counter	operation	Total de requisições MAP por tipo de operação

Métrica	Tipo	Rótulos	Descrição
cap_requests_total	Counter	operation	Total de requisições CAP por tipo de operação
map_request_duration_milliseconds	Histogram	operation	Duração da requisição em milissegundos
map_pending_requests	Gauge	-	Número de transações MAP pendentes

Configuração do Prometheus

Adicione ao seu `prometheus.yml`:

```
scrape_configs:
  - job_name: 'omniss7'
    static_configs:
      - targets: ['server-ip:8080']
    metrics_path: '/metrics'
    scrape_interval: 15s
```

Exemplos de Requisições

Exemplo em Python

```
import requests
import json

# Requisição SRI-for-SM
url = "http://localhost:8080/api/sri-for-sm"
payload = {
    "msisdn": "1234567890",
    "service_center": "5551234567"
}

response = requests.post(url, json=payload, timeout=15)

if response.status_code == 200:
    result = response.json()
    print(f"Sucesso: {result}")
elif response.status_code == 504:
    print("Timeout - sem resposta da rede")
else:
    print(f"Erro: {response.status_code} - {response.text}")
```

Exemplo em JavaScript

```
const axios = require('axios');

async function sendSRI() {
  try {
    const response = await axios.post('http://localhost:8080/api/sri', {
      msisdn: '1234567890',
      gmsc: '5551234567'
    }, {
      timeout: 15000
    });

    console.log('Sucesso:', response.data);
  } catch (error) {
    if (error.code === 'ECONNABORTED') {
      console.error('Timeout - sem resposta da rede');
    } else {
      console.error('Erro:', error.response?.data || error.message);
    }
  }
}

sendSRI();
```

Exemplo em Bash/cURL

```
#!/bin/bash

# Requisição UpdateLocation
response=$(curl -s -w "\n%{http_code}" -X POST http://localhost:8080/api/updateLocation \
-H "Content-Type: application/json" \
-d '{
  "imsi": "001001234567890",
  "vlr": "5551234800"
}')

http_code=$(echo "$response" | tail -n 1)
body=$(echo "$response" | sed '$d')

if [ "$http_code" -eq 200 ]; then
  echo "Sucesso: $body"
elif [ "$http_code" -eq 504 ]; then
  echo "Timeout - sem resposta da rede"
else
  echo "Erro $http_code: $body"
```


Diagramas de Fluxo

Fluxo de Requisição da API

Resumo

A API REST do OmniSS7 fornece:

- ❖ **Operações MAP** - Suporte total para SRI, SRI-for-SM, UpdateLocation, entrega de SMS, autenticação
- ❖ **Swagger UI** - Documentação interativa da API e testes
- ❖ **Métricas Prometheus** - Monitoramento e observabilidade
- ❖ **Timeouts Codificados** - Timeout de 10 segundos para todas as requisições MAP
- ❖ **Servidor HTTP** - Executa na porta 8080 (configurável via `start_http_server`)

Para acesso à Web UI, consulte o [Guia da Web UI](#).

Para detalhes de configuração, consulte a [Referência de Configuração](#).



Referência Técnica (Apêndice)

[← Voltar à Documentação Principal](#)

Referência técnica para protocolos SS7 e implementação do OmniSS7.

Pilha de Protocolos SS7

Códigos de Operação MAP

Operação	Opcode	Propósito
updateLocation	2	Registrar localização do assinante
cancelLocation	3	Cancelar registro no VLR
provideRoamingNumber	4	Solicitar MSRN
sendRoutingInfo	22	Consultar roteamento de chamadas
mt-forwardSM	44	Entregar SMS ao assinante
sendRoutingInfoForSM	45	Consultar roteamento de SMS
mo-forwardSM	46	Encaminhar SMS do assinante
sendAuthenticationInfo	56	Solicitar vetores de autenticação

Tipos de Mensagens TCAP

- **BEGIN** - Iniciar transação
 - **CONTINUE** - Meio da transação
 - **END** - Resposta final
 - **ABORT** - Cancelar transação
-

Endereçamento SCCP

Formatos de Título Global

- **E.164** - Número de telefone internacional (ex: 447712345678)
- **E.212** - Formato IMSI (ex: 234509876543210)
- **E.214** - Formato de código de ponto

Números de Subsistema (SSN)

- **SSN 6**: HLR

- **SSN 7:** VLR
 - **SSN 8:** MSC/SMSC
 - **SSN 9:** GMLC
 - **SSN 10:** SGSN
-

SMS TPDU

Tipos de Mensagens

- **SMS-DELIVER** (MT) - Rede para móvel
- **SMS-SUBMIT** (MO) - Móvel para rede
- **SMS-STATUS-REPORT** - Status de entrega
- **SMS-COMMAND** - Comando remoto

Codificações de Caracteres

- **GSM7** - Alfabeto GSM de 7 bits (160 caracteres por SMS)
 - **UCS2** - Unicode de 16 bits (70 caracteres por SMS)
 - **8-bit** - Dados binários (140 bytes por SMS)
-

Estados M3UA

- **DOWN** - Sem conexão SCTP
 - **CONNECTING** - SCTP conectando
 - **ASPUP_SENT** - Aguardando ACK de ASPUP
 - **INACTIVE** - ASP ativo, mas não em uso
 - **ASPAC_SENT** - Aguardando ACK de ASPAC
 - **ACTIVE** - Pronto para tráfego
-

Códigos de Ponto Comuns SS7

Os códigos de ponto são tipicamente valores de 14 bits (ITU) ou 24 bits (ANSI).

Formato de Exemplo (ITU):

- Rede: 3 bits
 - Cluster: 8 bits
 - Membro: 3 bits
-

Códigos de Erro SCCP

- **0** - Sem tradução para o endereço
 - **1** - Sem tradução para endereço específico
 - **2** - Congestionamento de subsistema
 - **3** - Falha de subsistema
 - **4** - Usuário não equipado
 - **5** - Falha de MTP
 - **6** - Congestionamento de rede
 - **7** - Não qualificado
 - **8** - Erro no transporte da mensagem
-

Códigos de Erro MAP

Código	Erro	Descrição
1	unknownSubscriber	Assinante não encontrado no HLR
27	absentSubscriber	Assinante não acessível
34	systemFailure	Falha na rede
35	dataMissing	Dados necessários não disponíveis
36	unexpectedDataValue	Valor de parâmetro inválido

Documentação Relacionada

- [← Voltar à Documentação Principal](#)
 - [Guia STP](#)
 - [Guia do Cliente MAP](#)
 - [Guia do Centro de SMS](#)
 - [Guia HLR](#)
 - [Recursos Comuns](#)
-

OmniSS7 por Omnitouch Network Services



Guia de Configuração do Gateway CAMEL

Visão Geral

O modo **Gateway CAMEL (CAMELGW)** transforma o OmniSS7 em uma plataforma de Rede Inteligente (IN) que fornece serviços de controle de chamadas e cobrança em tempo real usando o protocolo CAMEL Application Part (CAP).

OmniSS7 Stack CAMEL GW v1.0.2

Download to OmniStack
© 2025 OmniStack

SS7 Events

SS7 Client

M3UA

CAP Sessions

CAP Requests


Resources


Configuration

CAMEL Sessions

Active: 0

Auto-refresh every 2 seconds

Call ID	State	IMSI	Calling Number	Called Number	Service Key	Duration	Start Time	OTID
<div>No Active CAMEL Sessions</div> Sessions will appear here when calls are initiated								
Total Sessions								
0								
Initiated								
0								
Answered								
0								
<div>About CAMEL Sessions</div> This page displays active CAMEL (Customized Applications for Mobile network Enhanced Logic) sessions managed by the gsmSCF (Service Control Function). <div><div>initiated - Call setup in progress</div><div>answered - Call is active</div><div>terminated - Call ended (cleanup pending)</div></div>								



O que é CAMEL?

CAMEL (Aplicações Personalizadas para Lógica Aprimorada de Redes Móveis) é um conjunto de padrões projetados para funcionar em uma rede central GSM ou rede UMTS. Ele permite que os operadores forneçam serviços que requerem controle em tempo real das chamadas, como:

- **Chamadas pré-pagas** - Verificação de saldo e cobrança em tempo real
- **Serviços de tarifa premium** - Cobrança especial para serviços de valor agregado
- **Controle de roteamento de chamadas** - Roteamento dinâmico de destinos com base em tempo/localização
- **Redes privadas virtuais** - Planos de numeração corporativa
- **Filtragem de chamadas** - Permitir/bloquear chamadas com base em critérios

Versões do Protocolo CAP

O OmniSS7 CAMELGW suporta várias versões do CAP:

Versão	Fase	Recursos
CAP v1	Fase CAMEL 1	Controle básico de chamadas, operações limitadas
CAP v2	Fase CAMEL 2	Operações aprimoradas, suporte a SMS
CAP v3	Fase CAMEL 3	Suporte a GPRS, operações adicionais
CAP v4	Fase CAMEL 4	Recursos avançados, suporte a multimídia

Padrão: CAP v2 (mais amplamente implantado)

Arquitetura

Exemplo de Fluxo de Chamadas

Configuração

Pré-requisitos

- OmniSS7 instalado e em execução
- Conectividade M3UA com MSC/GMSC (gsmSSF)
- Sistema de Cobrança Online (OCS) com endpoint de API (opcional, para cobrança em tempo real)

Habilitar o Modo Gateway CAMEL

Edite config/runtime.exs e configure a seção do Gateway CAMEL:

```
config :omniSS7,  
  # Flags de modo - Habilitar recursos CAP/CAMEL  
  cap_client_enabled: true,  
  camelgw_mode_enabled: true,  
  
  # Desabilitar outros modos  
  map_client_enabled: false,  
  hlr_mode_enabled: false,  
  smsc_mode_enabled: false,  
  
  # Configuração da Versão CAP/CAMEL  
  # Determina qual versão CAP usar para solicitações e diálogos de saída  
  # Opções: :v1, :v2, :v3, :v4  
  cap_version: :v2,  
  
  # Integração OCS (para cobrança em tempo real)  
  ocs_enabled: true,  
  ocs_url: "http://your-ocs-server/api/charging",  
  ocs_timeout: 5000, # milissegundos  
  ocs_auth_token: "your-api-token" # Opcional, se OCS requer autenticação  
  
  # Configuração de Conexão M3UA para CAMEL  
  # Conectar como ASP (Processo do Servidor de Aplicação) para operações CAP  
  cap_client_m3ua: %{  
    mode: "ASP",  
    callback: {CapClient, :handle_payload, []},  
    process_name: :camelgw_client_asp,  
  
    # Endpoint local (sistema CAMELGW)  
    local_ip: {18, 179, 4, 13},  
    local_port: 2905,
```

```
# Endpoint remoto (MSC/GMSC - gsmSSF)
remote_ip: {10, 179, 4, 10},
remote_port: 2905,

# Parâmetros M3UA
routing_context: 1,
network_appearance: 0,
asp_identifier: 13
}
```

Configurar Páginas da Web UI

A Web UI inclui páginas especializadas para operações CAMEL:

```
config :control_panel,
  use_additional_pages: [
    {557.Web.EventsLive, "/events", "Eventos 557"},
    {557.Web.TestClientLive, "/client", "Cliente 557"},
    {557.Web.M3UAStatusLive, "/m3ua", "M3UA"},
    {557.Web.CAMELSessionsLive, "/camel_sessions", "Sessões CAP"},
    {557.Web.CAMELRequestLive, "/camel_request", "Solicitações CAP"}
  ],
  page_order: ["/events", "/client", "/m3ua", "/camel_sessions",
               "/camel_request", "/application", "/configuration"]
```

Operações CAP Suportadas

Operações de Entrada (de gsmSSF → gsmSCF)

Operação	Opcode	Descrição	Manipulador
InitialDP	0	Ponto de Detecção Inicial - notificação de configuração de chamada	handle_initial_dp/1
EventReportBCSM	6	Evento do Modelo de Estado de Chamada básico (resposta, desconexão, etc.)	handle_event_report_bcsn/1
ApplyChargingReport	71	Relatório de cobrança do gsmSSF	handle_apply_charging_report/1
AssistRequestInstructions	16	Solicitação de assistência do gsmSRF	handle_assist_request_instructions/1

Operações de Saída (de gsmSCF → gsmSSF)

Operação	Opcode	Descrição	Gerador
Connect	20	Conectar chamada ao número de destino	CapRequestGenerator.connect_request/2
Continue	31	Continuar o processamento da chamada sem modificação	CapRequestGenerator.continue_request/1
ReleaseCall	22	Liberar/terminar a chamada	CapRequestGenerator.release_call_request/2
RequestReportBCSMEvent	23	Solicitar notificação de eventos de chamada	CapRequestGenerator.request_report_bcsn_event_request/2
ApplyCharging	35	Aplicar cobrança à chamada	CapRequestGenerator.apply_charging_request/3

Recursos da Web UI

Página de Sessões CAMEL

URL: http://localhost/camel_sessions

Monitoramento em tempo real de sessões de chamadas CAMEL ativas:

Recursos:

- **Lista de sessões ao vivo** - Atualiza automaticamente a cada 2 segundos
- **Detalhes da sessão** - OTID, ID da Chamada, Estado, Duração
- **Versão CAP** - Exibe a versão do protocolo (CAP v1/v2/v3/v4) detectada a partir do InitialDP
- **Informações da chamada** - IMSI, Número A, Número B, Chave de Serviço
- **Rastreamento de estado** - Iniciada, Respondida, Terminada
- **Cronômetro de duração** - Exibição em tempo real da duração da chamada

Colunas da Tabela:

- ID da Chamada, Estado, Versão, IMSI, Número Chamador, Número Chamado, Chave de Serviço, Duração, Hora de Início, OTID

Estados da Sessão:

- **◇ Iniciada** - InitialDP recebido, aguardando resposta
- **◇ Respondida** - Chamada respondida, cobrança em andamento
- **◇ Terminada** - Chamada encerrada, CDR gerado

Detecção da Versão CAP: O sistema detecta automaticamente a versão do protocolo CAP a partir da parte do diálogo do InitialDP e a exibe na coluna de Versão. Isso ajuda a identificar qual versão CAP cada MSC está usando.

Construtor de Solicitações CAMEL

URL: http://localhost/camel_request

Ferramenta interativa para construir e enviar solicitações CAP:

Recursos:

- **Seletor de tipo de solicitação** - InitialDP, Connect, ReleaseCall, etc.
- **Campos de formulário dinâmicos** - Adapta-se ao tipo de solicitação selecionado
- **Opções SSCP/M3UA** - Configuração avançada de endereçamento
- **Histórico de solicitações** - Últimas 20 solicitações com status
- **Rastreamento de sessão** - Mantém OTID para solicitações de acompanhamento
- **Feedback em tempo real** - Mensagens de sucesso/erro

Tipos de Solicitações:

- InitialDP** - Iniciar nova sessão de chamada
 - Chave de Serviço (inteiro)
 - Número Chamador (A-party)
 - Número Chamado (B-party)
- Connect** - Roteamento da chamada para o destino
 - Número de Destino
- ReleaseCall** - Terminar a chamada
 - Código de Causa (16=Normal, 17=Ocupado, 31=Não Especificado)
- RequestReportBCSMEvent** - Solicitar notificações de eventos
 - Eventos: oAnswer, oDisconnect, tAnswer, tDisconnect
- Continue** - Continuar a chamada sem modificação
 - Nenhum parâmetro necessário
- ApplyCharging** - Aplicar limites de duração da chamada
 - Duração (segundos, 1-864000)
 - Liberar ao Expirar (booleano)
 - Veja [Guia do Construtor de Solicitações CAMEL](#) para uso detalhado

Opções SSCP Avançadas:

- Título Global da Parte Chamadora
- Título Global da Parte Chamadora
- SSN Chamado (padrão: 146 = gsmSSF)
- SSN Chamador (padrão: 146)

Opções M3UA:

- OPC (Código de Ponto de Origem, padrão: 5013)
- DPC (Código de Ponto de Destino, padrão: 5011)

Integração com OCS

Ciclo de Vida da Chamada com Cobrança

1. Iniciação da Chamada (InitialDP)

Quando o MSC envia o InitialDP, o CAMELGV:

1. **Detecta a versão CAP** - Examina a parte do diálogo para identificar CAP v1/v2/v3/v4
2. **Decodifica a mensagem CAP** - Extraí IMSI, números chamadores/chamados
3. **Chama OCS** - API InitiateSession
4. **Recebe autorização** - MaxUsage (por exemplo, 30 segundos)
5. **Armazena a sessão** - No SessionStore (tabela ETS) com a versão CAP
6. **Responde ao MSC** - RequestReportBCSMEvent + Continue (usando a mesma versão CAP)

Exemplo:

```
# Dados decodificados do InitialDP
%{
  imsi: "310150123456789",
  calling_party_number: "14155551234",
  called_party_number: "14155556789",
  service_key: 1,
  msc_address: "19216800123",
  cap_version: :v2 # Detectado a partir do diálogo
}
```

```
# Resposta do OCS
{:ok, %{max_usage: 30}} # 30 segundos autorizados
```

```
# Entrada do SessionStore
%(
  call_id: "CAMEL-48080173",
  initial_dp_data: %(.),
  cap_version: v2, # Armazenado para geração de resposta
  start_time: 1730246400,
  state: :initiated
)
```

2. Resposta da Chamada (EventReportBCSM - oAnswer)

Quando a chamada é respondida:

1. **Recebe o evento oAnswer** - Do MSC
2. **Atualiza OCS** - UpdateSession com usage=0
3. **Inicia o loop de débito** - OCS começa a cobrar
4. **Atualiza o estado da sessão** - :answered no SessionStore
5. **Continua a chamada** - Envia Continue para o MSC

3. Atualizações Periódicas (Opcional)

Para chamadas longas, solicitar crédito adicional:

```
# A cada 30 segundos
OCS.Client.update_session(call_id, %{}), current_usage)
```

Se MaxUsage retornar 0, o assinante não tem crédito → Enviar ReleaseCall

4. Término da Chamada (EventReportBCSM - oDisconnect)

Quando a chamada termina:

1. **Recebe o evento oDisconnect** - Do MSC
2. **Calcula a duração total** - A partir do horário de início da sessão
3. **Termina a sessão OCS** - API TerminateSession
4. **CDR gerado** - Pelo OCS com custo final
5. **Limpa a sessão** - Remove do SessionStore
6. **Envia ReleaseCall** - Confirma o término para o MSC

Análise de CDR

Os CDRs são gerados pelo seu OCS e normalmente incluem:

Campos CDR do CAMEL:

- Account - IMSI ou número chamador
- Destination - Número da parte chamada
- OriginID - Identificador único da chamada (CAMEL-OTID)
- Usage - Duração total da chamada (segundos)
- Cost - Custo calculado
- IMSI - IMSI do assinante
- CallingPartyNumber - A-party
- CalledPartyNumber - B-party
- MSCAddress - Código de ponto MSC que atende
- ServiceKey - Chave de serviço CAMEL

Testes

Teste Manual com o Construtor de Solicitações

1. Navegue até o Construtor de Solicitações:

http://localhost/camel_request

2. Envie o InitialDP:

- Selecione "InitialDP" no dropdown
- Chave de Serviço: 100
- Número Chamador: 14155551234
- Número Chamado: 14155556789
- Clique em "Enviar Solicitação InitialDP"
- Anote o OTID gerado

3. Monitore a Sessão:

- Abra uma nova aba: http://localhost/camel_sessions
- Veja a sessão ativa com estado "Iniciada"

4. Simule a Resposta da Chamada:

- Retorne ao Construtor de Solicitações
- Selecione "EventReportBCSM"
- Tipo de Evento: oAnswer
- Clique em "Enviar Solicitação EventReportBCSM"
- O estado da sessão muda para "Respondida"

5. Encerrar Chamada:

- Selecione "ReleaseCall"
- Código de Causa: 16 (Normal)
- Clique em "Enviar Solicitação ReleaseCall"
- O estado da sessão muda para "Terminada"

Testando com MSC Real

Configure o Serviço CAMEL do MSC

No seu MSC/VLR, configure o serviço CAMEL:

```
# Exemplo de configuração do MSC Huawei
ADD CAMELSERVICE:
  SERVICEID=1,
  SERVICEKEY=100,
  GSMSCFADDR="55512341234", # Título Global do CAMEL GW
  DEFAULTCALLHANDLING=CONTINUE;
```

```
ADD CAMELSUBSCRIBER:
  IMSI="310150123456789",
  SERVICEID=1,
  TRIGGERTYPE=TERMCALL;
```

Monitorar Logs

Assista aos logs do CAMEL GW para mensagens CAP recebidas:

```
# Ver logs em tempo real
tail -f /var/log/omni77/omni77.log

# Filtrar eventos CAP
grep "CAP:" /var/log/omni77/omni77.log

# Ver log de eventos (formatado em JSON)
curl http://localhost/api/events | jq '.[] | select(.map_event | startswith("CAP:"))'
```

Teste de Carga

Use o Construtor de Solicitações em um loop para teste de carga:

```
# Enviar 100 solicitações InitialDP
for i in {1..100}; do
  curl -X POST http://localhost/api/camel/initial_dp \
    -H "Content-Type: application/json" \
    -d '{
      "service_key": 100,
      "calling_number": "141555551234",
      "called_number": "141555556789"
    }'
  sleep 0.1
done
```

Monitoramento & Operações

Métricas do Prometheus

O CAMEL GW expõe métricas em http://localhost:8080/metrics:

Métricas específicas do CAP:

- cap_requests_total(operation) - Total de solicitações CAP por tipo de operação (por exemplo, initialDP requestReportBCSMEvent)

Métricas adicionais de MAP/API:

- map_requests_total(operation) - Total de solicitações MAP por tipo de operação
- map_request_duration_milliseconds(operation) - Histograma de duração das solicitações
- map_pending_requests - Número de transações MAP pendentes

Métricas STP M3UA (se o modo STP estiver habilitado):

- m3ua_stp_messages_received_total(peer_name,point_code) - Mensagens recebidas do pares
- m3ua_stp_messages_sent_total(peer_name,point_code) - Mensagens enviadas para pares
- m3ua_stp_routing_failures_total(reason) - Falhas de roteamento por motivo

Consultas de exemplo:

```
# Solicitações CAP
curl http://localhost:8080/metrics | grep cap_requests_total

# Total de InitialDP recebidos
curl http://localhost:8080/metrics | grep 'cap_requests_total{operation="initialDP"}'

# Solicitações MAP pendentes
curl http://localhost:8080/metrics | grep map_pending_requests
```

Verificações de Saúde

```
# Verificar conectividade M3UA
curl http://localhost/api/m3ua-status

# Verificar conectividade OCS
curl http://localhost/api/ocs-status

# Verificar sessões ativas
curl http://localhost/api/camel/sessions/count
```

Configuração de Logs

```
Ajuste o nível de log em config/runtime.exs:

config :logger,
  level: :info, # Opções: :debug, :info, :warning, :error

# Habilitar logging de depuração CAP
config :logger, :console,
  metadata: [:cap_operation, :otid, :call_id]
```

Resolução de Problemas

Problema: Nenhuma mensagem CAP recebida

Sintomas: O Construtor de Solicitações funciona, mas o MSC não envia InitialDP

Verifique:

- 1. Status do link M3UA: curl http://localhost/api/m3ua-status
- 2. Configuração do serviço CAMEL do MSC (Chave de Serviço, endereço gsmSCF)
- 3. Roteamento SSCP (Título Global deve roteá-lo para o CAMELGW)
- 4. Regras de firewall (permitir porta SCTP 2905)

Solução:

```
# Verificar conectividade M3UA
tcpdump -i eth0 sctp

# Verificar se o MSC pode alcançar o CAMELGW
ss -tuln | grep 2905
```

Problema: Erros OCS

Sintomas: INSUFFICIENT_CREDIT ou erros de timeout

Verifique:

- 1. OCS é acessível: curl http://your-ocs-server/api/health
- 2. A conta tem saldo no OCS
- 3. Plano de avaliação configurado no OCS
- 4. Conectividade de rede com o OCS
- 5. O token de autenticação é válido (se necessário)

Solução:

- Verifique a configuração da URL do OCS em runtime.exs
- Verifique os logs do OCS para erros
- Teste a API do OCS manualmente com curl
- Verifique se as regras de firewall permitem conectividade

Problema: Sessão não encontrada

Sintomas: EventReportBCSM falha com "Sessão não encontrada"

Causa: Desvio de OTID ou sessão expirada

Solução:

- 1. Verifique o OTID nos logs
- 2. Verifique o tempo limite da sessão (padrão: sem expiração)
- 3. Certifique-se de que o DTID corresponda ao OTID nas mensagens Continue/End

```
# Verificar sessões ativas
iex> CAMELGW.SessionStore.list_sessions()
```

Problema: Erros de decodificação

Sintomas: Falha ao decodificar InitialDP nos logs

Causa: Desvio de versão CAP ou mensagem malformada

Solução:

- 1. Verifique se a configuração da versão CAP corresponde ao MSC
- 2. Verifique se a codificação ASN.1 está correta
- 3. Capture PCAP e analise com Wireshark

```
# Capturar mensagens CAP
tcpdump -i eth0 -w cap_trace.pcap sctp port 2905

# Analisar com Wireshark (filtro: m3ua)
wireshark cap_trace.pcap
```

Configuração Avançada

Múltiplas Versões CAP

Suporte a diferentes versões CAP por chave de serviço:

```
config :omniess7,
  cap_version_map: %{
    100 => :v2, # Chave de Serviço 100 usa CAP v2
    200 => :v3, # Chave de Serviço 200 usa CAP v3
    300 => :v4, # Chave de Serviço 300 usa CAP v4
  },
  cap_version: :v2 # Padrão
```

Resumo

O modo Gateway CAMEL permite que o OmniSS7 funcione como uma plataforma completa de Rede Inteligente com:

- ◊ **Suporte total ao protocolo CAP** (v1/v2/v3/v4)
- ◊ **Cobrança em tempo real** via integração OCS
- ◊ **Operações de controle de chamadas** (Connect, Release, Continue)
- ◊ **Gerenciamento de sessões** com armazenamento ETS
- ◊ **Teste interativo** via Construtor de Solicitações da Web UI
- ◊ **Monitoramento ao vivo** de sessões de chamadas ativas
- ◊ **Geração de CDR** para faturamento e análises
- ◊ **Desempenho e confiabilidade prontos para produção**

Para informações adicionais:

- [Documentação do Construtor de Solicitações CAMEL](#)
- [Referência Técnica - Operações CAP](#)

Produto: OmniSS7 CAMEL Gateway

Versão da Documentação: 1.0

Última Atualização: 2025-10-26

CAMEL Request Builder - Resumo da Implementação

Visão Geral

Um novo componente LiveView foi criado para construir e enviar solicitações CAMEL/CAP para fins de teste. Isso fornece uma interface interativa para criar InitialDP e outras operações CAMEL.

OmniSS7 Stack CAMEL GW v1.0.2

Download to GitHub
© 2025 Omnitouch

SS7 Events

SS7 Client

M3UA

CAP Sessions

CAP Requests

Resources

Configuration

CAMEL Request Builder

Build and send CAP/CAMEL requests for testing

ApplyCharging request sent successfully! OTID: DC6EBC13 (CAP v2)

Request Parameters

Request Type

InitialDP (Initial Detection Point)

CAP/CAMEL Version (Current: V2)

CAP v2 (OID: 0.4.0.0.1.0.50.1) - Default

Determines the application context OID in TCAP dialogue for outgoing requests. This selection persists across form changes.

Call Duration (seconds)

60

Maximum call duration (1-864000 seconds, max 10 days)

☒ Release call when duration expires

If unchecked, call continues but ApplyChargingReport is sent.

Show Advanced SCCP/M3UA Options

Send ApplyCharging Request

Current Session

OTID: DC6EBC13

Last Request Size: 38 bytes

Request History (Last 20)

Timestamp	Request Type	CAP Ver	OTID	Status	Size
23:56:25	ApplyCharging	v2	DC6EBC13...	sent	38 bytes
23:56:21	InitialDP	v2	DC6EBC13...	sent	74 bytes

CAMEL Request Types

InitialDP - Initial Detection Point: Notifies gsmSCF about call setup

Connect - Connect call to specified destination

ReleaseCall - Release/terminate an ongoing call

RequestReportBCSMEEvent - Request notifications for call state events

ApplyCharging - Set maximum call duration and charging parameters

Continue - Continue call processing

Novos Componentes

1. CAMEL Request Builder LiveView

Recursos:

- Interface interativa baseada em formulário para construir solicitações CAMEL
- Suporte para múltiplos tipos de solicitação:
 - InitialDP** - Ponto de Detecção Inicial (notificação de configuração de chamada)
 - Connect** - Conectar chamada ao destino
 - ReleaseCall** - Liberar/terminar chamada
 - RequestReportBCSMEEvent** - Solicitar notificações de eventos
 - Continue** - Continuar o processamento da chamada
 - ApplyCharging** - Aplicar limites de cobrança/duração às chamadas

Principais Capacidades:

- Dropdown de seleção de tipo de solicitação
- Campos de formulário dinâmicos com base no tipo de solicitação selecionado
- Opções avançadas de SCCP/M3UA (seção colapsável)
 - Títulos Globais da Parte Chamadora/Chamado
 - Configuração de SSN (Número de Subsistema)
 - Configurações de OPC/DPC (Código de Ponto)
- Histórico de solicitações em tempo real (últimas 20 solicitações)
- Rastreamento de sessão via OTID
- Feedback de sucesso/erro
- Rastreamento do tamanho da solicitação

Rota: /camel_request

2. EventLog Aprimorado com Suporte a CAMEL

Novas Funções:

- pklog_camel/2 - Registro dedicado de mensagens CAMEL/CAP
- lookup_cap_opcode_name/1 - Consulta de código de operação CAP
- find_cap_opcode/1 - Extrair opcode CAP de JSON
- extract_cap_tids/1 - Extrair OTID/DTID de mensagens CAP
- format_cap_to_json/1 - Converter PDUs CAP para formato JSON

Códigos de Operação CAP Suportados:

```
0 => "initialDP"
5 => "connect"
6 => "releaseCall"
7 => "requestReportBCSMEEvent"
8 => "eventReportBCSM"
10 => "continue"
13 => "furnishChargingInformation"
35 => "applyCharging"
... (47 operações no total)
```

Recursos:

- Registro em JSON de todas as solicitações/respostas CAMEL
- Detecção automática de ação TCAP (Iniciar/Continuar/Finalizar/Abortar)
- Extração de endereçamento SCCP
- Tratamento de erros para mensagens malformadas
- Processamento de tarefas em segundo plano (não bloqueante)
- Evento prefixado com "CAP:" para fácil filtragem

3. CapClient Atualizado

Mudanças:

- Adicionados chamados pklog_camel/2 para mensagens de entrada e saída
- Registro duplo: Tanto MAP (pklog) quanto CAP (pklog_camel) para compatibilidade
- Mensagens de saída registradas em sccp_m3ua_maker/2
- Mensagens de entrada registradas em handle_payload/1

Configuração

As novas páginas LiveView foram adicionadas à configuração de tempo de execução:

Arquivo: config/runtime.exs

```
config :control_panel,
  use_additional_pages: [
    {S57.Web.EventsLive, "/events", "Eventos S57"},
    {S57.Web.TestClientLive, "/client", "Cliente S57"},
    {S57.Web.M3UAStatusLive, "/m3ua", "M3UA"},
    {S57.Web.HlrLinksLive, "/hlr_links", "Links HLR"},
    {S57.Web.CamelSessionLive, "/camel_sessions", "Sessões CAMEL"},
    {S57.Web.CamelRequestLive, "/camel_request", "Construtor de Solicitações CAMEL"}
  ],
  page_order: ["/events", "/client", "/m3ua", "/hlr_links",
    "/camel_sessions", "/camel_request",
    "/application", "/configuration"]
```

Uso

Acessando o Construtor de Solicitações

- Navegue até: https://your-server:8087/camel_request
- Selecione o tipo de solicitação no dropdown
- Preencha os parâmetros obrigatórios
- Opcionalmente, expanda "Opções Avançadas de SCCP/M3UA" para ajustes finos
- Clique em "Enviar Solicitação [RequestType]"

Fluxo de Solicitação

InitialDP (Nova Chamada)

- Defina a Chave de Serviço (por exemplo, 100)
- Defina o Número Chamador (A-Party)
- Defina o Número Chamado (B-Party)
- Envie a solicitação → Gera novo OTID
- OTID armazenado na sessão para solicitações de acompanhamento

Solicitações de Acompanhamento (Connect, ReleaseCall, etc.)

- Deve ter OTID ativo da InitialDP
- A solicitação usa automaticamente o OTID armazenado
- Aviso exibido se não houver OTID ativo

Parâmetros da Solicitação

InitialDP:

- Chave de Serviço (inteiro)
- Número Chamador (formato ISDN)
- Número Chamado (formato ISDN)

Connect:

- Número de Destino (para onde rotear a chamada)

ReleaseCall:

- Código de Causa (16 = Normal, 17 = Ocupado, 31 = Não Especificado)

RequestReportBCSMEvent:

- Eventos BCSM (separados por vírgula: oAnswer, oDisconnect, etc.)

Continue:

- Sem parâmetros (usa OTID ativo)

ApplyCharging:

- Duração (segundos, 1-864000) - Duração máxima da chamada antes da ação
- Liberar em Timeout (booleano) - Se deve liberar a chamada quando a duração expira

Opções Avançadas

Endereçamento SCCP:

- GT da Parte Chamadora (Título Global)
- GT da Parte Chamado
- SSN Chamado (padrão 146 = gsmSSF)
- SSN Chamador (padrão 146)

Códigos de Ponto M3UA:

- OPC (Código de Ponto de Origem, padrão 5013)
- DPC (Código de Ponto de Destino, padrão 5011)

Registro em JSON

Todas as mensagens CAMEL agora são registradas em formato JSON no log de eventos com:

- Direção:** entrada/saída
- Ação TCAP:** Iniciar/Continuar/Finalizar/Abortar
- Operação CAP:** e.g. "CAP:initialDP", "CAP:connect"
- Endereçamento SCCP:** Informações da Parte Chamadora/Chamado
- TTID:** OTID/OTID para correlação
- Mensagem Completa:** PDU CAP codificada em JSON

Exemplo de Entrada de Log

```
{
  "map_event": "CAP:initialDP",
  "direction": "outgoing",
  "tcap_action": "Begin",
  "otid": "A1B2C3D4",
  "sccp_called": {
    "SSN": 146,
    "GlobalTitle": {
      "Digits": "55512341234",
      "NumberingPlan": "isdn_tele",
      "NatureOfAddress_Indicator": "International"
    }
  },
  "event_message": "{ ... full CAP PDU ... }"
}
```

Histórico de Solicitações

A interface exibe as últimas 20 solicitações com:

- Timestamp
- Tipo de solicitação (com badge colorido)
- OTID (primeiros 8 caracteres hexadecimais)
- Status (enviado/erro)
- Tamanho da mensagem em bytes

Rastreamento de Sessão

Painel de Informações da Sessão Atual:

- Exibe OTID ativo
- Mostra o tamanho da última solicitação em bytes
- Visível apenas quando a sessão está ativa

Fluxo de Trabalho de Teste

1. Iniciar Nova Chamada:

- Enviar InitialDP → Obter OTID
- O sistema cria a sessão

2. Controlar Chamada:

- Enviar RequestReportBCSMEvent → Solicitar notificações
- Enviar ApplyCharging → Definir limite de duração da chamada (por exemplo, 290 segundos)
- Enviar Connect → Roteamento para o destino
- OU Enviar ReleaseCall → Terminar

3. Ver Resultados:

- Verificar histórico de solicitações
- Monitorar página de Sessões CAMEL
- Revisar logs de eventos com prefixo "CAP:"

ApplyCharging - Controle de Duração da Chamada

Visão Geral

A operação ApplyCharging permite definir uma duração máxima de chamada e, opcionalmente, liberar a chamada quando essa duração expira. Isso é tipicamente usado para cenários de cobrança pré-paga ou para impor limites de tempo em chamadas.

Casos de Uso

- Cobrança Pré-Paga:** Limitar a duração da chamada com base no saldo do assinante
- Cobrança Baseada em Tempo:** Impor intervalos de cobrança periódicos

- **Gerenciamento de Recursos:** Impedir que chamadas sejam executadas indefinidamente
- **Integração OCS:** Coordenar com Sistemas de Cobrança Online para controle de crédito em tempo real

Parâmetros

Duração (maxCallPeriodDuration)

- **Tipo:** Inteiro (1-864000 segundos)
- **Descrição:** Número máximo de segundos que a chamada pode durar antes que o temporizador expire
- **Exemplos:**
 - 60 = 1 minuto
 - 290 = 4 minutos e 50 segundos (valor de teste comum)
 - 3600 = 1 hora
 - 86400 = 24 horas

Liberar em Timeout (releaseIfDurationExceeded)

- **Tipo:** Booleano (true/false)
- **Padrão:** true
- **Descrição:** O que acontece quando a duração expira:
 - true: Liberar/desconectar automaticamente a chamada
 - false: Enviar notificação, mas manter a chamada ativa (permite que gsmSCF tome uma ação)

Estrutura da Mensagem

A mensagem ApplyCharging é codificada como um Continue TCAP com:

- **TCAP:** Mensagem Continue (usa transação existente)
- **Opcode:** 35 (applyCharging)
- **Parâmetros:** ApplyChargingArg contendo:
 - aChargingCharacteristics: Informações de cobrança baseadas em tempo
 - TimeDurationCharging: Duração máxima e sinalizador de liberação
 - partyToCharge: Qual parte é cobrada (padrão: sendingSideID)

Exemplo de Uso

Cenário: Chamada pré-paga com limite de 5 minutos

1. Enviar **InitialDP** para iniciar o monitoramento da chamada

```
Chave de Serviço: 100
Chamador: 447700900123
Chamado: 447700900456
→ OTID: A1B2C3D4
```

2. Enviar **ApplyCharging** para definir o limite de 5 minutos

```
Duração: 300 (segundos)
Liberar em Timeout: true
→ Usa OTID: A1B2C3D4
```

3. Enviar **Connect** para completar a chamada

```
Destino: 447700900456
→ Usa OTID: A1B2C3D4
```

4. Após 5 minutos (300 segundos):
 - Chamada liberada automaticamente pela rede
 - gsmSCF recebe notificação de desconexão

Melhores Práticas

1. **Sempre envie ApplyCharging ANTES de Connect**

- Garante que a cobrança esteja ativa quando a chamada se conecta
- Impede segmentos de chamadas não cobrados

2. **Use com RequestReportBCSMEvent**

- Solicite eventos oAnswer e oDisconnect
- Permite rastrear a duração real da chamada
- Habilita a reavaliação da cobrança, se necessário

3. **Defina durações razoáveis**

- Muito curto: Operações de cobrança frequentes, má experiência do usuário
- Muito longo: Risco de perda de receita em chamadas pré-pagas
- Típico: 60-300 segundos para pré-pago, mais longo para pós-pago

4. **Trate o timeout de forma elegante**

- Se release=false, esteja preparado para lidar com notificações de expiração do temporizador
- Implemente lógica para estender a duração ou liberar a chamada

Tratamento de Erros

Problemas comuns:

- **Nenhum OTID ativo:** Deve enviar InitialDP primeiro
- **Duração inválida:** Deve ser de 1-864000 segundos
- **Suporte da rede:** Algumas implementações de SSF podem não suportar ApplyCharging
- **Precisão do temporizador:** A resolução do temporizador da rede é tipicamente de 1 segundo, mas pode variar

Monitoramento

Rastreie operações ApplyCharging via:

- **Histórico de Solicitações:** Mostra solicitações ApplyCharging enviadas
- **Log de Eventos:** Pesquise por "CAP:applyCharging"
- **Sessões CAMEL:** Monitore sessões ativas com cobrança aplicada
- **Rastreamento TCAP:** Depure problemas de codificação/decodificação

Detalhes da Implementação

Gerenciamento de Estado

- LiveView atribui estado do formulário de rastreamento
- OTID armazenado no socket atribui
- Histórico de solicitações limitado a 20 entradas
- Atualização automática desativada (envio manual apenas)

Geração de Solicitações

- Usa o módulo existente CapRequestGenerator
- Constrói as estruturas TCAP/CAP adequadas
- Codifica com o codec: TCAPMessages
- Envolva em SCCP via CapClient.sccp_m3ua_maker/2

Mecanismo de Envio

- Envia via M3UA para :camelgw_client_asp
- Usa contexto de roteamento 1
- Encapsulamento automático SCCP/M3UA

Tratamento de Erros

- Validação de formulário com feedback ao usuário
- Tratamento elegante de OTID ausente
- Erros de análise exibidos na interface
- Falhas de codificação registradas

Melhorias Futuras

Adições potenciais:

1. Modelos/presets de solicitações
2. Correlação e exibição de respostas
3. Visualização do fluxo de chamadas
4. Detalhamento de sessões
5. Exportar histórico de solicitações
6. Teste de carga (solicitações em massa)
7. Exportação PCAP de mensagens geradas
8. Validação de parâmetros CAP

Notas de Integração

- Compatível com registro MAP existente (paklog)
- Compartilha banco de dados de log de eventos com eventos MAP
- Usa a mesma infraestrutura SCCP/M3UA
- Funciona com CAMELSessionsLive para monitoramento
- Integra-se com roteamento M3UA existente

Arquivos Modificados

- config/runtime.exe - ATUALIZADO

Dependências

- CapRequestGenerator existente
- CapClient para envio M3UA
- M3UA.Server para transmissão de pacotes

- EventLog para registro de mensagens
- Framework Phoenix LiveView
- Painel de Controle para infraestrutura da interface



Guia de Recursos Comuns

[-- Voltar à Documentação Principal](#)

Este guia cobre recursos comuns a todos os modos de operação do OmniSS7.

Índice

- [1. Visão Geral da Interface Web](#)
- [2. Documentação da API](#)
- [3. Monitoramento e Métricas](#)
- [4. Melhores Práticas](#)

Visão Geral da Interface Web

A Interface Web é acessível através do endereço do servidor web configurado.

OmniSS7 Stack CAMEL GW v1.0.0

Learned to: Omnitouch © 2023 Omnitouch

SS7 Events

SS7 Client

M3UA

CAP Sessions

CAP Requests

Resources

Configuration

M3UA Status

Last updated: 2025-10-25 23:57:55 UTC Refresh

Name	PID	Status	ASP State	Assoc/SCTP	Local	Remote	RC	24h Uptime	Actions
:camelgw_client_asp	#PID=0..1322..0>	Up	active	established	10.5.190.200:2905	10.179.4.10:2905	4	0.1%	▼

M3UA DETAILS

24-Hour Availability Timeline

Uptime: 6.1% Total Up: 1m Total Down: 23h 50m

24h ago

18h ago

12h ago

6h ago

Now

Up

Down

Basic Information

Name:	camelgw_client_asp	PID:	#PID=0..1322..0>	Status:	Up	Mode:	ASP	ASP State:	active	Association State:	established
Routing Context:	4										

Network Configuration

Local IP:	10.5.190.200	Local Port:	2905	Remote IP:	10.179.4.10	Remote Port:	2905
-----------	--------------	-------------	------	------------	-------------	--------------	------

Additional Details

► Raw Data (click to expand)

Navegação Principal

- **Eventos** - Eventos de sinalização SS7 em tempo real e logs de mensagens
- **Aplicação** - Status da aplicação e informações em tempo de execução
- **Configuração** - Visualizador de configuração do sistema
- **Status M3UA** - Conexões de pares M3UA (modo STP)
- **Fila de SMS** - Mensagens SMS de saída (modo SMSc)

Acessando a Interface Web

1. Abra seu navegador web
2. Navegue até o nome do host configurado (por exemplo, <http://localhost>)
3. Visualize o painel de status do sistema

Documentação da API Swagger

Documentação interativa da API:

<http://your-server/swagger>

Configuração da Interface Web

Configure em `config/runtime.exs`:

```
config :control_panel,
  # Ordem das páginas no menu de navegação
  page_order: ["/events", "/application", "/configuration"],

  # Configurações do servidor web
  web: %{
    listen_ip: "0.0.0.0", # IP para vincular (0.0.0.0 para todas as interfaces)
    port: 80, # Porta HTTP (443 para HTTPS)
    hostname: "localhost", # Nome do host do servidor para geração de URL
    enable_tls: false, # Defina como verdadeiro para habilitar HTTPS
    tls_cert: "cert.pem", # Caminho para o arquivo de certificado TLS
    tls_key: "key.pem" # Caminho para o arquivo de chave privada TLS
  }
```

Parâmetros de Configuração:

Parâmetro	Tipo	Padrão	Descrição
page_order	Lista	["/events", "/application", "/configuration"]	Ordem das páginas no menu de navegação
listen_ip	String	"0.0.0.0"	Endereço IP para vincular o servidor web
port	Inteiro	80	Porta HTTP (use 443 para HTTPS)
hostname	String	"localhost"	Nome do host do servidor para geração de URL
enable_tls	Booleano	false	Habilitar HTTPS com TLS
tls_cert	String	"cert.pem"	Caminho para o certificado TLS (quando TLS habilitado)
tls_key	String	"key.pem"	Caminho para a chave privada TLS (quando TLS habilitado)

Configuração do Logger

Configure o nível de log em `config/runtime.exs`:

```
config :logger,
  level: :debug # Opções: :debug, :info, :warning, :error
```

Níveis de Log:

- `:debug` - Informações detalhadas de depuração
- `:info` - Mensagens informativas gerais
- `:warning` - Mensagens de aviso para potenciais problemas

- `:error` - Somente mensagens de erro

Documentação da API

URL Base da API

http://your-server/api

Códigos de Resposta

- **200** - Sucesso
- **400** - Solicitação Inválida
- **504** - Tempo Limite do Gateway

Especificação OpenAPI

http://your-server/swagger.json

Monitoramento e Métricas

Endpoint de Métricas do Prometheus

http://your-server/metrics

Principais Categorias de Métricas

Métricas M3UA/SCTP:

- Mudanças de estado de associação SCTP
- Transições de estado ASP M3UA
- Unidades de dados de protocolo enviadas/recebidas

Métricas M2PA:

- Transições de estado de link (DOWN → ALIGNMENT → PROVING → READY)
- Mensagens e bytes enviados/recebidos por link
- Erros específicos de link (decodificar, codificar, SCTP)

Métricas STP:

- Mensagens recebidas/enviadas por par
- Falhas de roteamento por motivo
- Distribuição de tráfego entre pares

Métricas do Cliente MAP:

- Solicitações MAP por tipo de operação
- Histogramas de duração de solicitações
- Medidor de transações pendentes

Métricas CAP:

- Solicitações CAP por tipo de operação
- Operações de gateway CAMEL

Métricas SMS:

- Profundidade da fila
- Taxas de entrega
- Mensagens falhadas

Integração com Grafana

As métricas do OmniSS7 são compatíveis com Prometheus e Grafana.

Melhores Práticas

Recomendações de Segurança

- 1. Isolamento de Rede**
 - Implantar em VLAN dedicada
 - Regras de firewall para restringir acesso
 - Permitir SCTP apenas de endereços conhecidos
- 2. Segurança da Interface Web**
 - Habilitar TLS para produção
 - Usar proxy reverso com autenticação
 - Restringir a IPs de gerenciamento
- 3. Segurança da API**
 - Implementar limitação de taxa
 - Usar chaves de API ou OAuth
 - Registrar todas as solicitações para auditoria

Ajuste de Desempenho

- 1. Limites de TPS**
 - Configurar TPS apropriado
 - Monitorar carga do sistema
 - Ajustar buffers SCTP
- 2. Otimização de Banco de Dados**
 - Adicionar índices
 - Arquivar mensagens antigas
 - Monitorar pool de conexões
- 3. Ajuste M3UA**
 - Ajustar intervalos de heartbeat SCTP
 - Configurar valores de tempo limite
 - Usar múltiplos links para redundância

Monitoramento e Alerta

Métricas Chave:

- Estado da conexão M3UA
- Taxa de sucesso de solicitações MAP
- Tempos de resposta da API
- Profundidade da fila de mensagens

Limiares de Alerta:

- M3UA inativo > 1 minuto
- Taxa de tempo limite MAP > 10%
- Profundidade da fila > 1000
- Taxa de erro da API > 5%

Referência Completa de Configuração

Todos os Parâmetros de Configuração

Esta seção fornece uma referência completa de todos os parâmetros de configuração disponíveis em todos os modos de operação.

Configuração do Logger (:logger)

```
config :logger,
  level: :debug # :debug | :info | :warning | :error
```

Configuração da Interface Web (:control_panel)

```
config :control_panel,
  page_order: ["/events", "/application", "/configuration"],
  web: %{
    listen_ip: "0.0.0.0",
    port: 80,
    hostname: "localhost",
    enable_tls: false,
    tls_cert: "cert.pem",
    tls_key: "key.pem"
  }
```

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
page_order	Lista de Strings	Não	["/events", "/application", "/configuration"]	Ordem das páginas no menu de navegação
web.listen_ip	String	Sim	"0.0.0.0"	Endereço IP para vincular o servidor web
web.port	Inteiro	Sim	80	Número da porta HTTP/HTTPS
web.hostname	String	Sim	"localhost"	Nome do host do servidor
web.enable_tls	Booleano	Não	false	Habilitar HTTPS
web.tls_cert	String	Se TLS habilitado	"cert.pem"	Caminho para o certificado TLS
web.tls_key	String	Se TLS habilitado	"key.pem"	Caminho para a chave privada TLS

Configuração STP M3UA (:ommiss7)

```
config :ommiss7,
  m3ua_stp: %{
    enabled: false,
    local_ip: {127, 0, 0, 1},
    local_port: 2905
  },
  enable_gt_routing: true,
  m3ua_peers: [...],
  m3ua_routes: [...],
  m3ua_gt_routes: [...]
```

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
m3ua_stp.enabled	Booleano	Sim	false	Habilitar modo STP na inicialização
m3ua_stp.local_ip	Tupla	Sim	{127, 0, 0, 1}	IP para vincular para M3UA de entrada
m3ua_stp.local_port	Inteiro	Sim	2905	Porta SCTP para M3UA
enable_gt_routing	Booleano	Não	false	Habilitar roteamento de Título Global

Parâmetros do Par M3UA:

Parâmetro	Tipo	Obrigatório	Descrição
peer_id	Inteiro	Sim	Identificador único do par
name	String	Sim	Nome descritivo do par
role	Atomo	Sim	:client ou :server
local_ip	Tupla	Se :client	IP local para vincular
local_port	Inteiro	Se :client	Porta local (0 para dinâmica)
remote_ip	Tupla	Sim	IP do par remoto
remote_port	Inteiro	Se :client	Porta do par remoto
routing_context	Inteiro	Sim	Contexto de roteamento M3UA
point_code	Inteiro	Sim	Código de ponto SS7
network_indicator	Atomo	Não	:international ou :national

Parâmetros de Rota M3UA:

Parâmetro	Tipo	Obrigatório	Descrição
dest_pc	Inteiro	Sim	Código de ponto de destino
peer_id	Inteiro	Sim	Par pelo qual roteá-lo
priority	Inteiro	Sim	Prioridade da rota (menor = maior prioridade)
network_indicator	Atomo	Não	:international ou :national

Parâmetros de Rota GT M3UA:

Parâmetro	Tipo	Obrigatório	Descrição
gt_prefix	String	Sim	Prefixo de Título Global a ser correspondido
peer_id	Inteiro	Sim	Par de destino
priority	Inteiro	Sim	Prioridade da rota
description	String	Não	Descrição da rota para registro
source_ssn	Inteiro	Não	Correspondência apenas se SSN de origem corresponder
dest_ssn	Inteiro	Não	Reescrever SSN de destino para este valor

Configuração do Cliente MAP (:ommiss7)

```
config :ommiss7,
  map_client_enabled: false,
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :map_client_asp,
    local_ip: {10, 0, 0, 100},
    local_port: 2905,
    remote_ip: {10, 0, 0, 1},
    remote_port: 2905,
    routing_context: 1
  }
```

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
map_client_enabled	Booleano	Sim	false	Habilitar modo cliente MAP
map_client_m3ua.mode	String	Sim	"ASP"	Modo de conexão M3UA ("ASP" ou "SGP")
map_client_m3ua.callback	Tupla	Sim	{MapClient, :handle_payload, []}	Manipulador de callback de mensagem
map_client_m3ua.process_name	Atomo	Sim	:map_client_asp	Nome do processo registrado
map_client_m3ua.local_ip	Tupla	Sim	-	Endereço IP local
map_client_m3ua.local_port	Inteiro	Sim	2905	Porta SCTP local
map_client_m3ua.remote_ip	Tupla	Sim	-	IP remoto do STP/SGP
map_client_m3ua.remote_port	Inteiro	Sim	2905	Porta SCTP remota
map_client_m3ua.routing_context	Inteiro	Sim	-	Contexto de roteamento M3UA

Configuração do Centro de SMS (:ommiss7)

```
config :ommiss7,
  auto_flush_enabled: false,
  auto_flush_interval: 10_000,
  auto_flush_dest_smsc: nil,
  auto_flush_tps: 10
```

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
auto_flush_enabled	Booleano	Não	false	Habilitar auto-limpeza da fila de SMS
auto_flush_interval	Inteiro	Não	10000	Intervalo de pesquisa da fila (milissegundos)
auto_flush_dest_smsc	String	nil	nil	Filtrar por SMSC de destino (nil = todos)
auto_flush_tps	Inteiro	Não	10	Máximo de transações por segundo

Configuração da API HTTP (:ommiss7)

O backend de SMS agora usa a API HTTP em vez de conexões diretas ao banco de dados.

```
config :ommiss7,
  smsc_api_base_url: "https://10.5.198.200:8443",
  frontend_name: "omni-smsc01" # Opcional: padrão para hostname_SMSC
```

Parâmetros da API:

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
smsc_api_base_url	String	Sim	"https://10.5.198.200:8443"	URL base para a API do backend de SMS
frontend_name	String	Não	"(hostname)_SMSC"	Identificador do frontend para registro

Endpoints da API Utilizados:

- POST /api/frontends - Registrar esta instância de frontend com o backend
- POST /api/messages_raw - Inserir novas mensagens SMS
- GET /api/messages - Recuperar fila de mensagens (com cabeçalho smsc)
- PATCH /api/messages/{id} - Marcar mensagem como entregue
- PUT /api/messages/{id} - Atualizar status da mensagem
- POST /api/events - Adicionar rastreamento de eventos
- GET /api/status - Endpoint de verificação de saúde

Registro do Frontend:

O sistema se registra automaticamente com a API do backend na inicialização e se re-registra a cada 5 minutos. O registro inclui:

- Nome e tipo do frontend (SMSC)
- Nome do host
- Tempo de atividade em segundos
- Detalhes de configuração (formato JSON)

Notas de Configuração:

- A verificação SSL está desabilitada por padrão para certificados autoassinados
- As solicitações HTTP têm tempo limite após 5 segundos
- Todos os timestamps estão no formato ISO 8601
- A API usa JSON para corpos de solicitação/resposta

Documentação Relacionada

- [= Voltar à Documentação Principal](#)
- [Guia STP](#)
- [Guia do Cliente MAP](#)
- [Guia do Centro de SMS](#)
- [Guia HLR](#)



Referência de Configuração

[← Voltar à Documentação Principal](#)

Este documento fornece uma referência abrangente para todos os parâmetros de configuração do OmniSS7.

Índice

1. [Visão Geral](#)
2. [Flags de Modo Operacional](#)
3. [Parâmetros do Modo HLR](#)
4. [Parâmetros do Modo SMS](#)
5. [Parâmetros do Modo STP](#)
6. [Parâmetros de NAT de Título Global](#)
7. [Parâmetros de Conexão M3UA](#)
8. [Parâmetros do Servidor HTTP](#)
9. [Parâmetros do Banco de Dados](#)
10. [Valores Codificados](#)

Visão Geral

A configuração do OmniSS7 é gerenciada através de `config/runtime.exs`. O sistema suporta três modos operacionais:

- **Modo STP** - Ponto de Transferência de Sinal para roteamento
- **Modo HLR** - Registro de Localização do Usuário para gerenciamento de assinantes
- **Modo SMS** - Centro de SMS para entrega de mensagens

Arquivo de Configuração: `config/runtime.exs`

Flags de Modo Operacional

Controla quais recursos estão habilitados.

Parâmetro	Tipo	Padrão	Descrição	Modos
<code>map_client_enabled</code>	Booleano	<code>false</code>	Habilitar cliente MAP e conectividade M3UA	Todos
<code>hlr_mode_enabled</code>	Booleano	<code>false</code>	Habilitar recursos específicos do HLR	HLR
<code>smc_mode_enabled</code>	Booleano	<code>false</code>	Habilitar recursos específicos do SMS	SMS

Exemplo:

```
config :omniss7,  
  map_client_enabled: true,  
  hlr_mode_enabled: true,  
  smc_mode_enabled: false
```

Parâmetros do Modo HLR

Configuração para o modo HLR (Registro de Localização do Usuário).

Configuração da API HLR

Parâmetro	Tipo	Padrão	Necessário	Descrição
hlr_api_base_url	String	-	Sim	URL do endpoint da API HLR de backend (verificação SSL codificada como desabilitada)
hlr_service_center_gt_address	String	-	Sim	Endereço do Título Global HLR retornado nas respostas UpdateLocation
smc_service_center_gt_address	String	-	Sim	Endereço GT do SSMc retornado nas respostas SRI-for-SM

Exemplo:

```
config :omniss7,  
  hlr_api_base_url: "https://10.180.2.140:8443",  
  hlr_service_center_gt_address: "55512341111",  
  smc_service_center_gt_address: "55512341112"
```

Mapeamento MSISDN ↔ IMSI

Configuração para geração sintética de IMSI a partir de MSISDNs. Para uma explicação técnica detalhada do algoritmo de mapeamento, consulte [Mapeamento MSISDN ↔ IMSI no Guia HLR](#).

Parâmetro	Tipo	Padrão	Necessário	Descrição
hlr_imsi_plmn_prefix	String	"50557"	Não	Prefixo PLMN (MCC+MNC) para geração sintética de IMSI
hlr_msisdn_country_code	String	"61"	Não	Prefixo do código do país para mapeamento reverso IMSI→MSISDN
hlr_msisdn_nsn_offset	Inteiro	0	Não	Deslocamento no MSISDN onde o NSN começa (tipicamente o comprimento do código do país)
hlr_msisdn_nsn_length	Inteiro	9	Não	Comprimento do Número Nacional do Assinante a ser extraído do MSISDN

Exemplo (código do país de 2 dígitos):

```
config :omniss7,  
  hlr_imsi_plmn_prefix: "50557",      # MCC 505 + MNC 57  
  hlr_msisdn_country_code: "99",      # Exemplo de código do país de 2 dígitos  
  hlr_msisdn_nsn_offset: 2,           # Ignorar código do país de 2 dígitos  
  hlr_msisdn_nsn_length: 9            # Extrair NSN de 9 dígitos
```

Exemplo (código do país de 3 dígitos):

```
config :omniss7,  
  hlr_imsi_plmn_prefix: "50557",      # MCC 505 + MNC 57  
  hlr_msisdn_country_code: "999",      # Exemplo de código do país de 3 dígitos  
  hlr_msisdn_nsn_offset: 3,           # Ignorar código do país de 3 dígitos  
  hlr_msisdn_nsn_length: 8            # Extrair NSN de 8 dígitos
```

Importante: Defina `nsn_offset` para o comprimento do seu código do país para extrair corretamente o NSN. Por exemplo:

- Código do país "9" (1 dígito) → `nsn_offset: 1`
- Código do país "99" (2 dígitos) → `nsn_offset: 2`
- Código do país "999" (3 dígitos) → `nsn_offset: 3`

Configuração do InsertSubscriberData (ISD)

Configuração para dados de provisionamento de assinantes enviados para VLRs durante UpdateLocation. Para uma explicação detalhada da sequência ISD e do fluxo de mensagens, consulte [Configuração do InsertSubscriberData no Guia HLR](#).

Parâmetro	Tipo	Padrão	Necessário	Descrição
isd_network_access_mode	Átomo	:packetAndCircuit	Não	Tipo de acesso à rede: :packetAndCircuit, :packetOnly ou :circuitOnly
isd_send_ss_data	Booleano	true	Não	Enviar ISD #2 com dados de Serviços Suplementares
isd_send_call_barring	Booleano	true	Não	Enviar ISD #3 com dados de Bloqueio de Chamadas

Exemplo:

```
config :omniss7,
  isd_network_access_mode: :packetAndCircuit,
  isd_send_ss_data: true,
  isd_send_call_barring: true
```

Configuração CAMEL

Configuração para roteamento inteligente de chamadas baseado em CAMEL. Para uma explicação detalhada da integração CAMEL e das chaves de serviço, consulte [Integração CAMEL no Guia HLR](#).

Parâmetro	Tipo	Padrão	Necessário	Descrição
camel_service_key	Inteiro	11_110	Não	Chave de serviço CAMEL para respostas SRI
camel_trigger_detection_point	Átomo	:termAttemptAuthorized	Não	Ponto de detecção do gatilho CAMEL: :termAttemptAuthorized, :tBusy, :tNoAnswer, :tAnswer
camel_gsmcf_gt_address	String	(usa GT chamado)	Não	Título Global gsmSCF padrão para respostas CAMEL (pode ser substituído por GT NAT)

Exemplo:

```
config :omniss7,
  camel_service_key: 11_110,
  camel_trigger_detection_point: :termAttemptAuthorized
```

Prefixos do VLR Local

Configuração para distinguir assinantes locais e em roaming. Para uma explicação detalhada da detecção de local/em roaming e das operações PRN, consulte [Tratamento de Assinantes em Roaming no Guia HLR](#).

Parâmetro	Tipo	Padrão	Necessário	Descrição
home_vlr_prefixes	Lista	["5551231"]	Não	Prefixos GT do VLR considerados rede "local"

Exemplo:

```
config :omniss7,
  home_vlr_prefixes: ["5551231", "5551234"]
```

Parâmetros do Modo SMSc

Configuração para o modo Centro de SMS.

Configuração da API SMS

Parâmetro	Tipo	Padrão	Necessário	Descrição
smc_api_base_url	String	-	Sim	URL do endpoint da API SMS de backend (verificação SSL codificada como desabilitada)
smc_name	String	"{hostname}_SMC"	Não	Identificador do SMS para registro de backend
smc_service_center_gt_address	String	-	Sim	Endereço do Título Global do Centro de Serviço

Exemplo:

```
config :omniss7,  
  smc_api_base_url: "https://10.179.3.219:8443",  
  smc_name: "ipsmgw",  
  smc_service_center_gt_address: "55512341112"
```

Nota: O registro de frontend ocorre a cada **5 minutos** (codificado) através do módulo SMS.FrontendRegistry.

Configuração de Auto-Flush

Parâmetro	Tipo	Padrão	Necessário	Descrição
auto_flush_enabled	Booleano	true	Não	Habilitar processamento automático da fila de SMS
auto_flush_interval	Inteiro	10_000	Não	Intervalo de processamento da fila em milissegundos
auto_flush_dest_smc	String	-	Sim	Nome do SMS de destino para auto-flush
auto_flush_tps	Inteiro	10	Não	Taxa de processamento de mensagens (transações/segundo)

Exemplo:

```
config :omniss7,  
  auto_flush_enabled: true,  
  auto_flush_interval: 10_000,  
  auto_flush_dest_smc: "ipsmgw",  
  auto_flush_tps: 10
```

Parâmetros do Modo STP

Configuração para o modo Ponto de Transferência de Sinal M3UA. Para configuração de roteamento detalhada e exemplos, consulte o [Guia de Configuração STP](#).

Servidor STP Autônomo

Parâmetro	Tipo	Padrão	Necessário	Descrição
m3ua_stp.enabled	Booleano	false	Não	Habilitar servidor STP M3UA autônomo
m3ua_stp.local_ip	Tupla	{127, 0, 0, 1}	Não	Endereço IP para escutar conexões
m3ua_stp.local_port	Inteiro	2905	Não	Porta para escutar
m3ua_stp.point_code	Inteiro	-	Sim (se habilitado)	Código de ponto SS7 deste STP

Exemplo:

```
config :omniss7,  
  m3ua_stp: %{  
    enabled: true,
```

```

local_ip: {10, 179, 4, 10},
local_port: 2905,
point_code: 100
}

```

Roteamento de Título Global

Parâmetro	Tipo	Padrão	Necessário	Descrição
enable_gt_routing	Booleano	false	Não	Habilitar roteamento GT além do roteamento PC

Exemplo:

```

config :omniss7,
  enable_gt_routing: true

```

Parâmetros de NAT de Título Global

A Tradução de Endereço de Rede de Título Global permite diferentes GTs de resposta com base no prefixo da parte chamadora. Para uma explicação detalhada e exemplos, consulte o [Guia de NAT de Título Global](#).

Parâmetro	Tipo	Padrão	Necessário	Descrição
gt_nat_enabled	Booleano	false	Não	Habilitar/desabilitar recurso de NAT GT
gt_nat_rules	Lista de Mapas []		Sim (se habilitado)	Lista de mapeamentos de prefixo para GT

Formato da Regra: Cada regra em gt_nat_rules deve ser um mapa com:

- calling_prefix: Prefixo de string para corresponder ao GT chamador
- response_gt: Título Global a ser usado nas respostas

Exemplo:

```

config :omniss7,
  gt_nat_enabled: true,
  gt_nat_rules: [
    # Quando chamado de GT começando com "8772", responda com "55512341112"
    %{calling_prefix: "8772", response_gt: "55512341112"},
    # Quando chamado de GT começando com "8773", responda com "55512341111"
    %{calling_prefix: "8773", response_gt: "55512341111"},
    # Fallback padrão (prefixo vazio corresponde a todos)
    %{calling_prefix: "", response_gt: "55512311555"}
  ]

```

Veja Também: [Guia de NAT GT](#) para uso detalhado e exemplos.

Parâmetros de Conexão M3UA

Configuração de conexão M3UA para modo cliente MAP. Para uso detalhado e exemplos, consulte o [Guia do Cliente MAP](#).

Parâmetro	Tipo	Padrão	Necessário	Descrição
map_client_m3ua.mode	String	-	Sim	Modo de conexão: "ASP" ou "SGP"
map_client_m3ua.callback	Tupla	-	Sim	Módulo/função de callback: {MapClient, :handle_payload, []}
map_client_m3ua.process_name	Átomo	-	Sim	Nome do processo para registro
map_client_m3ua.local_ip	Tupla	-	Sim	Endereço IP local para vincular
map_client_m3ua.local_port	Inteiro	2905	Sim	Porta SCTP local
map_client_m3ua.remote_ip	Tupla	-	Sim	Endereço IP remoto STP/SGW
map_client_m3ua.remote_port	Inteiro	2905	Sim	Porta SCTP remota

Parâmetro	Tipo	Padrão	Necessário	Descrição
map_client_m3ua.routing_context	Inteiro-		Sim	ID do contexto de roteamento M3UA

Exemplo:

```
config :omniss7,
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :h3r_client_asp,
    local_ip: {10, 179, 4, 11},
    local_port: 2905,
    remote_ip: {10, 179, 4, 10},
    remote_port: 2905,
    routing_context: 1
  }
```

Parâmetros do Servidor HTTP

Configuração para o servidor HTTP da API REST.

Parâmetro	Tipo	Padrão	Necessário	Descrição
start_http_server	Booleano	true	Não	Habilitar/desabilitar servidor HTTP (porta 8080)

Valores Codificados (não configuráveis):

- **IP:** 0.0.0.0 (todas as interfaces)
- **Porta:** 8080
- **Transporte:** Plug.Cowboy

Exemplo:

```
config :omniss7,
  start_http_server: true # Defina como false para desabilitar
```

Endpoints da API:

- API REST: [http://\[server-ip\]:8080/api/*](http://[server-ip]:8080/api/*)
- Swagger UI: [http://\[server-ip\]:8080/swagger](http://[server-ip]:8080/swagger)
- Métricas Prometheus: [http://\[server-ip\]:8080/metrics](http://[server-ip]:8080/metrics)

Consulte o [Guia da API](#) para detalhes.

Parâmetros do Banco de Dados

Configuração para persistência do banco de dados Mnesia.

Parâmetro	Tipo	Padrão	Necessário	Descrição
mnesia_storage_type	Átomo	:disc_copies	Não	Tipo de armazenamento Mnesia: :disc_copies ou :ram_copies

Exemplo:

```
config :omniss7,
  mnesia_storage_type: :disc_copies # Produção
  # mnesia_storage_type: :ram_copies # Apenas para testes
```

Tipos de Armazenamento:

- `:disc_copies` - Armazenamento em disco persistente (sobrevive a reinicializações) - **Recomendado para produção**
- `:ram_copies` - Somente em memória (perdido na reinicialização) - Apenas para testes

Tabelas Mnesia:

- `m3ua_peer` - Conexões de pares M3UA
- `m3ua_route` - Rotas de Código de Ponto
- `m3ua_gt_route` - Rotas de Título Global

Localização: `diretório Mnesia.{node_name}/`

Valores Codificados

Os seguintes valores são **codificados no código-fonte** e não podem ser alterados via configuração.

Timeouts

Valor	Impacto	Solução Alternativa
Timeout de requisição MAP: 10 segundos	Todas as operações MAP expiram após 10s	Modificar código-fonte
Timeout ISD: 10 segundos	Cada mensagem ISD expira após 10s	Modificar código-fonte

Servidor HTTP

Valor	Impacto	Solução Alternativa
IP HTTP: 0.0.0.0	O servidor escuta em todas as interfaces	Modificar código-fonte
Porta HTTP: 8080	A API REST é executada na porta 8080	Modificar código-fonte

Verificação SSL

Valor	Impacto	Solução Alternativa
API HLR SSL: desabilitado	A verificação SSL está sempre desabilitada	Modificar código-fonte
API SMSc SSL: desabilitado	A verificação SSL está sempre desabilitada	Modificar código-fonte

Intervalos de Registro

Valor	Impacto	Solução Alternativa
Registro de frontend: 5 minutos	SMSc registra com o backend a cada 5 min	Modificar código-fonte

Atualização Automática da Interface Web

Página	Intervalo
Gerenciamento de Roteamento	5 segundos
Assinantes Ativos	2 segundos

Exemplos de Configuração

Configuração Mínima do HLR

```
config :omniss7,
  map_client_enabled: true,
  hlr_mode_enabled: true,
  smsc_mode_enabled: false,

  hlr_api_base_url: "https://10.180.2.140:8443",
```

```

hlr_service_center_gt_address: "55512341111",
smsc_service_center_gt_address: "55512341112",

map_client_m3ua: %{
  mode: "ASP",
  callback: {MapClient, :handle_payload, []},
  process_name: :hlc_client_asp,
  local_ip: {10, 179, 4, 11},
  local_port: 2905,
  remote_ip: {10, 179, 4, 10},
  remote_port: 2905,
  routing_context: 1
}

```

Configuração Mínima do SMSc

```

config :omniss7,
  map_client_enabled: true,
  hlr_mode_enabled: false,
  smsc_mode_enabled: true,

  smsc_api_base_url: "https://10.179.3.219:8443",
  smsc_name: "ipsmgw",
  smsc_service_center_gt_address: "55512341112",

  auto_flush_enabled: true,
  auto_flush_interval: 10_000,
  auto_flush_dest_smsc: "ipsmgw",
  auto_flush_tps: 10,

  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :stp_client_asp,
    local_ip: {10, 179, 4, 12},
    local_port: 2905,
    remote_ip: {10, 179, 4, 10},
    remote_port: 2905,
    routing_context: 1
  }

```

STP com Servidor Autônomo

```

config :omniss7,
  map_client_enabled: true,
  hlr_mode_enabled: false,
  smsc_mode_enabled: false,

  enable_gt_routing: true,
  mnesia_storage_type: :disc_copies,

  m3ua_stp: %{
    enabled: true,
    local_ip: {10, 179, 4, 10},
    local_port: 2905,
    point_code: 100
  },

  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :stp_client_asp,
    local_ip: {10, 179, 4, 10},

```

```
local_port: 2906,  
remote_ip: {10, 179, 4, 11},  
remote_port: 2905,  
routing_context: 1  
}
```

Resumo

Total de Parâmetros de Configuração: 32

Por Categoria:

- Modo Operacional: 3 parâmetros
- Modo HLR: 13 parâmetros
- Modo SMSc: 7 parâmetros
- Modo STP: 5 parâmetros
- Conexão M3UA: 8 parâmetros
- Servidor HTTP: 1 parâmetro
- Banco de Dados: 1 parâmetro

Parâmetros Necessários (devem ser configurados):

- hlr_api_base_url (modo HLR)
 - hlr_service_center_gt_address (modo HLR)
 - smsc_api_base_url (modo SMSc)
 - smsc_service_center_gt_address (modo SMSc/HLR)
 - Todos os parâmetros map_client_m3ua.*
 - m3ua_stp.point_code (se STP habilitado)
-

Documentação Relacionada

- [Guia HLR](#) - Configuração específica do HLR
- [Guia SMSc](#) - Configuração específica do SMSc
- [Guia STP](#) - Configuração de roteamento STP
- [Guia da API](#) - Referência da API REST
- [Guia da Interface Web](#) - Documentação da interface web



Guia de NAT de Título Global

Visão Geral

A Tradução de Endereço de Título Global (GT NAT) é um recurso que permite ao OmniSS7 responder com diferentes endereços de Título Global com base no prefixo GT da parte chamadora, no prefixo GT da parte chamada ou em uma combinação de ambos. Isso é essencial ao operar com múltiplos Títulos Globais e precisar garantir que as respostas utilizem o GT correto com base em qual rede ou par está chamando e/ou qual GT eles chamaram.

Novidades (GT NAT Aprimorado)

O recurso GT NAT foi aprimorado com novas capacidades poderosas:

Novos Recursos

1. **Correspondência de Prefixo da Parte Chamado:** As regras agora podem corresponder ao `called_prefix` além do `calling_prefix`
2. **Correspondência Combinada:** As regras podem corresponder a prefixos de chamada E de chamada simultaneamente
3. **Priorização Baseada em Peso:** As regras agora usam um campo `weight` (menor = maior prioridade) em vez de apenas o comprimento do prefixo
4. **Correspondência Flexível:** Agora você pode criar regras com:
 - Apenas prefixo de chamada
 - Apenas prefixo da parte chamada
 - Ambos os prefixos de chamada e da parte chamada
 - Nenhum (regra de curinga/recuo)

Novo Formato de Regra

Campos obrigatórios:

- `weight`: Prioridade inteira (menor = maior prioridade)
- `response_gt`: O GT a ser usado na resposta

Campos opcionais (pelo menos um recomendado para correspondência específica):

- `calling_prefix`: Correspondência no prefixo GT da parte chamadora
- `called_prefix`: Correspondência no prefixo GT da parte chamada

Exemplo:

```
gt_nat_rules: [  
  # Regra específica com ambos os prefixos - maior prioridade  
  %{calling_prefix: "8772", called_prefix: "555", weight: 1,  
  response_gt: "111111"},  
  
  # Regras específicas - prioridade média  
  %{calling_prefix: "8772", weight: 10, response_gt: "222222"},  
  %{called_prefix: "555", weight: 10, response_gt: "333333"},  
  
  # Recuo de curinga - menor prioridade  
  %{weight: 100, response_gt: "999999"}  
]
```

Casos de Uso

Operação Multi-Rede

Quando você tem várias redes parceiras e cada uma espera respostas de um GT específico:

- **Rede A** chama seu GT 111111 e espera respostas de 111111
- **Rede B** chama seu GT 222222 e espera respostas de 222222

Sem o GT NAT, você precisaria de instâncias separadas ou roteamento complexo. Com o GT NAT, uma única instância do OmniSS7 pode lidar com isso de forma inteligente.

Cenários de Roaming

Ao operar como um HLR ou SMSc com acordos de roaming:

- Assinantes da **rede doméstica** usam GT 555000
- **Parceiro de roaming 1** usa GT 555001
- **Parceiro de roaming 2** usa GT 555002

O GT NAT garante que cada parceiro receba respostas do GT correto ao qual estão configurados para rotear.

Testes e Migração

Durante migrações de rede ou testes:

- Migre gradualmente o tráfego do GT antigo para o novo GT
- Mantenha ambos os GTs durante o período de transição
- Roteie respostas com base em qual GT o chamador usou

Como Funciona

Fluxo de Tradução de Endereço

1. **Solicitação de Entrada:** O OmniSS7 recebe uma mensagem SCCP com:
 - GT da Parte Chamado: 55512341112 (seu GT)
 - GT da Parte Chamadora: 877234567 (GT deles)
2. **Consulta GT NAT:** O sistema verifica o GT de chamada 877234567 contra as regras de prefixo configuradas
3. **Correspondência de Prefixo:** Encontra o prefixo correspondente mais longo (por exemplo, 8772 corresponde a 877234567)
4. **Seleção do GT de Resposta:** Usa response_gt da regra correspondente (por exemplo, 55512341112)
5. **Resposta Enviada:** A resposta SCCP usa:
 - GT da Parte Chamado: 877234567 (invertido - GT deles)
 - GT da Parte Chamadora: 55512341112 (GT NAT'd)

Tipos de Resposta Afetados

O GT NAT se aplica a múltiplas camadas da pilha SS7:

Camada SCCP (Todas as Respostas)

- Endereços GT Chamado/Chamador SCCP em todas as mensagens de resposta
- Reconhecimentos de ISD (InsertSubscriberData)
- Respostas de UpdateLocation
- Respostas de erro

Camada MAP (Operação Específica)

- **Respostas SRI-for-SM:** networkNode-Number (endereço GT do SMSc)
- **UpdateLocation:** hlr-Number nas respostas
- **InsertSubscriberData:** GT do HLR em mensagens ISD

Configuração

Configuração Básica

Adicione ao config/runtime.exs:

```

config :omniss7,
  # Ativar GT NAT
  gt_nat_enabled: true,

  # Definir regras de GT NAT
  gt_nat_rules: [
    # Regra 1: Chamadas do prefixo "8772" recebem resposta do
    "55512341112"
    %{calling_prefix: "8772", response_gt: "55512341112"},

    # Regra 2: Chamadas do prefixo "8773" recebem resposta do
    "55512341111"
    %{calling_prefix: "8773", response_gt: "55512341111"},

    # Regra padrão (prefixo vazio corresponde a tudo)
    %{calling_prefix: "", response_gt: "55512311555"}
  ]

```

Parâmetros de Configuração

Para referência completa de configuração, veja [Parâmetros de NAT de Título Global na Referência de Configuração](#).

Parâmetro	Tipo	Obrigatório	Descrição
gt_nat_enabled	Booleano	Sim	Ativar/desativar recurso GT NAT
gt_nat_rules	Lista de Mapas	Sim (se ativado)	Lista de regras de correspondência de prefixo

Formato da Regra

Cada regra é um mapa com as seguintes chaves:

```

%{
  calling_prefix: "8772",      # (Opcional) Prefixo para corresponder
ao GT chamador
  called_prefix: "555",       # (Opcional) Prefixo para corresponder
ao GT chamado
  weight: 10,                 # (Obrigatório) Valor de prioridade
(menor = maior prioridade)
  response_gt: "55512341112"  # (Obrigatório) GT a ser usado nas
respostas
}

```

Campos da Regra:

- **calling_prefix** (Opcional): Prefixo de string para corresponder ao GT chamador de entrada

- A correspondência é feita por `String.starts_with?/2`
- A string vazia "" ou nil atua como curinga (corresponde a qualquer GT chamador)
- Pode ser omitido para corresponder a qualquer GT chamador
- **called_prefix** (Opcional): Prefixo de string para corresponder ao GT chamado de entrada
 - A correspondência é feita por `String.starts_with?/2`
 - A string vazia "" ou nil atua como curinga (corresponde a qualquer GT chamado)
 - Pode ser omitido para corresponder a qualquer GT chamado
- **weight** (Obrigatório): Valor de prioridade inteira
 - Peso menor = maior prioridade (processado primeiro)
 - Deve ser ≥ 0
 - Usado como critério de ordenação primário para regras de correspondência
- **response_gt** (Obrigatório): O endereço de Título Global a ser usado nas respostas
 - Deve ser uma string de número E.164 válida
 - Deve corresponder a um dos seus GTs configurados

Pelo menos um dos `calling_prefix` ou `called_prefix` deve ser especificado para roteamento específico. Ambos podem ser omitidos para uma regra de curinga/recuo.

Lógica de Correspondência de Regras

As regras são avaliadas por **peso primeiro (crescente)**, depois pela **especificidade combinada do prefixo**:

Algoritmo de Correspondência:

1. Filtrar regras onde todos os prefixos especificados correspondem
 - Se `calling_prefix` estiver definido, deve corresponder ao GT chamador
 - Se `called_prefix` estiver definido, deve corresponder ao GT chamado
 - Se ambos estiverem definidos, ambos devem corresponder
 - Se nenhum estiver definido, a regra atua como um curinga
2. Classificar regras correspondentes por:
 - **Primário**: Peso (crescente - valores menores primeiro)
 - **Secundário**: Comprimento combinado do prefixo (decrecente - mais longo = mais específico)
3. Retornar a primeira regra correspondente

Exemplos:

```
# Regras de exemplo
gt_nat_rules: [
  # Peso 1: Maior prioridade - corresponde a ambos os prefixos
  %{calling_prefix: "8772", called_prefix: "555", weight: 1,
  response_gt: "111111"},

  # Peso 10: Prioridade média - regras específicas
  %{calling_prefix: "8772", weight: 10, response_gt: "222222"}, #
  Apenas chamando
  %{called_prefix: "555", weight: 10, response_gt: "333333"}, #
  Apenas chamado

  # Peso 100: Menor prioridade - recuo curinga
  %{weight: 100, response_gt: "444444"} # Corresponde a tudo
]

# Exemplos de correspondência:
# Chamando: "877234567", Chamado: "555123" -> "111111" (peso 1, ambos
correspondem)
# Chamando: "877234567", Chamado: "999999" -> "222222" (peso 10,
apenas chamando)
# Chamando: "999999999", Chamado: "555123" -> "333333" (peso 10,
apenas chamado)
# Chamando: "999999999", Chamado: "888888" -> "444444" (peso 100,
curinga)
```

Exemplos

Exemplo 1: Dois Parceiros de Rede

Cenário: Você opera um SMSc com dois parceiros de rede. Cada um espera respostas de um GT diferente.

```
config :omniss7,
  gt_nat_enabled: true,

  # GT padrão do SMSc (usado quando o GT NAT está desativado ou
  nenhuma regra corresponde)
  smsc_service_center_gt_address: "5551000",

  # Regras de GT NAT para parceiros
  gt_nat_rules: [
    # Parceiro A (prefixo 4412) espera respostas do GT 5551001
    %{calling_prefix: "4412", weight: 10, response_gt: "5551001"},

    # Parceiro B (prefixo 4413) espera respostas do GT 5551002
```

```
%{calling_prefix: "4413", weight: 10, response_gt: "5551002"},  
  
# Padrão: usar GT padrão do SMS Sc (recuo curinga)  
%{weight: 100, response_gt: "5551000"}  
]
```

Fluxo de Tráfego:

SRI-for-SM de entrada de 44121234567:
GT Chamado: 5551001 (seu GT que o Parceiro A usa)
GT Chamador: 44121234567 (GT do Parceiro A)

Consulta GT NAT:
"44121234567" corresponde ao prefixo "4412"
GT de resposta selecionado: "5551001"

Resposta SRI-for-SM para 44121234567:
GT Chamado: 44121234567 (invertido)
GT Chamador: 5551001 (NAT'd)
networkNode-Number: 5551001 (na resposta MAP)

Exemplo 2: HLR com GTs Regionais

Cenário: HLR nacional com diferentes GTs por região.

```
config :omniss7,  
  gt_nat_enabled: true,  
  hlr_service_center_gt_address: "555000", # GT padrão do HLR  
  
  gt_nat_rules: [  
    # VLRs da região norte (prefixo 5551)  
    %{calling_prefix: "5551", weight: 10, response_gt: "555100"},  
  
    # VLRs da região sul (prefixo 5552)  
    %{calling_prefix: "5552", weight: 10, response_gt: "555200"},  
  
    # VLRs da região oeste (prefixo 5553)  
    %{calling_prefix: "5553", weight: 10, response_gt: "555300"},  
  
    # Padrão para outras regiões (curinga)  
    %{weight: 100, response_gt: "555000"}  
  ]
```

Exemplo 3: Cenário de Migração

Cenário: Migrando gradualmente do GT antigo para o novo GT.

```
config :omniss7,
```

```

gt_nat_enabled: true,
hlr_service_center_gt_address: "123456789", # GT antigo (padrão)

gt_nat_rules: [
  # Redes migradas (já atualizaram suas configurações)
  %{calling_prefix: "555", weight: 10, response_gt: "987654321"},
# Novo GT
  %{calling_prefix: "666", weight: 10, response_gt: "987654321"},
# Novo GT

  # Todos os outros ainda usam o GT antigo (curinga)
  %{weight: 100, response_gt: "123456789"} # GT antigo
]

```

Exemplo 4: Correspondência de Prefixo da Parte Chamado (NOVO)

Cenário: Você tem múltiplos GTs para diferentes serviços e deseja responder com o GT correto com base em qual GT foi chamado.

```

config :omniss7,
  gt_nat_enabled: true,

  gt_nat_rules: [
    # Quando chamam seu GT de SMS (5551xxx), responda com esse GT
    %{called_prefix: "5551", weight: 10, response_gt: "555100"},

    # Quando chamam seu GT de Voz (5552xxx), responda com esse GT
    %{called_prefix: "5552", weight: 10, response_gt: "555200"},

    # Quando chamam seu GT de Dados (5553xxx), responda com esse GT
    %{called_prefix: "5553", weight: 10, response_gt: "555300"},

    # Recuo padrão
    %{weight: 100, response_gt: "555000"}
  ]

```

Fluxo de Tráfego:

Solicitação de entrada para GT Chamado: 555100 (seu GT de SMS)
 GT Chamador: 441234567 (qualquer chamador)

Consulta GT NAT:
 GT Chamado "555100" corresponde ao prefixo "5551"
 GT de resposta selecionado: "555100"

Resposta usa GT Chamador: 555100 (corresponde ao que eles chamaram)

Exemplo 5: Correspondência Combinada de Prefixo Chamando + Chamado (AVANÇADO)

Cenário: Diferentes parceiros chamam diferentes GTs, e você deseja controle detalhado.

```
config :omniss7,  
  gt_nat_enabled: true,  
  
  gt_nat_rules: [  
    # Parceiro A chamando seu GT de SMS - maior prioridade (peso 1)  
    %{calling_prefix: "4412", called_prefix: "5551", weight: 1,  
response_gt: "555101"},  
  
    # Parceiro B chamando seu GT de SMS - maior prioridade (peso 1)  
    %{calling_prefix: "4413", called_prefix: "5551", weight: 1,  
response_gt: "555102"},  
  
    # Qualquer um chamando seu GT de SMS - prioridade média (peso 10)  
    %{called_prefix: "5551", weight: 10, response_gt: "555100"},  
  
    # Parceiro A chamando qualquer GT - prioridade média (peso 10)  
    %{calling_prefix: "4412", weight: 10, response_gt: "555200"},  
  
    # Recuo padrão - baixa prioridade (peso 100)  
    %{weight: 100, response_gt: "555000"}  
  ]
```

Exemplos de Correspondência:

```
# Parceiro A chama GT de SMS  
Chamando: "441234567", Chamado: "555100"  
→ Corresponde à regra de peso 1 (ambos os prefixos) → "555101"  
  
# Parceiro A chama GT de Voz  
Chamando: "441234567", Chamado: "555200"  
→ Corresponde à regra de peso 10 (apenas chamando) → "555200"  
  
# Chamador desconhecido chama GT de SMS  
Chamando: "999999999", Chamado: "555100"  
→ Corresponde à regra de peso 10 (apenas chamado) → "555100"  
  
# Chamador desconhecido chama GT de Voz  
Chamando: "999999999", Chamado: "555200"  
→ Corresponde ao peso 100 curinga → "555000"
```

Modos Operacionais

O GT NAT funciona em todos os modos operacionais do OmniSS7:

Modo HLR

O GT NAT afeta:

- Respostas de UpdateLocation (GT do HLR na resposta)
- Mensagens de InsertSubscriberData (GT do HLR como parte chamadora)
- Respostas de SendAuthenticationInfo
- Respostas de Cancel Location

Para mais informações sobre operações HLR, veja o [Guia de Configuração HLR](#).

Configuração:

```
config :omniss7,
  hlr_mode_enabled: true,
  hlr_service_center_gt_address: "5551234567", # GT padrão do HLR

  gt_nat_enabled: true,
  gt_nat_rules: [
    %{calling_prefix: "331", weight: 10, response_gt: "5551234568"},
# França
    %{calling_prefix: "44", weight: 10, response_gt: "5551234569"},
# Reino Unido
    %{weight: 100, response_gt: "5551234567"} # Curinga padrão
  ]
```

Modo SMSc

O GT NAT afeta:

- Respostas SRI-for-SM (campo networkNode-Number) - veja [Detalhes SRI-for-SM](#)
- Reconhecimentos de MT-ForwardSM

Para mais informações sobre operações SMSc, veja o [Guia de Configuração SMSc](#).

Configuração:

```
config :omniss7,
  smsc_mode_enabled: true,
  smsc_service_center_gt_address: "5559999", # GT padrão do SMSc

  gt_nat_enabled: true,
  gt_nat_rules: [
```

```

    %{calling_prefix: "1", weight: 10, response_gt: "5559991"},    #
América do Norte
    %{calling_prefix: "44", weight: 10, response_gt: "5559992"},    #
Reino Unido
    %{calling_prefix: "86", weight: 10, response_gt: "5559993"},    #
China
    %{weight: 100, response_gt: "5559999"}    # Curinga padrão
]

```

Modo Gateway CAMEL

O GT NAT afeta:

- Todas as respostas em nível SCCP (GT gsmSCF como Parte Chamadora)
- Respostas de operações CAMEL/CAP (InitialDP, EventReportBCSM, etc.)
- Reconhecimentos de RequestReportBCSMEvent
- Respostas de ApplyCharging
- Respostas de Continue

Configuração:

```

config :omniss7,
  camelgw_mode_enabled: true,
  camel_gsmscf_gt_address: "55512341112",    # GT padrão gsmSCF

  gt_nat_enabled: true,
  gt_nat_rules: [
    %{calling_prefix: "555", weight: 10, response_gt: "55512341111"},
# Rede A
    %{calling_prefix: "666", weight: 10, response_gt: "55512311555"},
# Rede B
    %{weight: 100, response_gt: "55512341112"}    # Curinga padrão
  ]

```

Caso de Uso: Ao operar como um gsmSCF (Função de Controle de Serviço) para várias redes, cada gsmSSF da rede pode esperar respostas de um GT gsmSCF específico. O GT NAT garante que o GT correto seja usado com base em qual gsmSSF está chamando.

Registro e Depuração

Ativar Registro de GT NAT

O GT NAT inclui registro automático de todas as traduções:

```

# Nos logs, você verá:
[info] GT NAT [resposta SRI-for-SM]: GT Chamador 877234567 -> GT de
Resposta 55512341112

```

```
[info] GT NAT [UpdateLocation ISD]: GT Chamador 331234567 -> GT de Resposta 55512341111
[info] GT NAT [resposta MAP BEGIN]: GT Chamador 441234567 -> GT de Resposta 55512311555
```

O campo de contexto mostra onde o NAT foi aplicado:

- "resposta SRI-for-SM" - No manipulador SRI-for-SM
- "UpdateLocation ISD" - Em mensagens InsertSubscriberData
- "UpdateLocation END" - Na resposta UpdateLocation END
- "resposta MAP BEGIN" - Respostas MAP BEGIN genéricas
- "ISD ACK" - Reconhecimento ISD
- "resposta de erro HLR" - Resposta de erro do HLR
- "resposta CAMEL" - Respostas de operações CAMEL/CAP (gsmSCF)

Validação

O sistema valida a configuração do GT NAT na inicialização:

```
# Verifique a configuração do GT NAT
iex> GtNat.validate_config()
{:ok, [
  %{calling_prefix: "8772", weight: 10, response_gt: "55512341112"},
  %{calling_prefix: "8773", weight: 10, response_gt: "55512341111"}
]}

# Verifique se está ativado
iex> GtNat.enabled?()
true

# Obtenha todas as regras
iex> GtNat.get_rules()
[
  %{calling_prefix: "8772", weight: 10, response_gt: "55512341112"},
  %{calling_prefix: "8773", weight: 10, response_gt: "55512341111"}
]
```

Testando o GT NAT

Teste a lógica do GT NAT programaticamente:

```
# Teste a tradução com GT chamador apenas (called_gt é nil)
iex> GtNat.translate_response_gt("877234567", nil, "default_gt")
"55512341112"

# Teste a tradução com ambos os GTs chamador e chamado
iex> GtNat.translate_response_gt("877234567", "555123", "default_gt")
"55512341112"
```

```
# Teste com registro (GT chamado nil)
iex> GtNat.translate_response_gt_with_logging("877234567", nil,
"default_gt", "test")
# Logs: GT NAT [test]: GT Chamador 877234567 -> GT de Resposta
55512341112
"55512341112"

# Teste com registro (ambos os GTs)
iex> GtNat.translate_response_gt_with_logging("877234567", "555123",
"default_gt", "test")
# Logs: GT NAT [test]: GT Chamador 877234567, GT Chamado 555123 -> GT
de Resposta 55512341112
"55512341112"

# Teste sem correspondência (retorna padrão)
iex> GtNat.translate_response_gt("999999999", "888888", "default_gt")
"default_gt"
```

Resolução de Problemas

Problema: GT NAT Não Funcionando

Verifique 1: Está ativado?

```
iex> Application.get_env(:omniss7, :gt_nat_enabled)
true # Deve ser true
```

Verifique 2: As regras estão configuradas?

```
iex> Application.get_env(:omniss7, :gt_nat_rules)
[%{calling_prefix: "8772", response_gt: "55512341112"}, ...] # Deve
retornar lista
```

Verifique 3: Verifique os logs Procure por "GT NAT" nos logs para ver se as traduções estão acontecendo.

Problema: GT Errado nas Respostas

Sintoma: Respostas usam endereço GT inesperado

Causa: A correspondência de prefixo da regra pode ser muito ampla ou a regra padrão está capturando o tráfego

Solução: Revise os pesos e prefixos das regras:

```
# RUIM: Curinga com peso baixo (captura tudo primeiro)
gt_nat_rules: [
```

```

    %{weight: 1, response_gt: "111111"},          # Isso
corresponde a tudo primeiro!
    %{calling_prefix: "8772", weight: 10, response_gt: "222222"} #
Nunca alcançado
]

# BOM: Regras específicas com peso menor, curinga com peso maior
gt_nat_rules: [
    %{calling_prefix: "8772", weight: 10, response_gt: "222222"}, #
Específico, peso baixo
    %{weight: 100, response_gt: "111111"} # Curinga, peso alto (recuo)
]

```

Problema: GT NAT Não Aplicado a Tipo de Mensagem Específico

Sintoma: Algumas respostas usam GT NAT'd, outras não

Cobertura Atual:

- ✧ GT Chamador SCCP (todas as respostas)
- ✧ Respostas SRI-for-SM (networkNode-Number)
- ✧ Mensagens ISD de UpdateLocation (GT do HLR)
- ✧ Respostas de UpdateLocation END
- ✧ Reconhecimentos de ISD
- ✧ Respostas MAP BEGIN

Se um tipo de mensagem específico não estiver usando GT NAT, pode não ter sido implementado ainda. Verifique o código-fonte ou entre em contato com o suporte.

Considerações de Desempenho

Desempenho de Consulta

O GT NAT usa correspondência de prefixo simples com complexidade $O(n)$ onde n é o número de regras.

Dicas de desempenho:

- Mantenha a contagem de regras abaixo de 100 para melhor desempenho
- Use prefixos específicos para reduzir a contagem de regras
- A regra padrão (prefixo vazio) deve ser a última

Benchmark (sistema típico):

- 10 regras: $< 1\mu s$ por consulta
- 50 regras: $< 5\mu s$ por consulta

- 100 regras: < 10µs por consulta

Uso de Memória

Cada regra requer ~100 bytes de memória:

- 10 regras ≈ 1 KB
- 100 regras ≈ 10 KB

Melhores Práticas

1. Sempre Inclua uma Regra de Recuo Curinga

```
gt_nat_rules: [  
    {%calling_prefix: "8772", weight: 10, response_gt: "111111"},  
    {%calling_prefix: "8773", weight: 10, response_gt: "222222"},  
    {%weight: 100, response_gt: "default_gt"} # Sempre tenha um  
curinga com peso alto  
]
```

2. Use Prefixos e Pesos Significativos

```
# BOM: Prefixos claros e específicos com pesos apropriados  
{%calling_prefix: "331", weight: 10, response_gt: "..."} # França  
{%calling_prefix: "44", weight: 10, response_gt: "..."} # Reino  
Unido  
  
# RUIM: Prefixos excessivamente amplos ou pesos confusos  
{%calling_prefix: "3", weight: 5, response_gt: "..."} # Muitos  
países  
{%calling_prefix: "331", weight: 100, response_gt: "..."} # Peso  
deve ser menor para regras específicas
```

3. Documente Suas Regras

```
gt_nat_rules: [  
    # Parceiro XYZ - rede do Reino Unido (faixa de GT: 4412xxxxxxx)  
    # Peso 10: Prioridade padrão do parceiro  
    {%calling_prefix: "4412", weight: 10, response_gt: "5551001"},  
  
    # Parceiro ABC - rede da França (faixa de GT: 33123xxxxxxx)  
    # Peso 10: Prioridade padrão do parceiro  
    {%calling_prefix: "33123", weight: 10, response_gt: "5551002"}  
]
```

4. Teste Antes da Implantação

```
# Teste no iex antes de implantar
iex> GtNat.translate_response_gt("44121234567", nil, "default")
"5551001" # Resultado esperado

# Teste com GT chamado
iex> GtNat.translate_response_gt("44121234567", "555123", "default")
"5551001" # Resultado esperado
```

5. Monitore os Logs

Ative o registro em nível INFO para ver todas as traduções do GT NAT em produção.

Integração com Outros Recursos

Modo STP

O GT NAT funciona independentemente do roteamento STP. O STP roteia com base em códigos de ponto e GTs de destino, enquanto o GT NAT lida com endereçamento de resposta.

Para mais informações sobre roteamento STP, veja o [Guia de Configuração STP](#).

Integração CAMEL

O GT NAT está **totalmente integrado** com operações CAMEL/CAP:

Camada SCCP:

- GT da Parte Chamadora em todas as respostas CAMEL
- Aplicado automaticamente com base no GT gsmSSF de entrada

Configuração:

- camel_gsmSCF_gt_address - GT padrão gsmSCF (opcional)
- Se não configurado, usa o GT da Parte Chamado da solicitação de entrada
- As regras de GT NAT substituem o padrão com base no prefixo da parte chamadora

Exemplo:

```
# Quando gsmSSF 555123456 chama seu gsmSCF
# Entrada: Chamado=55512341112, Chamando=555123456
# Consulta GT NAT: "555" -> response_gt="55512341111"
# Resposta: Chamado=555123456, Chamando=55512341111
```


Balanceamento de Carga

O GT NAT pode ser combinado com balanceamento de carga M3UA para gerenciamento avançado de tráfego.

Guia de Migração

Ativando o GT NAT em um Sistema Existente

1. Prepare a Configuração

```
# Adicione ao runtime.exs (mantenha desativado inicialmente)
config :omniss7,
  gt_nat_enabled: false, # Comece desativado
  gt_nat_rules: [
    # Suas regras aqui com pesos
    %{calling_prefix: "877", weight: 10, response_gt: "111111"},
    %{weight: 100, response_gt: "999999"} # Recuo curinga
  ]
```

2. Teste a Configuração

```
# Valide se a configuração compila
mix compile

# Teste no iex
iex -S mix
iex> GtNat.validate_config()
```

3. Ative em Staging

```
gt_nat_enabled: true # Mude para true
```

4. Monitore os Logs

```
tail -f log/omniss7.log | grep "GT NAT"
```

5. Implante em Produção

- Implemente durante a janela de manutenção
- Monitore as primeiras 24 horas de perto
- Tenha um plano de reversão pronto (defina `gt_nat_enabled: false`)

Suporte

Para problemas ou perguntas:

- Verifique os logs para mensagens "GT NAT"
- Valide a configuração com `GtNat.validate_config()`
- Revise a seção de resolução de problemas deste guia
- Entre em contato com o suporte OmniSS7 com trechos de log



Guia de Configuração do HLR

[— Voltar à Documentação Principal](#)

Este guia fornece a configuração para usar o OmniSS7 como um **Home Location Register (HLR/HSS)** com **OmniHSS** como o banco de dados de assinantes de backend.

Integração do OmniHSS

O modo HLR do OmniSS7 funciona como uma interface de sinalização SS7 que se conecta ao **OmniHSS**, um servidor de assinantes doméstico (HSS) completo. Esta arquitetura separa as preocupações:

- **OmniSS7 (Frontend HLR)**: Lida com toda a sinalização do protocolo SS7/MAP, roteamento SCCP e comunicação de rede
- **OmniHSS (Backend HSS)**: Gerencia dados de assinantes, autenticação, provisionamento e recursos avançados

Por que OmniHSS?

O OmniHSS fornece gerenciamento de assinantes de nível de operadora com recursos que incluem:

- **Suporte Multi-IMSI**: Cada assinante pode ter múltiplos IMSIs associados a um único MSISDN para roaming internacional, troca de rede e provisionamento de eSIM
- **Autenticação Flexível**: Suporte para algoritmos de autenticação Milenage (3G4G/5G) e COMP128 (2G)
- **Rastreamento de Sessão de Circuito & Pacote**: Rastreamento independente de registros de rede CS (circuit-switched) e PS (packet-switched)
- **Provisionamento Avançado**: Perfis de serviço personalizáveis, serviços suplementares e dados de assinatura CAMEL
- **Design API-First**: API HTTP RESTful para integração com sistemas de cobrança, CRM e provisionamento
- **Atualizações em Tempo Real**: Rastreamento de localização, gerenciamento de sessão e geração de vetores de autenticação

Todos os dados de assinantes, credenciais de autenticação e configurações de serviço são armazenados e gerenciados no OmniHSS. O OmniSS7 consulta o OmniHSS por meio de chamadas de API HTTPS para responder a operações MAP como UpdateLocation, SendAuthenticationInfo e SendRoutingInfo.

Importante: O modo HLR do OmniSS7 é um **frontend de sinalização apenas**. Toda a lógica de gerenciamento de assinantes, algoritmos de autenticação, regras de provisionamento e operações de banco de dados são tratadas pelo OmniHSS. Este guia cobre a configuração do protocolo SS7/MAP no OmniSS7. Para informações sobre provisionamento de assinantes, configuração de autenticação, perfis de serviço e operações administrativas, **consulte a documentação do OmniHSS**.

Suporte Multi-IMSI

O OmniHSS suporta nativamente configurações Multi-IMSI, permitindo que um único assinante (identificado por MSISDN) tenha múltiplos IMSIs. Isso possibilita:

- **Perfis de Roaming Internacional**: Diferentes IMSIs para diferentes regiões para reduzir custos de roaming
- **eSIM Multi-Perfil**: Múltiplos perfis de rede em um único dispositivo compatível com eSIM
- **Troca de Rede**: Troca sem costura entre redes sem mudar o MSISDN
- **Coordenação de Dual SIM**: Coordenação entre múltiplos SIMs físicos ou virtuais
- **Teste & Desenvolvimento**: Múltiplos IMSIs de teste apontando para o mesmo assinante

Como funciona:

- Cada IMSI tem suas próprias credenciais de autenticação (KI, OPc, algoritmo)
- Cada IMSI pode ter registros de sessão de circuito e pacote independentes
- Serviços e perfis de assinantes podem ser compartilhados ou personalizados por IMSI
- O OmniSS7 consulta o OmniHSS por IMSI, e o OmniHSS retorna os dados apropriados do assinante
- Sistemas de cobrança podem rastrear o uso por IMSI enquanto associam todos os IMSIs a uma única conta

Exemplo de cenário Multi-IMSI:

Assinante MSISDN: +1-555-123-4567
└ IMSI 1: 310260123456789 (Rede Doméstica dos EUA - autenticação Milenage)
└ IMSI 2: 208011234567890 (Perfil de Roaming da França - autenticação Milenage)
└ IMSI 3: 440201234567891 (Perfil de Roaming do Reino Unido - autenticação COMP128)

Todos os três IMSIs podem ser usados de forma independente para registro na rede, mas todos pertencem à mesma conta de assinante. O OmniHSS gerencia o mapeamento IMSI-para-assinante e garante a autenticação e o provisionamento adequados para cada IMSI.

OmniSS7 Stack v1.0.0

Download on Omnitouch
© 2025 Omnitouch

SS7 Events

SS7 Client

M3UA

HLR Links

Active Subscribers

Resources

Configuration

Active Subscribers

Total: 0 Auto-refresh every 2 seconds

Active Subscribers: Subscribers who have sent an UpdateLocation request and have not yet sent a CancelLocation.
This table shows the IMSI, current VLR, MSC, registration time, and duration for each active subscriber.

Clear All Subscribers

IMSI	VLR Number	MSC Number	Updated At	Duration
<div><div>No Active Subscribers</div><div>Subscribers will appear here when they send UpdateLocation requests.</div></div>				

Índice

1. [Integração do OmniHSS](#)
2. [Suporte Multi-IMSI](#)
3. [O que é o Modo HLR?](#)
4. [Habilitando o Modo HLR](#)
5. [Banco de Dados de Assinantes](#)
6. [Vetores de Autenticação](#)
7. [Atualizações de Localização](#)
8. [Integração CAMEL](#)
9. [Tratamento de Assinantes em Roaming](#)
10. [Operações HLR](#)
 - [Mapeamento de Campos de Resposta](#)
 - [SendRoutingInfo \(SRU\)](#)
 - [UpdateLocation \(LSU\)](#)
 - [SendRoutingInfoForSM](#)
 - [Resumo da Fonte de Campos](#)

O que é o Modo HLR?

Modo HLR permite que o OmniSS7 funcione como um Home Location Register para:

- **Gerenciamento de Assinantes:** Armazenar e gerenciar dados de assinantes
- **Autenticação:** Gerar vetores de autenticação para acesso à rede
- **Rastreamento de Localização:** Processar atualizações de localização de VLRs
- **Informações de Roteamento:** Fornecer informações de roteamento para chamadas e SMS

Arquitetura HLR

Habilitando o Modo HLR

O OmniSS7 pode operar em diferentes modos. Para usá-lo como um HLR, você precisa habilitar o modo HLR na configuração.

Mudando para o Modo HLR

O config/runtime.exe do OmniSS7 contém três modos operacionais pré-configurados. Para habilitar o modo HLR:

1. Abra config/runtime.exe
2. **Encontre** as três seções de configuração (linhas 53-174):
 - Configuração 1: Modo STP (linhas 53-85)
 - Configuração 2: Modo HLR (linhas 87-123)
 - Configuração 3: Modo SMSs (linhas 125-174)
3. **Comente** a configuração atualmente ativa (adicione # a cada linha)
4. **Descomente** a configuração HLR (remova # das linhas 87-123)
5. **Personalize** os parâmetros de configuração conforme necessário
6. **Reinicie** a aplicação: iex -S mix

Configuração do Modo HLR

A configuração completa do HLR fica assim:

```
config :omniSS7,
# Flags de modo - Habilitar recursos HLR apenas
map_client_enabled: true,
hlr_mode_enabled: true,
smsc_mode_enabled: false,

# Configuração da API Backend do OmniHSS
hlr_api_base_url: "https://10.180.2.140:8443",

# Endereço GT do Centro de Serviço HLR para operações de SMS
hlr_service_center_gt_address: "1234567890",

# Configuração de Mapeamento MSISDN - IMSI
# Veja: seção de Mapeamento MSISDN - IMSI para detalhes
hlr_imsi_plan_prefix: "50557",
hlr_msisd_n_country_code: "61",
hlr_msisd_n_nsn_offset: 0,
hlr_msisd_n_nsn_length: 9,

# Configuração de InsertSubscriberData
# Modo de Acesso à Rede: :packetAndCircuit, :packetOnly, ou :circuitOnly
isd_network_access_mode: :packetAndCircuit,

# Enviar ISD #2 (dados de Serviços Suplementares)
isd_send_ss_data: true,

# Enviar ISD #3 (dados de Bloqueio de Chamadas)
isd_send_call_barring: true,

# Configuração CAMEL (para respostas SendRoutingInfo)
# Chave de Serviço para iniciação de serviço CAMEL
camel_service_key: 11_110,

# Ponto de Detecção de Gatilho CAMEL
# Opções: :termAttemptAuthorized, :tBusy, :tNoAnswer, :tAnswer
camel_trigger_detection_point: :termAttemptAuthorized,

# Prefixos VLR Doméstico
# Lista de prefixos de endereço VLR que são considerados "rede" doméstica
# Se o VLR do assinante começar com um desses prefixos, use a resposta SRI padrão
# Caso contrário, o assinante está em roaming e precisamos enviar PRN para obter MSRN
home_vlr_prefixes: ["123456"],

# Configuração de Conexão M3UA
# Conectar como ASP para receber operações MAP (UpdateLocation, SendAuthInfo, etc.)
map_client_m3ua: %{
  mode: "ASP",
  callback: (MapClient, :handle_payload, []),
  process_name: :hlr_client_esp,
  # Endpoint local (sistema HLR)
  local_ip: {10, 179, 4, 11},
  local_port: 2905,
  # Endpoint remoto STP
  remote_ip: {10, 179, 4, 10},
  remote_port: 2905,
  routing_context: 1
}
```

OmniSS7 Stack

SS7 Events

SS7 Client

M3UA

HLR Links

Active

Subscribers

Resources

Configuration

HLR API Connection Status

HLR HTTP API

Latency

46.14 ms

Version/Role

—

Host

https://10.180.2.140:8443

Details

{:tls_alert, {:bad_certificate, ~c"TLS client: In state wait_cert_cr at ssl_handshake.erl:2178 generated CLIENT ALERT: Fatal - Bad Certificate\n selfsigned_peer"}}}

DOWN

Subscriber Lookup

Lookup Type

IMSI

IMSI Value

e.g. 001010000000001

Lookup

Generate Authentication Vectors

IMSI

e.g. 001010000000001

Generate

OmniTouch

Parâmetros de Configuração e Personalizar

Para uma referência completa de todos os parâmetros de configuração, consulte a [Referência de Configuração](#).

Parâmetro	Tipo	Padrão	Descrição	Exemplo
hlr_api_base_url	String	Obrigatório	Endpoint da API do OmniHSS	"https://10.179.3.219:8443"
hlr_service_center_gt_address	String	Obrigatório	Endereço GT do HLR usado nas respostas UpdateLocation	"5551234568"
smc_service_center_gt_address	String	Obrigatório	Endereço GT do SMSC retornado nas respostas SRI-for-SM	"5551234567"
hlr_smc_alert_gts	Lista	1	Lista de GTs do SMSc para enviar alertServiceCenter após UpdateLocation	["15559876543", "15559876544"]
hlr_alert_location_expiry_seconds	Inteiro	172800	Tempo de expiração da localização (segundos) quando o SMSc recebe alertServiceCenter	86400
hlr_imsi_plmn_prefix	String	"50557"	Prefixo PLMN (MCC+MNC) para mapeamento MSISDN-IMSI (veja MSISDN == IMSI Mapping)	"001001"
hlr_msisdn_country_code	String	"61"	Prefixo do código do país para mapeamento reverso IMSI-MSISDN (veja MSISDN == IMSI Mapping)	"1"
hlr_msisdn_nsn_offset	Inteiro	0	Deslocamento no MSISDN para extração do NSN (veja MSISDN == IMSI Mapping)	0
hlr_msisdn_nsn_length	Inteiro	9	Comprimento do Número de Assinante Nacional a ser extraído (veja MSISDN == IMSI Mapping)	10
isd_network_access_mode	Atomo	:packetAndCircuit	Modo de acesso à rede para InsertSubscriberData	:packetOnly
isd_send_ss_data	Booleano	true	Enviar ISD #2 com dados de Serviços Suplementares	false
isd_send_call_barring	Booleano	true	Enviar ISD #3 com dados de Bloqueio de Chamadas	false
camel_service_key	Inteiro	11_110	Chave de serviço CAMEL para respostas SendRoutingInfo	100
camel_trigger_detection_point	Atomo	:termAttemptAuthorized	Ponto de detecção de gatilho CAMEL	:tBusy
home_vlr_prefixes	Lista	["5551231"]	Lista de prefixos de endereço VLR considerados "rede" doméstica	["555123"]
local_ip	Tupla	Obrigatório	Endereço IP do seu sistema HLR	{10, 179, 4, 12}
local_port	Inteiro	2905	Porta SCTP local	2905
remote_ip	Tupla	Obrigatório	Endereço IP do STP para conectividade SS7	{10, 179, 4, 10}
remote_port	Inteiro	2905	Porta SCTP remota	2905
routing_context	Inteiro	1	ID do contexto de roteamento M3UA	1

O que Acontece Quando o Modo HLR é Habilitado

Quando hlr_mode_enabled: true, a interface web mostrará:

- **Eventos SS7** - Registro de eventos
- **Cliente SS7** - Teste de operações MAP
- **M3UA** - Status da conexão
- **Links HLR** - Status da API HLR + gerenciamento de assinantes – *Específico do HLR*
- **Recursos** - Monitoramento do sistema
- **Configuração** - Visualizador de configuração

As abas **Roteamento**, **Teste de Roteamento** e **Links SMSc** serão ocultadas.

Notas Importantes

- **Configuração Obrigatória:** O parâmetro hlr_service_center_gt_address é **mandatório**. A aplicação falhará ao iniciar se não estiver configurado.
- **Backend OmniHSS:** A API do OmniHSS deve ser acessível no hlr_api_base_url configurado
- **Tempo Limite de Solicitação da API:** Todas as solicitações da API do OmniHSS têm um **tempo limite fixo de 5 segundos**
- **Tempo Limite de Solicitação MAP:** Todas as solicitações MAP (SRI, UpdateLocation, SendAuthInfo, etc.) têm um **tempo limite fixo de 10 segundos**
- **Tempo Limite de ISD:** Cada mensagem InsertSubscriberData (ISD) em uma sequência UpdateLocation tem um **tempo limite fixo de 10 segundos**
- A conexão M3UA com o STP é necessária para receber operações MAP
- Após mudar de modos, você deve reiniciar a aplicação para que as alterações tenham efeito
- **Interface Web:** Consulte o [Guia da Interface Web](#) para informações sobre como usar a interface web
- **Acesso à API:** Consulte o [Guia da API](#) para documentação da API REST e acesso ao Swagger UI

Banco de Dados de Assinantes

O OmniHSS gerencia todos os dados de assinantes incluindo identidades, credenciais de autenticação, perfis de serviço e informações de localização. O OmniSS7 recupera esses dados por meio de chamadas de API RESTful.

Modelo de Assinante do OmniHSS

O OmniHSS armazena informações abrangentes sobre assinantes:

- **Múltiplos IMSIs por assinante:** Suporte para configurações Multi-IMSI (eSIM, perfis de roaming, troca de rede)
- **Credenciais de autenticação:** Seleção de KI, OPc e algoritmo (Milenage ou COMP128)
- **Perfis de serviço:** Categoria do assinante, serviços permitidos, parâmetros de QoS
- **Rastreamento de localização:** Rastreamento independente da localização atual VLR/MSC (sessão de circuito) e SGSN/GGSN (sessão de pacote)
- **Dados de assinatura CAMEL:** Chaves de serviço, pontos de gatilho e endereços gsmSCF
- **Serviços suplementares:** Encaminhamento de chamadas, bloqueio, espera, configurações CLIP/CLIR
- **Estado administrativo:** Habilitado/desabilitado, restrições de serviço, datas de expiração

Vetores de Autenticação

Gerar Vetores de Autenticação

O OmniHSS gera vetores de autenticação usando os algoritmos Milenage ou COMP128 com base no método de autenticação configurado de cada assinante. Quando o OmniSS7 recebe solicitações MAP sendAuthenticationInfo:

1. O OmniSS7 extrai o IMSI da solicitação MAP
2. O OmniSS7 chama a API do OmniHSS para gerar vetores de autenticação
3. O OmniHSS recupera as credenciais KI e OPc do assinante
4. O OmniHSS gera o número solicitado de vetores (RAND, XRES, CK, IK, AUTN)
5. O OmniSS7 codifica os vetores no formato MAP e os retorna ao VLR/SGSN solicitante

Integração da API do OmniHSS

O OmniSS7 se comunica com o OmniHSS via API REST HTTPS para recuperar informações do assinante, atualizar dados de localização e gerar vetores de autenticação:

```
config :omniSS7,  
  hlr_api_base_url: "https://omnihss-server:8443"
```

Quando o OmniSS7 recebe operações MAP da rede SS7, ele consulta o OmniHSS para:

- **Recuperar dados do assinante** por IMSI ou MSISDN
- **Gerar vetores de autenticação** usando credenciais KJ/OPc armazenadas
- **Atualizar a localização da sessão de circuito** quando os assinantes realizam UpdateLocation
- **Verificar o status do assinante** e os direitos de serviço

Atualizações de Localização

Processamento de Atualização de Localização

Ao receber solicitações MAP **updateLocation**, o OmniSS7 coordena com o OmniHSS para registrar o assinante em um novo VLR:

1. **Extraí informações de localização** da solicitação UpdateLocation (IMSI, novo GT VLR, novo GT MSC)
2. **Consultar o OmniHSS** para verificar se o assinante existe e está habilitado
3. **Atualizar a sessão de circuito** no OmniHSS com a nova localização VLR/MSC
4. **Enviar mensagens InsertSubscriberData (ISD)** para provisionar o assinante no novo VLR
5. **Retornar a resposta UpdateLocation** ao VLR (inclui GT HLR do hlr_service_center_gt_address)
6. **Enviar alertServiceCenter** para os GTs do SMSc configurados (se hlr_smsc_alert_gts estiver populado)

Nota: O parâmetro de configuração hlr_service_center_gt_address especifica o Título Global do HLR que é retornado nas respostas UpdateLocation. Isso permite que o VLR/MSC identifique e roteie mensagens de volta para este HLR.

Integração do Centro de Serviço de Alerta

Após uma atualização de localização bem-sucedida, o HLR pode notificar automaticamente os sistemas SMSc que um assinante agora está acessível, enviando mensagens **alertServiceCenter** (opcode MAP 64). Para informações sobre como o SMSc lida com esses alertas, consulte [Tratamento do Centro de Serviço de Alerta no Guia do SMSc](#).

Configuração

Configure a lista de Títulos Globais do SMSc a serem notificados:

```
config :omniSS7,  
  # Lista de GTs do SMSc para enviar alertServiceCenter após UpdateLocation  
  hlr_smsc_alert_gts: [  
    "15559878543",  
    "15559878544"  
  ],  
  # Tempo de expiração da localização quando o SMSc recebe alertServiceCenter (padrão: 48 horas)  
  hlr_alert_location_expiry_seconds: 172800
```

Diagrama de Fluxo

Comportamento

Quando um assinante realiza UpdateLocation:

1. O HLR envia alertServiceCenter para **cada** GT do SMSc na lista hlr_smsc_alert_gts
2. A mensagem inclui o MSISDN do assinante
3. O HLR usa hlr_service_center_gt_address como o GT da parte chamadora
4. Endereçamento SCCP: ssn chamador=6 (HLR), ssn chamado=8 (SMSc)

O SMSc recebe o alerta e:

- **Remove o prefixo TON/NPI** do MSISDN (por exemplo, "19123123213" => "123123213")
- Marca o assinante como acessível em seu banco de dados de localização (via POST para [api/location](#))
- **Define o campo user_agent** para o GT do HLR ao chamar a API (para rastrear qual HLR enviou o alerta)
- Define o tempo de expiração da localização com base em hlr_alert_location_expiry_seconds
- Rastreia o assinante no SMSc Subscriber Tracker para monitoramento

Testes

Use a página **Assinantes Ativos** na interface web para enviar manualmente mensagens alertServiceCenter para testes:

1. Navegue até a aba "Assinantes Ativos"
2. Encontre a seção "Testar Centro de Serviço de Alerta"
3. Insira MSISDN, GT do SMSc e GT do HLR (os padrões são preenchidos a partir da configuração)
 - GT do SMSc padrão é a primeira entrada em hlr_smsc_alert_gts
 - GT do HLR padrão é hlr_service_center_gt_address
4. Clique em "Enviar alertServiceCenter"

Isso é útil para testar o tratamento de alertas do SMSc sem exigir um fluxo completo de UpdateLocation. O formulário usa validação phx-blur para evitar mostrar erros enquanto digita.

Configuração do InsertSubscriberData (ISD)

Após uma atualização de localização bem-sucedida, o HLR envia dados de provisionamento do assinante para o VLR usando mensagens **InsertSubscriberData** (ISD). A configuração ISD permite que você personalize quais dados são enviados e como.

Para referência de parâmetros de configuração, consulte [Configuração ISD na Referência de Configuração](#).

Sequência ISD

O HLR pode enviar até 3 mensagens ISD sequenciais:

1. **ISD #1** (Sempre enviado) - Dados básicos do assinante:
 - IMSI
 - MSISDN
 - Categoria do assinante
 - Status do assinante (serviceGranted)
 - Lista de serviços de portadora
 - Lista de teleseleções
 - Modo de acesso à rede
2. **ISD #2** (Opcional) - Dados de Serviços Suplementares (SS):
 - Configurações de encaminhamento de chamadas (incondicional, ocupado, sem resposta, não acessível)
 - Espera de chamadas
 - Manter chamada
 - Serviço de múltiplos participantes
 - Status e recursos de serviços suplementares
3. **ISD #3** (Opcional) - Dados de Bloqueio de Chamadas:
 - Bloqueio de todas as chamadas de saída (BAOC)
 - Bloqueio de chamadas internacionais de saída (BOIC)
 - Dados de restrição de acesso

Opções de Configuração

```
# Configuração do InsertSubscriberData  
# Modo de Acesso à Rede: :packetAndCircuit, :packetOnly, ou :circuitOnly  
isd_network_access_mode: :packetAndCircuit,  
  
# Enviar ISD #2 (dados de Serviços Suplementares)  
isd_send_ss_data: true,  
  
# Enviar ISD #3 (dados de Bloqueio de Chamadas)  
isd_send_call_barring: true,
```

Modo de Acesso à Rede

O parâmetro isd_network_access_mode controla que tipo de acesso à rede o assinante está autorizado:

Valor	Descrição	Caso de Uso
:packetAndCircuit	Tanto packet-switched (GPRS/LTE) quanto circuit-switched (voz)	Padrão - Assinantes de serviço completo
:packetOnly	Apenas packet-switched (dados/LTE)	Cartões SIM apenas para dados, dispositivos IoT
:circuitOnly	Apenas circuit-switched (voz/SMS)	Dispositivos legados, planos apenas de voz

Controlando Mensagens ISD

Você pode controlar quais mensagens ISD são enviadas com base em suas necessidades de rede:

Enviar todos os ISDs (Padrão - Conjunto completo de recursos):

```
isd_send_ss_data: true,  
isd_send_call_barring: true,
```

Enviar apenas dados básicos do assinante (Provisionamento mínimo):

```
isd_send_ss_data: false,  
isd_send_call_barring: false,
```

Enviar básico + serviços suplementares (Sem bloqueio de chamadas):

```
isd_send_ss_data: true,  
isd_send_call_barring: false,
```

Exemplo de Fluxo ISD

Quando UpdateLocation é recebido:

```
HLR → HLR: UpdateLocation (INÍCIO)  
HLR → VLR: InsertSubscriberData #1 (CONTINUAR) - Dados básicos  
VLR → HLR: ISD #1 ACK (CONTINUAR)
```

HLR → VLR: InsertSubscriberData #2 (CONTINUAR) - Dados SS [se habilitado]
VLR → HLR: ISD #2 ACK (CONTINUAR)
HLR → VLR: InsertSubscriberData #3 (CONTINUAR) - Bloqueio de chamadas [se habilitado]
VLR → HLR: ISD #3 ACK (CONTINUAR)
HLR → VLR: Resposta UpdateLocation (FIM)

Se isd_send_ss_data ou isd_send_cal_barring forem definidos como false, essas mensagens ISD são puladas, e o UpdateLocation FIM é enviado mais cedo.

Melhores Práticas

- **Configuração Padrão:** Use :packet&ndCircuit e habilite todos os ISDs para máxima compatibilidade
- **IoT/M2M:** Use :packetOnly e desative dados SS/bloqueio de chamadas para dispositivos apenas para dados
- **Interoperabilidade:** Alguns VLRs mais antigos podem não suportar todos os serviços suplementares - desative isd_send_ss_data se encontrar problemas
- **Desempenho:** Desativar ISDs não utilizados reduz a sobrecarga de mensagens e acelera as atualizações de localização

Integração CAMEL

Configuração CAMEL para SendRoutingInfo

Ao responder a solicitações **SendRoutingInfo** (SRI) de um GMSC (Gateway MSC), o HLR pode instruir o GMSC a invocar serviços CAMEL para roteamento inteligente de chamadas e controle de serviços.

Para referência de parâmetros de configuração, consulte [Configuração CAMEL na Referência de Configuração](#).

O que é CAMEL?

CAMEL (Aplicações Personalizadas para Lógica Aprimorada de Redes Móveis) é um protocolo que permite serviços de rede inteligentes em redes GSM/UMTS. Ele permite que os operadores de rede implementem serviços de valor agregado, como:

- Cobrança pré-paga
- Filtragem e bloqueio de chamadas
- Redes Privadas Virtuais (VPN)
- Serviços de taxa premium
- Encaminhamento de chamadas com lógica personalizada
- Serviços baseados em localização

Opções de Configuração

```
# Configuração CAMEL (para respostas SendRoutingInfo)
# Chave de Serviço para iniciação de serviço CAMEL
camel_service_key: 11_110,

# Ponto de Detecção de Gatilho CAMEL
# Opções: :termAttemptAuthorized, :tBusy, :tNoAnswer, :tAnswer
camel_trigger_detection_point: :termAttemptAuthorized,
```

Chave de Serviço

A camel_service_key identifica qual serviço CAMEL deve ser invocado no gsmSCF (Função de Controle de Serviço). Este é um identificador numérico configurado em sua rede:

Chave de Serviço	Caso de Uso Típico
11_110	Controle de chamada pré-paga (padrão)
100	Serviço pré-pago de origem
200	Encaminhamento de chamadas com lógica personalizada
300	Rede Privada Virtual (VPN)
Personalizado	Serviços específicos do operador

Exemplo de Configuração:

```
# Para controle de chamada pré-paga
camel_service_key: 11_110,
```

```
# Para serviço VPN
camel_service_key: 300,
```

Ponto de Detecção de Gatilho

O camel_trigger_detection_point especifica quando o serviço CAMEL deve ser acionado durante a configuração da chamada:

Ponto de Detecção	Descrição	Quando Acionado
:termAttemptAuthorized	Tentativa de chamada autorizada (padrão)	Antes que a chamada seja roteada para o assinante
:tBusy	Ocupado na terminação	Quando o assinante está ocupado
:tNoAnswer	Sem resposta na terminação	Quando o assinante não atende
:tAnswer	Resposta na terminação	Quando o assinante atende a chamada

Exemplos de Configuração:

Controle pré-pago padrão (acionar antes do roteamento):

```
camel_trigger_detection_point: :termAttemptAuthorized,
```

Tratamento de ocupado personalizado (acionar quando ocupado):

```
camel_trigger_detection_point: :tBusy,
```

Cobrança baseada em resposta (acionar na resposta):

```
camel_trigger_detection_point: :tAnswer,
```

Resposta SRI com CAMEL

Quando configurado, as respostas SendRoutingInfo incluem informações de assinatura CAMEL:

```
GMSC → HLR: SendRoutingInfo (INICIO)
HLR → GMSC: Resposta SRI (FIM) com:
- IMSI
- Número VLR
- Estado do assinante
- Informações de roteamento CAMEL:
  * Chave de Serviço: 11_110
  * Endereço gsmSCF: <endereço configurado>
  * Ponto de Detecção de Gatilho: termAttemptAuthorized
  * Tratamento de Chamada Padrão: continueCall
```

GMSC contata gsmSCF no ponto de gatilho para executar o serviço CAMEL

Melhores Práticas

- **Redes de Produção:** Use chaves de serviço padronizadas acordadas com seu provedor gsmSCF
- **Testes:** Use :termAttemptAuthorized para testes mais abrangentes
- **Serviços Pré-pagos:** A chave de serviço 11_110 é um padrão comum da indústria para chamadas pré-pagas de terminação
- **Tratamento de Fallback:** defaultCallHandling: :continueCall garante que as chamadas prossigam se o gsmSCF estiver inacessível

Tratamento de Assinantes em Roaming

Detecção de VLR Doméstico vs VLR em Roaming

Quando o HLR recebe uma solicitação **SendRoutingInfo** (SRI), ele precisa determinar se o assinante está em um VLR "doméstico" (dentro da sua rede) ou em um VLR em roaming (visitando outra rede). O comportamento difere com base nessa determinação:

Para referência de parâmetros de configuração, consulte [Prefixos VLR Doméstico na Referência de Configuração](#).

- **VLR Doméstico:** Retornar resposta SRI padrão com informações de roteamento CAMEL
- **VLR em Roaming:** Enviar uma solicitação Provide Roaming Number (PRN) para obter um MSRN, e então retorná-lo na resposta SRI

Configuração

```
# Prefixos VLR Doméstico
# Lista de prefixos de endereço VLR que são considerados "rede" doméstica
# Se o endereço VLR do assinante começar com um desses prefixos, use a resposta SRI padrão
# Caso contrário, o assinante está em roaming e precisamos enviar PRN para obter MSRN
home_vlr_prefixes: ["555123"],
```

Exemplo de Configuração:

```
# Único operador de rede doméstica
home_vlr_prefixes: ["555123"],
```

```
# Múltiplos operadores de rede doméstica (por exemplo, diferentes regiões ou subsidiárias)
home_vlr_prefixes: ["555123", "555124", "555125"],
```

Como Funciona

1. Fluxo de Assinante Doméstico (Padrão)

Quando o endereço VLR do assinante começa com um prefixo doméstico configurado:

```
GMSC → HLR: SendRoutingInfo (MSISDN: "1234567890")
HLR consulta a API de backend para dados do assinante
HLR verifica o endereço VLR: "5551234567"
HLR determina: VLR começa com "555123" → Rede doméstica
HLR → GMSC: Resposta SRI com informações de roteamento CAMEL:
- IMSI
- Número VLR: "5551234567"
- Endereço gsmSCF (MSC): "5551234501"
- Chave de serviço CAMEL: 11_110
- Ponto de detecção de gatilho: termAttemptAuthorized
```

2. Fluxo de Assinante em Roaming (PRN Necessário)

Quando o endereço VLR do assinante NÃO corresponde a nenhum prefixo doméstico:

GMSC → HLR: SendRoutingInfo (MSISDN: "1234567890")
HLR consulta a API de backend para dados do assinante
HLR verifica o endereço VLR: "49170123456"
HLR determina: VLR não começa com "555123" → Roaming
HLR → MSC: ProvideRoamingNumber (PRN):
- MSISDN: "1234567890"
- IMSI: "999999876543210"
- Número MSC: "49170123456"
- Endereço GMSC: "5551234501"
MSC → HLR: Resposta PRN com MSRN: "49170999888777"
HLR → GMSC: Resposta SRI com informações de roteamento:
- IMSI
- Número VLR: "49170123456"
- Número em Roaming (MSRN): "49170999888777"

Diferenças na Estrutura da Resposta

Resposta SRI de Assinante Doméstico

```
{
  "imsi": "999999876543210",
  "extendedRoutingInfo": {
    "camelRoutingInfo": {
      "gsmCamelSubscriptionInfo": {
        "t-CSI": {
          "serviceKey": 11,
          "gsmSCF-Address": "5551234501",
          "defaultCallHandling": "continueCall",
          "t-BcmTriggerDetectionPoint": "itemAttemptAuthorized"
        }
      }
    }
  },
  "subscriberInfo": {
    "locationInformation": {
      "vlr-number": "5551234567",
      "subscriberState": "notProvidedFromVLR"
    }
  }
}
```

Resposta SRI de Assinante em Roaming

```
{
  "imsi": "999999876543210",
  "extendedRoutingInfo": {
    "routingInfo": {
      "roamingNumber": "49170999888777"
    }
  },
  "subscriberInfo": {
    "locationInformation": {
      "vlr-number": "49170123456",
      "subscriberState": "notProvidedFromVLR"
    }
  }
}
```

Operação Provide Roaming Number (PRN)

Estrutura da Solicitação PRN

A solicitação PRN enviada ao MSC/VLR contém:

MSISDN	Campo	Fonte	Descrição
IMSI		Solicitação SRI	Número de telefone do assinante
Número MSC		API HLR	IMSI do assinante
Endereço GMSC		API HLR	MSC que atende o assinante em roaming (serving_msc)
Número de Referência de Chamada		Solicitação SRI	GMSC que faz a solicitação SRI original
Fases CAMEL Suportadas		Estático	Identificador de referência de chamada
		Estático	Fases CAMEL suportadas pelo GMSC

Tratamento da Resposta PRN

O HLR espera uma resposta PRN contendo:

- **MSRN** (Número de Roaming da Estação Móvel): Um número temporário alocado pela rede visitada para roteamento da chamada

Tratamento de Erros:

- Se o PRN expirar → Retorna erro 27 (Assinante Ausente) na resposta SRI
- Se o PRN falhar → Retorna erro 27 (Assinante Ausente) na resposta SRI
- Se o MSRN não puder ser extraído → Retorna erro 27 (Assinante Ausente) na resposta SRI

Exemplos de Configuração

Operador de Rede Doméstica Único

Todos os endereços VLR começando com "555123" são considerados domésticos
home_vlr_prefixes: ["555123"],

- VLR 5551234567 → Doméstico (resposta CAMEL)
- VLR 5551235001 → Doméstico (resposta CAMEL)
- VLR 49170123456 → Roaming (PRN + resposta MSRN)

Operador Multi-Região

Múltiplas redes domésticas em diferentes regiões
home_vlr_prefixes: ["555123", "555124", "555125"],

- VLR 5551234567 → Doméstico (região 1)
- VLR 5552341234 → Doméstico (região 2)
- VLR 5553411111 → Doméstico (região 3)
- VLR 44201234567 → Roaming (internacional)

Configuração de Teste

Para testar a funcionalidade PRN, defina uma lista vazia para tratar todos os VLRs como roaming:

Todos os VLRs são tratados como roaming (para testar o fluxo PRN)
home_vlr_prefixes: [],

Melhores Práticas

- **Seleção de Prefixo:** Use o prefixo único mais curto que identifique os VLRs da sua rede (por exemplo, código do país + código da rede)
- **Múltiplos Prefixos:** Inclua todos os prefixos de VLR na sua rede, incluindo diferentes regiões e subsidiárias
- **Acordos de Roaming:** Certifique-se de que o PRN seja adequadamente suportado pelas redes parceiras de roaming
- **Testes:** Teste minuciosamente os cenários domésticos e em roaming antes da implantação em produção
- **Monitoramento:** Monitore as taxas de tempo limite do PRN para identificar problemas de conectividade com parceiros de roaming

Solução de Problemas

Sintoma: Todos os assinantes tratados como roaming

- **Causa:** home_vlr_prefixes não configurado ou prefixos não correspondem aos endereços VLR
- **Solução:** Verifique os endereços VLR em seu banco de dados e atualize os prefixos conforme necessário

Sintoma: Solicitações PRN expiram

- **Causa:** Problemas de conectividade de rede com o MSC/VLR parceiro de roaming
- **Solução:** Verifique o roteamento M3UA/SCCP para endereços MSC remotos

Sintoma: MSRN inválido na resposta SRI

- **Causa:** Formato da resposta PRN do parceiro de roaming não corresponde à estrutura esperada
- **Solução:** Revise os logs de resposta PRN e ajuste extract_msrn_from_prn/1 se necessário

Operações HLR

Operações MAP Suportadas

- updateLocation (Opcode 2) - Registrar localização VLR
- sendAuthenticationInfo (Opcode 56) - Gerar vetores de autenticação
- sendRoutingInfo (Opcode 22) - Fornecer MSRN para chamadas com suporte CAMEL
- sendRoutingInfoForSM (Opcode 45) - Fornecer GT MSC para SMS
- cancelLocation (Opcode 3) - Cancelar registro no VLR antigo
- insertSubscriberData (Opcode 7) - Enviar perfil do assinante

Mapeamento de Campos de Resposta

Esta seção detalha de onde cada campo nas respostas do HLR vem.

Resposta SendRoutingInfo (SRI)

Propósito: Fornece informações de roteamento para chamadas recebidas para um assinante.

O HLR fornece dois tipos diferentes de resposta com base em se o assinante está em um VLR doméstico ou em roaming:

Resposta de Assinante Doméstico (Roteamento CAMEL)

Usado quando o endereço VLR do assinante começa com um valor configurado em home_vlr_prefixes.

Estrutura da Resposta:

Campo	Fonte	Descrição	Exemplo
IMSI	API OmniHSS	IMSI do assinante do banco de dados OmniHSS	"999999876543218"
Número VLR	API OmniHSS	VLR atual atendendo o assinante (circuit_session.assigned_vlr)	"5551234567"
Estado do Assinante	Estático	Sempre notProvidedFromVLR	:notProvidedFromVLR
extendedRoutingInfo	-	Tipo: camelRoutingInfo	-
Endereço gsmSCF	API OmniHSS	MSC atendendo o assinante (circuit_session.assigned_msc)	"5551234501"
Chave de Serviço	runtime.exe	Identificador de serviço CAMEL (camel_service_key)	11, 110
Ponto de Detecção de Gatilho	runtime.exe	Quando acionar CAMEL (camel_trigger_detection_point)	:termAttemptAuthorized
Tratamento de Capacidade CAMEL	Estático	Nível de suporte à fase CAMEL	3
Tratamento de Chamada Padrão	Estático	Fallback se gsmSCF inacessível	:continueCall

Resposta de Assinante em Roaming (Roteamento MSRN)

Usado quando o endereço VLR do assinante NÃO corresponde a nenhum valor configurado em home_vlr_prefixes.

Estrutura da Resposta:

Campo	Fonte	Descrição	Exemplo
IMSI	API OmniHSS	IMSI do assinante do banco de dados OmniHSS	"999999876543218"
Número VLR	API OmniHSS	VLR atual atendendo o assinante (circuit_session.assigned_vlr)	"49170123456"
Estado do Assinante	Estático	Sempre notProvidedFromVLR	:notProvidedFromVLR
extendedRoutingInfo	-	Tipo: routingInfo	-
Número em Roaming (MSRN)	Resposta PRN MSRN obtido da solicitação ProvideRoamingNumber		"49170999888777"

Lógica de Decisão de Roteamento:

- OmniSS7 recebe solicitação SendRoutingInfo
- OmniSS7 consulta dados do assinante da API OmniHSS
- OmniSS7 verifica o endereço VLR contra home_vlr_prefixes:

Se VLR começa com prefixo doméstico:
- Retornar informações de roteamento CAMEL (fluxo de assinante doméstico)

Se VLR NÃO corresponder a nenhum prefixo doméstico:
- Enviar ProvideRoamingNumber (PRN) para MSC
- Extrair MSRN da resposta PRN
- Retornar informações de roteamento com MSRN (fluxo de assinante em roaming)

Fluxo de Dados:

- OmniSS7 consulta o OmniHSS para informações do assinante
- OmniHSS retorna IMSI, localização atual VLR/MSC e estado do assinante
- OmniSS7 usa esses dados para construir a resposta MAP

Requisitos de Configuração:

```
# Em runtime.exe
home_vlr_prefixes: ["555123"], # Lista de prefixos VLR domésticos
```

Respostas de Erro:

- Se serving_vlr e serving_msc forem null: Retorna erro 27 (Assinante Ausente)
- Se assinante não encontrado: Retorna erro 1 (Assinante Desconhecido)
- Se a solicitação PRN expirar (caso de roaming): Retorna erro 27 (Assinante Ausente)
- Se a resposta PRN for inválida (caso de roaming): Retorna erro 27 (Assinante Ausente)

Resposta UpdateLocation com InsertSubscriberData

Propósito: Registra o assinante em um novo VLR e provisiona os dados do assinante.

Resposta FIM UpdateLocation

Campo	Fonte	Descrição	Exemplo
Número HLR	runtime.exe	Título Global deste HLR (vlr_service_center_gt_address)	"5551234568"
Tipo de Mensagem TCAP	Estático	Resposta final após todos os ISDs	FIM

InsertSubscriberData #1 (Dados Básicos do Assinante)

Campo	Fonte	Descrição	Exemplo
IMSI	Solicitação	Da solicitação UpdateLocation	"999999876543218"
MSISDN	API OmniHSS	Número de telefone do assinante do OmniHSS	"555123456"
Categoria	Estático	Categoria do assinante	"v" (VoDA)
Status do Assinante	Estático	Status do serviço	:serviceGranted
Lista de Serviços de Portadora	Estático	Serviços de portadora suportados	[<<31>>]
Lista de Teleserviços	Estático	Teleserviços suportados	[<<17>>, "1", "\"]
Modo de Acesso à Rede	runtime.exe	Acesso a pacotes/circuitos (isd_network_access_mode)	:packetAndCircuit

InsertSubscriberData #2 (Serviços Suplementares) - Opcional

Campo	Fonte	Descrição	Controlado Por
SS Provisionado	Estático	Dados de serviços suplementares	isd_send_ss_data: true
Encaminhamento de Chamadas	Estático	Configurações de encaminhamento (incondicional, ocupado, sem resposta, não acessível)	Config habilitado
Espera de Chamadas	Estático	Status do serviço de espera de chamadas	Config habilitado
Serviço de Múltiplos Participantes	Estático	Suporte a chamadas de conferência	Config habilitado

ISD #2 inclui:

- Encaminhamento de chamadas incondicional (código SS 21)
- Encaminhamento de chamadas em ocupado (código SS 41)
- Encaminhamento de chamadas sem resposta (código SS 42)
- Encaminhamento de chamadas não acessíveis (código SS 62)
- Espera de chamadas (código SS 43)
- Serviço de múltiplos participantes (código SS 51)
- Serviços CLIP/CLIR

InsertSubscriberData #3 (Bloqueio de Chamadas) - Opcional

Campo	Fonte	Descrição	Controlado Por
Informações de Bloqueio de Chamadas	Estático	Configurações de bloqueio de chamadas	isd_send_call_barring: true
BACD	Estático	Bloqueio de Todas as Chamadas de Saída (código SS 146)	Config habilitado
BOIC	Estático	Bloqueio de Chamadas Internacionais de Saída (código SS 147)	Config habilitado
Dados de Restrição de Acesso	Estático	Restrições de acesso à rede	Config habilitado

Controle da Sequência ISD:

- ISD #1: **Sempre enviado** - Contém dados essenciais do assinante
- ISD #2: Enviado apenas se isd_send_ss_data: true em runtime.exe
- ISD #3: Enviado apenas se isd_send_call_barring: true em runtime.exe

Resposta SendRoutingInfoForSM (SRI-for-SM)

Propósito: Fornece informações de roteamento MSC/MSMC para entrega de SMS. Quando um SMSc precisa entregar um SMS a um assinante, ele envia uma solicitação SRI-for-SM ao HLR para determinar onde roteá-lo.

Estrutura da Resposta:

Campo	Fonte	Descrição	Como Gerado	Exemplo
IMSI	Calculado	IMSI sintético derivado de MSISDN	PLMN_PREFIX + zero_padded_MSISDN	"001001555123456"
Número do Nó de Rede	runtime.exe	Endereço GT do SMSc para roteamento de SMS	smc_service_center_gt_address	"5551234567"

Parâmetros de Configuração (do runtime.exe):

```
# Endereço do Centro de Serviço GT (retornado nas respostas SRI-for-SM)
# Isso informa ao SMSc solicitante onde enviar mensagens MT-ForwardSM
smc_service_center_gt_address: "5551234567", # Obrigatório
```

```
# Configuração de Mapeamento MSISDN ↔ IMSI
# Prefixo PLMN: MCC (001 = Rede de Teste) + MNC (01 = Operador de Teste)
hlr_imsi_plmn_prefix: "001001", # Único parâmetro de configuração necessário!
```

Mapeamento MSISDN ↔ IMSI

Parâmetros de Configuração:

Esses parâmetros controlam como o OmniSS7 gera IMSIs sintéticos a partir de MSISDNs para respostas SRI-for-SM:

- hlr_imsi_plmn_prefix: O prefixo MCC+MNC a ser usado ao construir IMSIs sintéticos (por exemplo, "50557" para MCC=505, MNC=57)
- hlr_msisd_n_country_code: Código do país a ser precedido ao fazer mapeamento reverso IMSI→MSISDN (por exemplo, "61" para Austrália, "1" para EUA/Canadá)
- hlr_msisd_n_offset: Posição do caractere onde o Número de Assinante Nacional (NSN) começa dentro do MSISDN (normalmente 0 se o MSISDN não incluir código do país, ou comprimento do código do país se incluir)
- hlr_msisd_nsn_length: Número de dígitos a serem extraídos do MSISDN como o NSN

Para detalhes adicionais de configuração, consulte [Mapeamento MSISDN ↔ IMSI na Referência de Configuração](#).

Por que o Mapeamento MSISDN para IMSI é Necessário?

O protocolo MAP para **SendRoutingInfoForSM** (SRI-for-SM) requer que o HLR retorne um **IMSI** (Identidade Internacional do Assinante Móvel) em sua resposta. No entanto, o SMSc solicitante só conhece o **MSISDN** (número de telefone) do assinante.

Em uma rede tradicional:

- O SMSc envia SRI-for-SM com o MSISDN de destino (por exemplo, "5551234567")
- O HLR deve procurar o assinante em seu banco de dados para encontrar seu IMSI
- O HLR retorna o IMSI na resposta SRI-for-SM
- O SMSc então usa esse IMSI ao enviar MT-ForwardSM para o MSC/VLR

Abordagem do OmniSS7 - IMSIs Sintéticos:

Em vez de manter um banco de dados completo de assinantes com mapeamentos MSISDN-para-IMSI, o OmniSS7 usa um esquema de codificação simples para **calcular** IMSIs sintéticos diretamente a partir do MSISDN. Essa abordagem oferece dois benefícios principais:

1. **Privacidade:** IMSIs reais de assinantes armazenados no banco de dados do HLR nunca são expostos nas respostas SRI-for-SM enviadas pela rede SS7
2. **Simplicidade:** Não há necessidade de consultar o banco de dados do HLR para buscas de IMSI durante operações SRI-for-SM - o IMSI é calculado em tempo real a partir do MSISDN

Como Funciona:

MSISDNs são codificados diretamente na parte do assinante do IMSI (os dígitos após MCC+MNC):

```
IMSI = PLMN_PREFIX + zero_padded_MSISDN
```

Onde:

- **PLMN_PREFIX:** MCC + MNC (por exemplo, "001001" para Rede de Teste)
- **MSISDN:** Todos os dígitos numéricos do número de telefone
- **Zero Padding:** Preenchido à esquerda com zeros para preencher o IMSI exatamente em 15 dígitos

Exemplo Passo a Passo:

```
# Configuração
plmn_prefix = "001001" # MCC 001 + MNC 01

# Entrada: MSISDN da solicitação SRI-for-SM (decodificado TBCD)
msisdn = "555123456" # 9 dígitos

# Passo 1: Calcular espaço disponível para o número do assinante
subscriber_digits = 15 - String.length("001001") # = 9 dígitos

# Passo 2: Preencher MSISDN à esquerda com zeros para preencher a parte do assinante
padded_msisdn = String.pad_left("555123456", 9, "0") # = "555123456" (nenhum preenchimento necessário)

# Passo 3: Concatenar prefixo PLMN + MSISDN preenchido
imsi = "001001" <-> "555123456" # = "001001555123456" (exatamente 15 dígitos)
```

Exemplos Completos:

MSISDN de Entrada	Prefixo PLMN	Dígitos do Assinante	Disponíveis MSISDN Preenchido	IMSI Final	Notas
"555123456"	"001001"	(6)	9	"555123456"	"001001555123456" Ajuste exato, sem preenchimento
"99"	"001001"	(6)	9	"000000099"	"001001000000099" Preenchido à esquerda com zeros
"999999999"	"001001"	(6)	9	"999999999"	"001001999999999" Ajuste exato
"91123456789"	"001001"	(6)	9	"555123456"	"001001555123456" Muito longo, últimos 9 dígitos mantidos

Tratamento de Casos Limites:

- **MSISDNs Curtas:** Preenchidas à esquerda com zeros (por exemplo, "99" → "000000099")
- **MSISDNs Longas:** Os dígitos mais à direita são mantidos, os dígitos mais à esquerda são truncados (por exemplo, "91123456789" → "555123456")
- **Comprimento do IMSI:** Sempre exatamente 15 dígitos

Tratamento Reverso (IMSI → MSISDN):

O SMSc pode reverter esse mapeamento para converter IMSIs de volta em MSISDNs:

```
# Entrada: IMSI da resposta SRI-for-SM
imsi = "001001555123456"

# Passo 1: Remover prefixo PLMN
plmn_prefix = "001001"
subscriber_portion = String.slice(imsi, 6, 9) # = "555123456"

# Passo 2: Remover zeros à esquerda para obter MSISDN real
msisdn = String.replace_leading(subscriber_portion, "0", "") # = "555123456"
```

Exemplos de Mapeamento Reverso:

IMSI de Entrada	Prefixo PLMN	Porção do Assinante	Remover Zeros à Esquerda	MSISDN Final
"001001555123456"	"001001"	"555123456"	"555123456"	"555123456"
"001001000000099"	"001001"	"000000099"	"99"	"99"
"001001999999999"	"001001"	"999999999"	"999999999"	"999999999"

Propriedades deste Mapeamento:

- **↔ Determinístico:** O mesmo MSISDN sempre produz o mesmo IMSI
- **↔ Reversível:** Pode converter de IMSI para MSISDN
- **↔ Configuração Mínima:** Apenas requer hlr_imsi_plmn_prefix
- **↔ Preservação da Privacidade:** IMSIs reais nunca expostas
- **↔ Sem Consulta ao Banco de Dados:** Cálculo rápido, sem chamadas de API necessárias
- **↔ Sempre 15 Dígitos:** IMSI é sempre exatamente 15 dígitos

Tratamento de Entrada MSISDN:

Quando o HLR recebe uma solicitação SRI-for-SM, o MSISDN passa pela decodificação TBCD:

1. **Decodificação TBCD:** Converter TBCD binário em string (pode incluir prefixo TON/NPI como "91")
2. **Extrair Dígitos:** Manter apenas dígitos numéricos, remover quaisquer caracteres não numéricos
3. **Normalizar:** Se mais longo que o espaço disponível, pegar os dígitos mais à direita; se mais curto, preencher à esquerda com zeros
4. **Codificar:** Concatenar prefixo PLMN + MSISDN normalizado

Considerações de Segurança:

Os IMSIs sintéticos retornados nas respostas SRI-for-SM são puramente para fins de roteamento. Eles NÃO são os IMSIs reais armazenados no banco de dados de assinantes do HLR. Isso fornece uma camada adicional de proteção à privacidade, uma vez que os IMSIs reais dos assinantes são expostos apenas quando absolutamente necessário (por exemplo, durante operações UpdateLocation ou SendAuthenticationInfo que requerem vetores de autenticação reais).

Fluxo de Resposta:

1. SMSc → HLR: Solicitação SRI-for-SM
- MSISDN (TBCD): "91123456789" (inclui TON/NPI)
2. Processamento do HLR:
 - Decodificação TBCD: "91123456789"
 - Extrair dígitos: "91123456789" (11 dígitos)
 - Ajustar para 9 dígitos: "555123456" (últimos 9)
 - Adicionar PLMN: "001001" + "555123456" = "001001555123456"
 - Obter GT do SMSc da configuração: "555123456"
3. HLR → SMSc: Resposta SRI-for-SM
- IMSI: "001001555123456" (sintético, sempre 15 dígitos)
- Número do Nô de Rede: "5551234567" (onde enviar MT-ForwardSM)
4. SMSc envia MT-ForwardSM para "5551234567" com IMSI "001001555123456"

Configuração:

Os seguintes parâmetros são usados em runtime.exs:

```
# Prefixo PLMN: MCC (001 = Rede de Teste) + MNC (01 = Operador de Teste)
hlr_imsi_plmn_prefix: "001001",

# Extração de NSN (se MSISDNs incluírem código do país)
hlr_msisdn_country_code: "1", # Usado para mapeamento reverso (IMSI→MSISDN)
hlr_msisdn_nsn_offset: 1, # Pular 1 dígito do código do país
hlr_msisdn_nsn_length: 10 # Extrair 10 dígitos NSN
```

Tratamento de Extração de NSN:

Se seus MSISDNs incluírem o código do país (por exemplo, "68980000088" em vez de apenas "88000088"), você deve configurar a extração do NSN:

- **hlr_msisdn_nsn_offset:** Posição onde o NSN começa (normalmente o comprimento do seu código do país)
- **hlr_msisdn_nsn_length:** Número de dígitos no NSN

Exemplos:

Exemplo	Código do País	MSISDN	Exemplon	nsn_offset	nsn_length	NSN Extraído
CC de 1 dígito "9"		"95551234567"	1	10		"5551234567"
CC de 2 dígitos "99"		"99412345678"	2	9		"412345678"
CC de 3 dígitos "999"		"99988000088"	3	8		"88000088"

Como Funciona:

1. **MSISDN → IMSI:** Extrair NSN do MSISDN, preencher com zeros à esquerda, concatenar com prefixo PLMN

```
MSISDN: "99988000088"
NSN: String.slice("99988000088", 3, 8) = "88000088"
MSISDN Preenchido: "088000088" (9 dígitos)
IMSI: "547050" + "088000088" = "547050880000088"
```
2. **IMSI → MSISDN:** Remover prefixo PLMN, remover zeros à esquerda, proceder com código do país

```
IMSI: "547050880000088"
Porção do assinante: "088000088"
Remover zeros: "88000088"
MSISDN: "+999" + "88000088" = "+999880000088"
```

Requisitos da API: Nenhum - SRI-for-SM usa valores calculados e configuração apenas. Nenhuma chamada ao backend API é necessária.

Resumo da Fonte de Campos

Tipo de Fonte	Descrição	Exemplos
API OmniHSS	Dados dinâmicos do banco de dados de assinantes OmniHSS	IMSI, MSISDN, VLR/MSR atendido de circuit_session
runtime.exs	Parâmetros de configuração do OmniSS7	smc_service_center_gt_address, camel_service_key, isd_network_access_mode
Estatístico	Valores codificados nas respostas	Status do assinante, serviços de portadora, códigos SS
Solicitação	Campos extraídos da solicitação MAP recebida	IMSI da UpdateLocation, MSISDN da SRI
Calculado	Valores derivados usando lógica	IMSI sintético em SRI-for-SM (hlr_imsi_prefix + NSN)

Dependências de Configuração

Obrigatório em runtime.exs:

- hlr_service_center_gt_address - Usado nas respostas UpdateLocation
- smsc_service_center_gt_address - Usado nas respostas SRI-for-SM (onde as mensagens MT-ForwardSM devem ser roteadas)

Opcional em runtime.exe (com padrões):

- camel_service_key - Padrão: 11_110
- camel_trigger_detection_point - Padrão: :termAttemptAuthorized
- isd_network_access_mode - Padrão: :packetAndCircuit
- isd_send_ss_data - Padrão: true
- isd_send_call_barring - Padrão: true
- hlr_imsi_plmn_prefix - Padrão: "001001" (prefixo PLMN para mapeamento MSISDN~IMSI)

Requisitos do OmniHSS:

O OmniHSS deve fornecer endpoints da API REST para:

- Consulta de assinante por IMSI e MSISDN
- Atualizações de localização da sessão de circuito (atribuição VLR/MSC)
- Geração de vetores de autenticação
- Consultas sobre status do assinante e perfil de serviço

Documentação Relacionada

Documentação do OmniSS7:

- [= Voltar à Documentação Principal](#)
- [Guia de Recursos Comuns](#)
- [Guia do Cliente MAP](#)
- [Referência Técnica](#)
- [Referência de Configuração](#)

Documentação do OmniHSS: Para gerenciamento de assinantes, provisionamento, configuração de autenticação e operações administrativas, consulte a **documentação do produto OmniHSS**. O OmniHSS contém toda a lógica do banco de dados de assinantes, algoritmos de autenticação, regras de provisionamento de serviços e capacidades de gerenciamento Multi-IMSI.

OmniSS7 por Omnitouch Network Services

Guia de Configuração do Cliente MAP

[← Voltar para a Documentação Principal](#)

Este guia fornece uma configuração detalhada para usar o OmniSS7 como um **Cliente MAP** para enviar solicitações do protocolo MAP para elementos de rede.

OmniSS7 Stack v1.0.0

OmniSS7 Stack v1.0.0

Copyright © 2024 OmniSS7

SS7 Events

SS7 Client

M3UA

Routing

Routing Test

SMSc Links

Resources

Configuration

MAP Test Client

SendRoutingInfo

PRN

SendRoutingInfoForSM

SendAuthenticationInfo

MT-forwardSM

ForwardSM

UpdateLocation

CancelLocation

AnyTimeUpdate

SendSM

SMS-Deliver-PDU

mslscn

123456789

gmsc

Submit

Got response in 0 seconds

RESPONSE

```
{
  "components": [
    {
      "type": "BasicRMS",
      "value": {
        "action": "returnError",
        "errorCode": 27,
        "severity": 0
      }
    }
  ],
  "id": "B75ABF93",
  "type": "end"
}
```

Índice

- 1. [O que é o Modo Cliente MAP?](#)
- 2. [Habilitando o Modo Cliente MAP](#)
- 3. [Operações MAP Disponíveis](#)
- 4. [Enviando Solicitações via API](#)
- 5. [Métricas e Monitoramento](#)
- 6. [Solução de Problemas](#)

O que é o Modo Cliente MAP?

Modo Cliente MAP permite que o OmniSS7 se conecte como um **Processo de Servidor de Aplicação (ASP)** a um par M3UA (STP ou SGP) e envie/receba mensagens **MAP (Mobile Application Part)** para serviços como:

- **Consultas HLR:** SRI (Enviar Informações de Roteamento), SRI-para-SM, Informações de Autenticação
- **Atualizações de Localização:** Atualizar Localização, Cancelar Localização
- **Gestão de Assinantes:** Fornecer Número de Roaming (PRN), Inserir Dados do Assinante

Arquitetura da Rede

Habilitando o Modo Cliente MAP

Edite config/runtime.exe e configure as definições do cliente MAP. Para referência completa de configuração, veja [Parâmetros de Conexão M3UA na Referência de Configuração](#).

Configuração Básica

```
config :omni7,
# Habilitar modo Cliente MAP
map_client_enabled: true,

# Conexão M3UA para Cliente MAP (conecta como ASP ao STP/SGP remoto)
map_client_m3ua: %{
  mode: "ASP",
  callback: (MapClient, :handle_payload, []), # Modo M3UA: "ASP" (cliente) ou "SGP" (servidor)
  process_name: :map_client_asp, # Nome do processo registrado
  local_ip: {10, 0, 0, 100}, # Endereço IP local
  local_port: 2905, # Porta SCTP local
  remote_ip: {10, 0, 0, 1}, # IP STP/SGP remoto
  remote_port: 2905, # Porta STP/SGP remoto
  routing_context: 1 # Contexto de roteamento M3UA
}
```

Exemplo de Configuração para Produção

```
config :omni7,
# Habilitar Cliente MAP para produção
map_client_enabled: true,

# Conexão M3UA para produção
map_client_m3ua: %{
  mode: "ASP",
  callback: (MapClient, :handle_payload, []),
  process_name: :map_client_asp,
  local_ip: {10, 0, 0, 100},
  local_port: 2905,
  remote_ip: {10, 0, 0, 1}, # IP STP de produção
  remote_port: 2905,
  routing_context: 1
}

config :control_panel,
web: %{
  listen_ip: "0.0.0.0",
  port: 443,
  hostname: "ss7-gateway.example.com",
  enable_tls: true,
  tls_cert: "/etc/ssl/certs/gateway.crt",
  tls_key: "/etc/ssl/private/gateway.key"
}
```

Operações MAP Disponíveis

MAP Tester API 1.0.0 OAS 3.0

hwaggeer.json

default ^

POST

/api/mt-forwardSM

MT-forwardSM MAP Request (For sending SMS for delivery by remote MSC/SMSC)

^

POST

/api/deliverPDU

Utility: Build SMS-DELIVER TPDU from originating address + GSM7

^

POST

/api/forwardSM

forwardSM MAP Request

^

POST

/api/prn

ProvideRoamingNumber (PRN) MAP Request

^

POST

/api/send-auth-info

SendAuthenticationInfo MAP Request

^

POST

/api/sendSM

Utility: Perform SRI-for-SM + MT-forwardSM from MSISDN and GSM7

^

POST

/api/sri

SendRoutingInfo MAP Request

^

POST

/api/sri-for-sm

SendRoutingInfoForSM MAP Request

^

POST

/api/updateLocation

UpdateLocation MAP Request

^

GET

/metrics

Prometheus metrics

^

GET

/swagger.json

OpenAPI spec

^

Schemas ^

AuthInfoRequest

>

ErrorResponse

>

ForwardSMRequest

>

PRNRequest

>

SMSDeliverPDURequest

>

SMSDeliverPDUResponse

>

SRIForSMRequest

>

SRIRequest

>

SRIResponse

>

SendSMRequest

>

UpdateLocationRequest

>

1. Enviar Informações de Roteamento para SM (SRI-para-SM)

Consulta o HLR para determinar o MSC que atende a entrega de SMS. Para informações detalhadas sobre como o HLR processa solicitações SRI-para-SM, veja [SRI-para-SM no Guia HLR](#).

Endpoint da API: POST /api/sri-for-sm

Solicitação:

```
{
  "msisdn": "447712345678",
  "serviceCenter": "447999123456"
}
```

Resposta:

```
{
  "result": {
    "imsi": "234509876543210",
    "locationInfoWithLMSI": {
      "networkNode-Number": "447999555111"
    }
  }
}
```

Exemplo de cURL:

```
curl -X POST http://localhost/api/sri-for-sm \
-H 'Content-Type: application/json' \
-d '{
  "msisdn": "447712345678",
  "serviceCenter": "447999123456"
}'
```

2. Enviar Informações de Roteamento (SRI)

Consulta o HLR para informações de roteamento de chamadas de voz.

Endpoint da API: POST /api/sri

Solicitação:

```
{
  "msisdn": "447712345678",
  "gmsc": "447999123456"
}
```

Resposta:

```
{
  "result": {
    "imsi": "234509876543210",
    "extendedRoutingInfo": {
      "routingInfo": {
        "roamingNumber": "447999555222"
      }
    }
  }
}
```

3. Fornecer Número de Roaming (PRN)

Solicita um número de roaming temporário (MSRN) do MSC que atende.

Endpoint da API: POST /api/prn

Solicitação:

```
{
  "msisdn": "447712345678",
  "gmsc": "447999123456",
  "msc_number": "447999555111",
  "imsi": "234509876543210"
}
```

4. Enviar Informações de Autenticação

Solicita vetores de autenticação do HLR para autenticação do assinante.

Endpoint da API: POST /api/send-auth-info

Solicitação:

```
{
  "imsi": "234509876543210",
  "vectors": 5
}
```

Resposta:

```
{
  "result": {
    "authenticationSetList": [
      {
        "rand": "0123456789ABCDEF0123456789ABCDEF",
        "xres": "ABCDEF0123456789",
        "ck": "0123456789ABCDEF0123456789ABCDEF",
        "ik": "FEDCBA9876543210FEDCBA9876543210",
        "autn": "0123456789ABCDEF0123456789ABCDEF"
      }
    ]
  }
}
```

5. Atualizar Localização

Registra a localização atual de um assinante no HLR. Para informações detalhadas sobre o processamento de AtualizarLocalização e sequências de InserirDadosDoAssinante, veja [Atualizações de Localização no Guia HLR](#).

Endpoint da API: POST /api/updateLocation

Solicitação:

```
{
  "imsi": "234509876543210",
  "vlr": "447999555111"
}
```

Resumo das Operações MAP

Enviando Solicitações via API

Usando o Swagger UI

O Swagger UI fornece uma interface interativa para enviar solicitações SS7.

Acessar o Swagger UI:

1. Navegue até `http://your-server/swagger`
2. Navegue pelos endpoints da API disponíveis
3. Clique em qualquer endpoint para expandir seus detalhes

Enviando uma Solicitação:

1. Clique no endpoint que deseja usar (por exemplo, `/api/sri-for-sm`)
2. Clique no botão "Try it out"
3. Preencha os parâmetros necessários no corpo da solicitação
4. Clique em "Execute"
5. Veja a resposta abaixo

Códigos de Resposta da API

- **200** - Sucesso, resultado retornado no corpo da resposta
- **400** - Solicitação Inválida, parâmetros inválidos
- **504** - Tempo Limite do Gateway, sem resposta da rede SS7 dentro de 10 segundos

Métricas do Cliente MAP

Métricas Disponíveis

Métricas de Solicitação:

- `map_requests_total` - Número total de solicitações MAP enviadas
 - Rótulos: `operation` (valores: `sri`, `sri_for_sm`, `prn`, `authentication_info`, etc.)
- `map_request_errors_total` - Número total de erros de solicitação MAP
 - Rótulos: `operation`
- `map_request_duration_milliseconds` - Histograma das durações das solicitações MAP
 - Rótulos: `operation`
- `map_pending_requests` - Número atual de solicitações MAP pendentes (gauge)

Exemplos de Consultas Prometheus

```
# Total de solicitações SRI-para-SM na última hora
increase(map_requests_total{operation="sri_for_sm"}[1h])

# Tempo médio de resposta para solicitações SRI
rate(map_request_duration_milliseconds_sum{operation="sri"}[5m]) /
rate(map_request_duration_milliseconds_count{operation="sri"}[5m])

# Taxa de erro para todas as operações MAP
sum(rate(map_request_errors_total[5m])) by (operation)

# Solicitações pendentes atuais
map_pending_requests
```

Solução de Problemas do Cliente MAP

Problema: Tempo Limite das Solicitações

Sintomas:

- A API retorna 504 Gateway Timeout
- Sem resposta do HLR/MSC

Verificações:

1. Verifique se a conexão M3UA está ATIVA:

```
# No console IEx
:sys.get_state(:map_client_asp)
```
2. Verifique a conectividade de rede com o STP
3. Verifique o contexto de roteamento e o endereçamento SCCP
4. Verifique os logs em busca de erros SCCP

Problema: Erros SCCP

Sintomas:

- A API retorna respostas de erro SCCP
- Logs mostram mensagens de "serviço unitdata SCCP"

Códigos de Erro SCCP Comuns:

- **Sem Tradução:** Título Global não encontrado na tabela de roteamento do STP
- **Falha de Subsistema:** Subsistema de destino (HLR SSN 6) está indisponível
- **Falha de Rede:** Congestionamento ou falha na rede

Soluções:

- Contate o administrador do STP para verificar a configuração de roteamento
- Verifique se o Título Global de destino é acessível
- Verifique se o subsistema de destino está operacional

Documentação Relacionada

- [~ Voltar para a Documentação Principal](#)
- [Guia de Recursos Comuns](#) - Interface Web, API, Monitoramento
- [Guia SIP](#) - Configuração de roteamento
- [Guia do Centro de SMS](#) - Entrega de SMS
- [Referência Técnica](#) - Especificações do protocolo

OmniSS7 por Omnitouch Network Services



Guia de Configuração do Centro de SMS (SMSc)

[← Voltar para a Documentação Principal](#)

Este guia fornece uma configuração detalhada para usar o OmniSS7 como um **Centro de SMS (SMSc)** na frente com o **OmniMessage** como o backend de armazenamento e entrega de mensagens.

Integração com o OmniMessage

O modo SMSc do OmniSS7 funciona como uma **frente de sinalização SS7** que se conecta ao **OmniMessage**, uma plataforma de SMS de nível de operadora. Esta arquitetura separa as preocupações:

- **OmniSS7 (Frontend SMSc):** Lida com toda a sinalização do protocolo SS7/MAP, roteamento SCCP e comunicação de rede
- **OmniMessage (Backend SMS):** Gerencia armazenamento de mensagens, enfileiramento, lógica de repetição, rastreamento de entrega e decisões de roteamento

Por que o OmniMessage?

O OmniMessage fornece capacidades de mensagens SMS de nível de operadora com recursos, incluindo:

- **Gerenciamento de Fila de Mensagens:** Armazenamento persistente com lógica de repetição configurável e enfileiramento por prioridade
- **Rastreamento de Entrega:** Status de entrega em tempo real, relatórios de entrega (DLR) e rastreamento de motivos de falha
- **Suporte a Múltiplos SMSc:** Várias instâncias de frontend podem se conectar a um único backend OmniMessage para balanceamento de carga e redundância
- **Inteligência de Roteamento:** Regras de roteamento avançadas com base no destino, remetente, conteúdo da mensagem e hora do dia
- **Limitação de Taxa:** Controles de TPS (transações por segundo) por rota para evitar congestionamento da rede
- **Design API-First:** API HTTP RESTful para integração com sistemas de cobrança, portais de clientes e aplicativos de terceiros
- **Análise e Relatórios:** Estatísticas de volume de mensagens, taxas de sucesso de entrega e métricas de desempenho

Todos os dados de mensagens, estado de entrega e configurações de roteamento são armazenados e gerenciados no OmniMessage. O OmniSS7 consulta o OmniMessage por meio de chamadas de API HTTPS para recuperar mensagens pendentes, atualizar o status de entrega e registrar-se como um frontend ativo.

Importante: O modo SMSc do OmniSS7 é um **frontend de sinalização apenas**. Toda a lógica de roteamento de mensagens, gerenciamento de filas, algoritmos de repetição, rastreamento de entrega e regras de negócios são tratados pelo OmniMessage. Este guia cobre a configuração do protocolo SS7/MAP no OmniSS7. Para informações sobre roteamento de mensagens, configuração de filas, relatórios de entrega, limitação de taxa e análises, **consulte a documentação do OmniMessage**.

Índice

1. [Integração com o OmniMessage](#)
 2. [O que é o Modo Centro de SMS?](#)
 3. [Habilitando o Modo SMSc](#)
 4. [Configuração da API HTTP](#)
 5. [Fluxos de Mensagens SMS](#)
 6. [Prevenção de Loop](#)
 7. [Rastreamento de Assinantes SMSc](#)
 8. [Configuração de Auto-Flush](#)
 9. [Métricas e Monitoramento](#)
 10. [Solução de Problemas](#)
-

O que é o Modo Centro de SMS?

Nota: Esta seção cobre apenas a configuração de sinalização SS7 do OmniSS7. Para regras de roteamento de mensagens, gerenciamento de filas, rastreamento de entrega e configuração de lógica de negócios, consulte a **documentação do produto OmniMessage**.

Modo Centro de SMS permite que o OmniSS7 funcione como um SMSc para:

- **Entrega MT-SMS:** Entrega de SMS Terminado em Móvel para assinantes
- **Manipulação de MO-SMS:** Recepção e roteamento de SMS Originado em Móvel
- **Enfileiramento de Mensagens:** Fila de mensagens com suporte a banco de dados e lógica de repetição
- **Auto-Flush:** Entrega automática de SMS da fila
- **Relatórios de Entrega:** Rastrear status de entrega de mensagens

Arquitetura do Centro de SMS

Habilitando o Modo SMSc

O OmniSS7 pode operar em diferentes modos. Para usá-lo como um SMSc, você precisa habilitar o modo SMSc na configuração.

Mudando para o Modo SMSc

O config/runtime.exs do OmniSS7 contém três modos operacionais pré-configurados. Para habilitar o modo SMSc:

1. **Abra** config/runtime.exs
2. **Encontre** as três seções de configuração (linhas 53-204):
 - Configuração 1: Modo STP (linhas 53-95)
 - Configuração 2: Modo HLR (linhas 97-142)
 - Configuração 3: Modo SMSc (linhas 144-204)
3. **Comente** qualquer outra configuração ativa (adicione # a cada linha)
4. **Descomente** a configuração SMSc (remova # das linhas 144-204)
5. **Personalize** os parâmetros de configuração conforme necessário
6. **Reinicie** a aplicação: iex -S mix

Configuração do Modo SMSc

A configuração completa do SMSc se parece com isto:

```
config :omniss7,
  # Flags de modo - Habilitar recursos STP + SMSc
  # Nota: map_client_enabled é verdadeiro porque o SMSc precisa de capacidades de roteamento
  map_client_enabled: true,
  hlr_mode_enabled: false,
  smsc_mode_enabled: true,

  # Configuração da API do Backend OmniMessage
  smsc_api_base_url: "https://10.179.3.219:8443",
  # Identificação do SMSc para registro com o backend
  smsc_name: "ipsmgw",
  # Endereço GT do Centro de Serviço para operações de SMS
  smsc_service_center_gt_address: "5551234567",

  # Configuração de Auto Flush (processamento de fila de SMS em segundo plano)
  auto_flush_enabled: true,
  auto_flush_interval: 10_000,
  auto_flush_dest_smsc: "ipsmgw",
  auto_flush_tps: 10,

  # Configuração de Conexão M3UA
  # Conectar como ASP para enviar/receber operações MAP SMS
  map_client_m3ua: %{}
}
```

```

mode: "ASP",
callback: {MapClient, :handle_payload, []},
process_name: :stp_client_asp,
# Endpoint local (sistema SMSc)
local_ip: {10, 179, 4, 12},
local_port: 2905,
# Endpoint remoto STP
remote_ip: {10, 179, 4, 10},
remote_port: 2905,
routing_context: 1
}

config :control_panel,
  use_additional_pages: [
    {SS7.Web.EventsLive, "/events", "Eventos SS7"},
    {SS7.Web.TestClientLive, "/client", "Cliente SS7"},
    {SS7.Web.M3UAStatusLive, "/m3ua", "M3UA"},
    {SS7.Web.RoutingLive, "/routing", "Roteamento"},
    {SS7.Web.RoutingTestLive, "/routing_test", "Teste de Roteamento"},
    {SS7.Web.SmscLinksLive, "/smsc_links", "Links SMSc"}
  ],
  page_order: ["/events", "/client", "/m3ua", "/routing", "/routing_test", "/smsc_links",
"/application", "/configuration"]

```

Parâmetros de Configuração para Personalizar

Para uma referência completa de todos os parâmetros de configuração, consulte a [Referência de Configuração](#).

Parâmetro	Tipo	Padrão	Descrição	Exemplo
smsc_api_base_url	String	Obrigatório	Endpoint da API do backend OmniMessage	"https://10.179.3.219:8443"
smsc_name	String	"{hostname}_SMSc"	Seu identificador SMSc para registro	"ipsmgw"
smsc_service_center_gt_address	String	Obrigatório	Título Global do Centro de Serviço	"5551234567"
auto_flush_enabled	Boolean	true	Habilitar processamento automático de filas	false
auto_flush_interval	Integer	10_000	Intervalo de processamento de filas em milissegundos	5_000
auto_flush_dest_smsc	String	Obrigatório	Nome do SMSc de destino para auto-flush	"ipsmgw"
auto_flush_tps	Integer	10	Taxa de processamento de mensagens (transações/segundo)	20
local_ip	Tuple	Obrigatório	Endereço IP do seu sistema SMSc	{10, 179, 4, 12}
local_port	Integer	2905	Porta SCTP local	2905
remote_ip	Tuple	Obrigatório	Endereço IP do STP para conectividade	{10, 179, 4, 10}

Parâmetro	Tipo	Padrão	Descrição	Exemplo
remote_port	Integer	2905	SS7 Porta SCTP remota	2905
routing_context	Integer	1	ID do contexto de roteamento M3UA	1

O que Acontece Quando o Modo SMS Sc é Habilitado

Quando `smssc_mode_enabled: true` e `map_client_enabled: true`, a interface web mostrará:

- **Eventos SS7** - Registro de eventos
- **Ciente SS7** - Teste de operações MAP
- **M3UA** - Status da conexão
- **Roteamento** - Gerenciamento da tabela de rotas (STP habilitado)
- **Teste de Roteamento** - Teste de rotas (STP habilitado)
- **Links SMS Sc** - Status da API SMS Sc + gerenciamento de fila de SMS ← *Específico do SMS Sc*
- **Recursos** - Monitoramento do sistema
- **Configuração** - Visualizador de configuração

A aba **Links HLR** será oculta.

Notas Importantes

- O modo SMS Sc requer `map_client_enabled: true` para capacidades de roteamento
- **Backend OmniMessage**: A API do OmniMessage deve ser acessível no `smssc_api_base_url` configurado
- **Registro do Frontend**: O sistema se registra automaticamente com o OmniMessage a cada **5 minutos** por meio do módulo `SMS.FrontendRegistry`
- **Tempo Limite de Solicitação da API**: Todas as solicitações da API do OmniMessage têm um **tempo limite fixo de 5 segundos**
- **Tempo Limite de Solicitação MAP**: Todas as solicitações MAP (SRI-for-SM, MT-ForwardSM, etc.) têm um **tempo limite fixo de 10 segundos**
- O auto-flush processa automaticamente a fila de SMS em segundo plano
- A conexão M3UA com o STP é necessária para enviar/receber operações MAP SMS
- Após mudar de modo, você deve reiniciar a aplicação para que as alterações tenham efeito
- **Interface Web**: Consulte o [Guia da Interface Web](#) para informações sobre como usar a interface web
- **Acesso à API**: Consulte o [Guia da API](#) para documentação da API REST e acesso ao Swagger UI

Configuração da API HTTP

Configuração do Backend OmniMessage

O OmniSS7 se comunica com o OmniMessage via API REST HTTPS para gerenciar a entrega de mensagens, rastrear o estado do assinante e registrar-se como um frontend ativo:

```
config :omniss7,
  # URL base da API OmniMessage
  smssc_api_base_url: "https://10.5.198.200:8443",
  # Identificador do nome do SMSC para registro (padrão para hostname_SMS Sc se vazio)
  smssc_name: "omni-smssc01",
  # Endereço GT do Centro de Serviço para operações de SMS
  smssc_service_center_gt_address: "5551234567"
```

Parâmetros de Configuração:

Parâmetro	Tipo Obrigatório	Padrão	Descrição
smssc_api_base_url	String Sim	"https://localhost:8443"	URL base para a API do OmniMessage
smssc_name	String Não	" " (usa "{hostname}_SMS Sc")	Identificador do SMSC para registro e gerenciamento de filas
smssc_service_center_gt_address	String Não	"5551234567"	Endereço GT do Centro de

Parâmetro	Tipo Obrigatório	Padrão	Descrição
			Serviço retornado nas respostas SRI-for-SM. Isso informa outros elementos da rede onde encaminhar mensagens MT-ForwardSM. Veja o Guia SRI-for-SM para detalhes.

Registro do Frontend

O sistema se registra automaticamente com o OmniMessage na inicialização e **re-registra a cada 5 minutos** por meio do módulo SMS.FrontendRegistry. Isso permite que o OmniMessage:

- Rastreie frontends ativos para balanceamento de carga
- Monitore tempo de atividade e status de saúde
- Coleta informações de configuração
- Gerencie roteamento de SMS distribuído entre vários frontends

Detalhes da Implementação:

- **Intervalo de Registro:** 5 minutos (fixo)
- **Processo:** Iniciado automaticamente quando `smsc_mode_enabled: true`

Payload de Registro:

```
{
  "frontend_name": "omni-smsc01",
  "configuration": "{...}",
  "frontend_type": "SS7",
  "hostname": "sm-sc-server01",
  "uptime_seconds": 12345
}
```

Nota: O nome do frontend é retirado do parâmetro de configuração `smsc_name`. Se não estiver definido, ele padrão para `{hostname}_SMSc`.

Comunicação da API OmniMessage

Quando o OmniSS7 recebe operações MAP da rede SS7 ou processa a fila de mensagens, ele se comunica com o OmniMessage para:

- **Registrar-se como um frontend ativo** e relatar status de saúde
- **Submeter mensagens originadas em móvel (MO)** recebidas de assinantes
- **Recuperar mensagens terminadas em móvel (MT)** da fila para entrega
- **Atualizar status de entrega** com relatórios de sucesso/falha
- **Consultar informações de roteamento** para encaminhamento de mensagens

Endpoint	Método	Propósito	Corpo da Solicitação
/api/frontends	POST	Registrar instância de frontend	{"frontend_name": "...", "frontend_type": "SMSc", "hostname": "...", "uptime_seconds": ...}
/api/messages_raw	POST	Inserir nova mensagem SMS	{"source_msisdn": "...", "source_smsc": "...", "message_body": "..."} Cabeçalho: smsc: <sm-sc_name>
/api/messages	GET	Obter fila de mensagens	
/api/messages/{id}	PATCH	Marcar mensagem como entregue	{"deliver_time": "...", "dest_smsc": "..."} Atualizar status da mensagem
/api/messages/{id}	PUT		{"dest_smsc": null}
/api/	POST	Inserir/	{"msisdn": "...", "imsi": "...", "location": "...",

Endpoint	Método	Propósito	Corpo da Solicitação
locations		atualizar localização do assinante	"ims_capable": true, "csfb": false, "expires": "...", "user_agent": "...", "ran_location": "...", "imei": "...", "registered": "..."}
/api/events	POST	Adicionar rastreamento de eventos	{"message_id": ..., "name": "...", "description": "..."}
/api/status	GET	Verificação de saúde	-

Formato de Resposta da API

Todas as respostas da API usam o formato JSON com as seguintes convenções:

- **Respostas de sucesso:** HTTP 200-201 com corpo JSON contendo dados de resultado
- **Respostas de erro:** HTTP 4xx/5xx com detalhes de erro no corpo da resposta
- **Carimbos de data/hora:** Formato ISO 8601 (por exemplo, "2025-10-21T12:34:56Z")
- **IDs de mensagem:** Identificadores inteiros ou de string

Módulos do Cliente da API

O sistema SMS consiste em três módulos principais:

1. SMS.APIClient

Módulo principal do cliente da API que fornece toda a comunicação da API HTTP com o OmniMessage:

- frontend_register/4 - Registrar frontend com o OmniMessage
- insert_message/3 - Inserir mensagem SMS bruta (versão de 3 parâmetros compatível com Python)
- insert_location/9 - Inserir/atualizar dados de localização do assinante
- get_message_queue/2 - Recuperar mensagens pendentes da fila
- mark_dest_smsc/3 - Marcar mensagem como entregue ou falhada
- add_event/3 - Adicionar rastreamento de eventos para mensagens
- flush_queue/2 - Processar mensagens pendentes (SRI-for-SM + MT-forwardSM)
- auto_flush/2 - Loop de processamento contínuo da fila

2. SMS.FrontendRegistry

Lida com o registro periódico do frontend com o backend:

- Registra automaticamente na inicialização
- Re-registra a cada 5 minutos
- Usa smsc_name da configuração (reverte para hostname)
- Coleta informações de configuração e tempo de atividade do sistema

3. SMS.Util

Funções utilitárias para operações SMS:

- generate_tp_scts/0 - Gerar timestamp de SMS no formato TPDU

Fluxos de Mensagens SMS

Fluxo de SMS de Entrada (Originado em Móvel)

Fluxo de SMS de Saída (Terminado em Móvel)

Principais Etapas Explicadas:

- **Solicitação SRI-for-SM:** O SMSc consulta o HLR com o MSISDN de destino para determinar onde encaminhar a mensagem SMS. O HLR responde com:
 - Um IMSI sintético (calculado a partir do MSISDN para privacidade) - veja [Mapeamento MSISDN](#)

↔ [IMSI](#)

- O endereço GT do SMSC (número do nó da rede) onde o MT-ForwardSM deve ser enviado
- Para detalhes completos sobre como isso funciona, veja [SRI-for-SM no Guia HLR](#)

- **Solicitação MT-forwardSM:** Uma vez obtidas as informações de roteamento, o SMSc envia a mensagem SMS real para o MSC/VLR que atende o assinante

Estrutura do SMS TPDU

Manipulação de Centro de Serviço de Alerta

O SMSc pode receber mensagens **alertServiceCenter** do HLR para rastrear o status de alcançabilidade do assinante.

Para informações sobre como o HLR envia mensagens alertServiceCenter, veja [Integração do Centro de Serviço de Alerta no Guia HLR](#).

O que é alertServiceCenter?

Quando um assinante realiza uma UpdateLocation no HLR (ou seja, registra-se em um novo VLR/MSC), o HLR pode notificar os sistemas SMSc de que o assinante agora está acessível, enviando uma mensagem **alertServiceCenter** (código de operação MAP 64).

Configuração

O tempo de expiração da localização é configurado no HLR:

```
config :omniss7,  
  # Tempo de expiração da localização quando o SMSc recebe alertServiceCenter (padrão: 48  
  horas)  
  hlr_alert_location_expiry_seconds: 172800
```

Comportamento

Quando o SMSc recebe uma mensagem alertServiceCenter:

1. **Decodificar MSISDN:** Extrair o MSISDN do assinante da mensagem (formato TBCD)
2. **Remover prefixo TON/NPI:** Remover prefixos comuns como "19", "11", "91" (por exemplo, "19123123213" → "123123213")
3. **Calcular IMSI:** Gerar IMSI sintético usando o mesmo mapeamento que SRI-for-SM
4. **POST para /api/location:** Atualizar o banco de dados de localização com:
 - msisdn: Número de telefone do assinante (limpo)
 - imsi: IMSI sintético
 - location: Nome do SMSc (por exemplo, "ipsmgw")
 - expires: Hora atual + hlr_alert_location_expiry_seconds
 - csfb: true (assinante acessível via Circuit-Switched Fallback)
 - ims_capable: false (este é o registro CS 2G/3G, não IMS/VoLTE)
 - user_agent: GT do HLR que enviou o alerta (para rastreamento)
 - ran_location: "SS7"
5. **Rastrear no Rastreador de Assinantes SMSc:** Registrar o assinante com GT do HLR, status=ativo, contadores de mensagens em 0
6. **Enviar ACK:** Responder ao HLR com reconhecimento de alertServiceCenter

Manipulação de Assinante Ausente

Quando o SMSc tenta entregar uma mensagem e recebe um erro de "assinante ausente" durante SRI-for-SM (para mais informações sobre SRI-for-SM, veja [SRI-for-SM no Guia HLR](#)):

1. **Detectar ausência:** SRI-for-SM retorna erro absentSubscriberDiagnosticSM
2. **Expirar localização:** POST para /api/location com expires=0 para marcar o assinante como inacessível
3. **User agent:** Definir como "SS7_AbsentSubscriber" para identificar a origem
4. **Atualizar rastreador:** Marcar assinante como failed no Rastreador de Assinantes SMSc

Isso garante que o banco de dados de localização e o rastreador reflitam com precisão o status de alcançabilidade do assinante.

Diagrama de Fluxo

Endpoint da API

POST /api/location

```
{
  "msisdn": "15551234567",
  "imsi": "001010123456789",
  "location": "ipsmgw",
  "ims_capable": false,
  "csfb": true,
  "expires": "2025-11-01T12:00:00Z",
  "user_agent": "15551111111",
  "ran_location": "SS7",
  "imei": "",
  "registered": "2025-10-30T12:00:00Z"
}
```

Nota: O campo `user_agent` contém o GT do HLR que enviou o `alertServiceCenter`, permitindo que o SMS Sc rastreie qual HLR está fornecendo atualizações de localização.

Para assinantes ausentes, `expires` é definido como hora atual (expiração imediata).

Prevenção de Loop

O SMS Sc implementa **prevenção automática de loop** para evitar loops infinitos de roteamento de mensagens quando as mensagens se originam de redes SS7.

Por que a Prevenção de Loop é Importante

Quando o SMS Sc recebe mensagens SMS originadas em móvel (MO) da rede SS7, ele as insere na fila de mensagens com um campo `source_smsc` identificando sua origem (por exemplo, "SS7_GT_15551234567"). Sem a prevenção de loop, essas mensagens poderiam ser:

1. Recebidas da rede SS7 → Enfileiradas com `source_smsc` contendo "SS7"
2. Recuperadas da fila → Processadas para entrega
3. Enviadas de volta para a rede SS7 → Criando um loop

Como Funciona

O SMS Sc detecta e previne automaticamente loops durante o processamento de mensagens:

Implementação

Ao processar mensagens da fila, o SMS Sc verifica o campo `source_smsc`:

- **Se `source_smsc` contém "SS7":**
 - A mensagem é pulada
 - Evento adicionado: "Prevenção de Loop" com descrição explicando o motivo do pulo
 - Mensagem marcada como falhada via solicitação PUT
 - Registrada com nível de aviso
- **Caso contrário:**
 - Mensagem processada normalmente
 - Operações SRI-for-SM e MT-ForwardSM prosseguem

Valores de SMSC de Origem

As mensagens podem ter vários valores de `source_smsc`:

Origem	Valor de Exemplo	Ação
Rede SS7 (MO-FSM)	"SS7_GT_15551234567"	Pulada - Prevenção de loop
API Externa/SMPP	"ipsmgw" ou "api_gateway"	Processada normalmente
Outro SMSc	"smsc-node-01"	Processada normalmente

Rastreamento de Eventos

Quando uma mensagem é pulada devido à prevenção de loop, um evento é registrado:

```
{
  "message_id": 12345,
  "name": "Prevenção de Loop",
  "description": "Mensagem pulada - source_smsc 'SS7_GT_15551234567' contém 'SS7', prevenindo loop de mensagem"
}
```

Esse evento é visível em:

- **Interface Web:** Página de Eventos SS7 (/events)
- **Banco de Dados:** Tabela events via API
- **Logs:** Entradas de log de nível de aviso

Configuração

A prevenção de loop está **sempre habilitada** e não pode ser desabilitada. Este é um recurso crítico de segurança para evitar interrupções na rede devido a loops de mensagens.

Cenário de Exemplo

Cenário: Assinante móvel envia SMS via rede SS7

1. Telefone móvel → MSC/VLR → SMSc (via MO-ForwardSM)
2. SMSc recebe MO-FSM do GT 15551234567
3. SMSc insere na fila: `source_smsc = "SS7_GT_15551234567"`
4. Auto-flush recupera mensagem da fila
5. SMSc detecta "SS7" em `source_smsc` → PULAR
6. Evento registrado: "Prevenção de Loop"
7. Mensagem marcada como falhada
8. Nenhum SRI-for-SM ou MT-ForwardSM enviado (loop prevenido)

Sem a prevenção de loop, a etapa 8 enviaria a mensagem de volta para a rede SS7, potencialmente criando um loop infinito.

Rastreamento de Assinantes SMSc

O SMSc inclui um GenServer de **Rastreador de Assinantes** que mantém o estado em tempo real para assinantes com base em mensagens `alertServiceCenter` e tentativas de entrega de mensagens.

Propósito

O rastreador fornece:

- **Monitoramento de alcançabilidade:** Quais assinantes estão atualmente acessíveis
- **Rastreamento de HLR:** Qual HLR enviou o `alertServiceCenter` para cada assinante
- **Contadores de mensagens:** Número de mensagens enviadas/recebidas por assinante
- **Rastreamento de falhas:** Marcar assinantes como falhados quando tentativas de entrega falham
- **Visibilidade na Interface Web:** Painel em tempo real mostrando todos os assinantes rastreados

Informações Rastreadas

Para cada assinante, o rastreador armazena:

Campo	Descrição	Exemplo
msisdn	Número de telefone do assinante (chave)	"15551234567"
imsi	IMSI do assinante	"001010123456789"
hlr_gt	HLR GT que enviou alertServiceCenter	"15551111111"
messages_sent	Contagem de mensagens MT-FSM enviadas	5
messages_received	Contagem de mensagens MO-FSM recebidas	2
status	:active ou :failed	:active
updated_at	Timestamp Unix da última atualização	1730246400

Transições de Estado

Comportamento

Quando alertServiceCenter é recebido:

- Criar ou atualizar entrada do assinante
- Definir status = :active
- Registrar HLR GT
- Reiniciar ou preservar contadores de mensagens

Quando SRI-for-SM é bem-sucedido:

- Incrementar contador messages_sent
- Atualizar timestamp updated_at

Quando SRI-for-SM falha:

- Definir status = :failed
- Manter no rastreador para monitoramento

Quando o assinante é removido:

- Excluir da tabela ETS
- Não aparece mais na Interface Web

Interface Web - Página de Assinantes SMSc

Caminho: /smsc_subscribers **Atualização automática:** A cada 2 segundos

Nota: Esta página só está disponível quando executada no modo SMSc. Após descomentar a configuração SMSc em config/runtime.exs, você deve reiniciar a aplicação para que a rota se torne disponível.

A página **Assinantes SMSc** fornece monitoramento em tempo real de todos os assinantes rastreados:

Recursos

1. Tabela de Assinantes

- MSISDN, IMSI, HLR GT
- Contadores de mensagens enviadas/recebidas
- Distintivo de status (Ativo/Falhado) com codificação de cores
- Timestamp da última atualização e duração
- Botão de remoção para assinantes individuais

2. Estatísticas Resumidas

- Total de assinantes rastreados
- Contagem de assinantes ativos
- Contagem de assinantes falhados
- Número de HLRs únicos

3. Ações

- Limpar Tudo: Remover todos os assinantes rastreados
- Remover: Remover assinante individual

Exemplo de Visualização

Assinantes Rastreados SMSc				Total: 3
MSISDN	IMSI	HLR GT	Msgs S/R	Status
15551234567	001010123456789	15551111111	5/2	● Ativo
15559876543	001010987654321	15551111111	0/0	● Ativo
15551112222	001010111222233	15552222222	3/1	○ Falhado

Resumo: Total: 3 | Ativos: 2 | Falhados: 1 | HLRs Únicos: 2

Funções da API

O rastreador expõe essas funções para acesso programático:

```
# Chamado quando alertServiceCenter é recebido
SMSc.SubscriberTracker.alert_received(msisdn, imsi, hlr_gt)

# Incrementar contadores de mensagens
SMSc.SubscriberTracker.message_sent(msisdn)
SMSc.SubscriberTracker.message_received(msisdn)

# Marcar como falhado (falha no SRI-for-SM)
SMSc.SubscriberTracker.mark_failed(msisdn)

# Remover do rastreamento
SMSc.SubscriberTracker.remove_subscriber(msisdn)

# Funções de consulta
SMSc.SubscriberTracker.get_active_subscribers()
SMSc.SubscriberTracker.get_subscriber(msisdn)
SMSc.SubscriberTracker.count_subscribers()
SMSc.SubscriberTracker.clear_all()
```

Integração

O rastreador está automaticamente integrado com:

- **Manipulador de alertServiceCenter:** Chama `alert_received/3` na atualização de localização bem-sucedida
- **Manipulador de SRI-for-SM:** Incrementa `messages_sent` na roteamento bem-sucedido
- **Manipulador de assinante ausente:** Chama `mark_failed/1` quando o assinante está ausente
- **Erros de assinante desconhecido:** Chama `mark_failed/1` quando SRI-for-SM falha

Configuração de Auto-Flush da Fila de SMS

O serviço **Auto-Flush** processa automaticamente mensagens SMS pendentes.

Para referência de parâmetros de configuração, veja [Configuração de Auto-Flush na Referência de Configuração](#).

Configuração

```
config :omniss7,
  auto_flush_enabled: true,          # Habilitar/desabilitar auto-flush
```

```
auto_flush_interval: 10_000,      # Intervalo de polling em milissegundos
auto_flush_dest_smsc: nil,        # Filtro: nil = todos
auto_flush_tps: 10                # Máximo de transações por segundo
```

Como Funciona

1. **Polling:** A cada `auto_flush_interval` milissegundos, consulta a API para mensagens pendentes
2. **Filtragem:** Opcionalmente filtra por `auto_flush_dest_smsc`
3. **Limitação de Taxa:** Processa até `auto_flush_tps` mensagens por ciclo
4. **Entrega:** Para cada mensagem:
 - Envia **SRI-for-SM** (Enviar Informações de Roteamento para Mensagem Curta) para o HLR para obter informações de roteamento
 - O HLR retorna um IMSI sintético calculado a partir do MSISDN
 - O HLR retorna o endereço GT do SMSC onde o MT-ForwardSM deve ser enviado
 - Veja [Detalhes SRI-for-SM no Guia HLR](#) para documentação completa
 - Em caso de sucesso, envia **MT-forwardSM** para MSC/VLR
 - Atualiza o status da mensagem via API (entregue/falha)
 - Adiciona rastreamento de eventos via API

◈ **Mergulho Técnico:** Para uma explicação completa de como o SRI-for-SM funciona, incluindo mapeamento de MSISDN para IMSI, configuração de endereço GT do centro de serviço e geração de IMSI sintético que preserva a privacidade, veja a [seção SRI-for-SM no Guia de Configuração HLR](#).

Métricas SMS

Métricas Disponíveis

Métricas da Fila de SMS:

- `smc_queue_depth` - Número atual de mensagens pendentes
- `smc_messages_delivered_total` - Total de mensagens entregues com sucesso
- `smc_messages_failed_total` - Total de mensagens que falharam na entrega
- `smc_delivery_duration_milliseconds` - Histograma dos tempos de entrega

Consultas de Exemplo:

```
# Profundidade atual da fila
smc_queue_depth

# Taxa de sucesso de entrega (últimos 5 minutos)
rate(smc_messages_delivered_total[5m]) /
(rate(smc_messages_delivered_total[5m]) + rate(smc_messages_failed_total[5m]))

# Tempo médio de entrega
rate(smc_delivery_duration_milliseconds_sum[5m]) /
rate(smc_delivery_duration_milliseconds_count[5m])
```

Solução de Problemas SMS

Problema: Mensagens Não Entregues

Verificações:

1. Verifique se o auto-flush está habilitado
2. Verifique a conexão com o banco de dados
3. Monitore os logs em busca de erros
4. Verifique se a conexão M3UA está ATIVA
5. Verifique os limites de TPS

Problema: Alta Profundidade da Fila

Causas Possíveis:

- Limite de TPS muito baixo
- Problemas de tempo limite do HLR
- Problemas de conectividade de rede
- Números de destino inválidos

Soluções:

- Aumentar `auto_flush_tps`
- Verificar disponibilidade do HLR
- Revisar logs de mensagens falhadas

API MT-forwardSM

Enviar SMS via API

Endpoint da API: POST /api/MT-forwardSM

Solicitação:

```
{
  "imsi": "234509876543210",
  "destination_serviceCentre": "447999555111",
  "originating_serviceCenter": "447999123456",
  "smsPDU": "040B917477218345F600001570301857140C0BD4F29C0E9281C4E1F11A"
}
```

Resposta:

```
{
  "result": "success",
  "message_id": "12345"
}
```

Documentação Relacionada

Documentação OmniSS7:

- [← Voltar para a Documentação Principal](#)
- [Guia de Configuração HLR](#) - Configuração e operações do modo HLR
 - [Detalhes Técnicos SRI-for-SM](#) - Documentação completa sobre mapeamento de MSISDN para IMSI e configuração do centro de serviço
- [Guia de Recursos Comuns](#) - Interface Web, API, Monitoramento
- [Guia do Cliente MAP](#) - Operações MAP
- [Referência Técnica](#) - Especificações de protocolo

Documentação OmniMessage: Para configuração de roteamento de mensagens, gerenciamento de filas, rastreamento de entrega, limitação de taxa e análises, consulte a **documentação do produto OmniMessage**. O OmniMessage contém toda a lógica de roteamento de mensagens, algoritmos de repetição de filas, manipulação de relatórios de entrega e mecanismo de regras de negócios.

OmniSS7 por Omnitouch Network Services

Guia de Configuração do STP M3UA

[Vá para a Documentação Original](#)

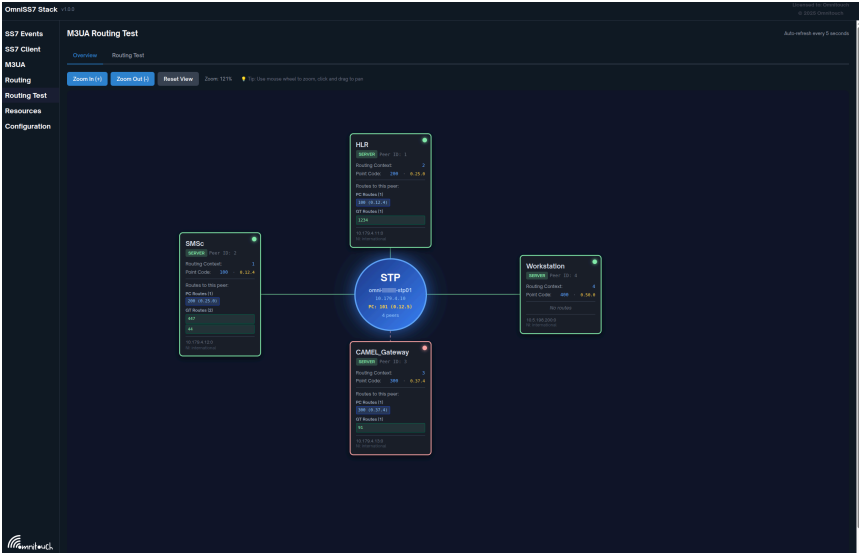
Este guia fornece uma configuração detalhada para usar o OmniSST como um **Ponto de Transferência de Sinalização (STP)**.

Índice

- [1. O que é um STP?](#)
- [2. Funções de Rede do STP](#)
- [3. Funções de Rede do STP](#)
- [4. Funções de Rede do STP](#)
- [5. Funções de Rede do STP](#)
 - [5.1. Funções de Rede do STP](#)
 - [5.2. Funções de Rede do STP](#)
- [6. Funções de Rede do STP](#)
- [7. Funções de Rede do STP](#)
- [8. Funções de Rede do STP](#)
- [9. Funções de Rede do STP](#)
- [10. Funções de Rede do STP](#)
- [11. Funções de Rede do STP](#)
- [12. Funções de Rede do STP](#)

O que é um Ponto de Transferência de Sinalização (STP)?

Um **Ponto de Transferência de Sinalização (STP)** é um elemento crítico da rede em redes de sinalização SST e baseado em IP que roteia mensagens de sinalização entre os nós da rede.



Funções do STP

- Rotamento de Mensagens:** Rotas o tráfego de sinalização SST com base no Código de Ponto de Destino (PC) ou Título Global (GT).
- Tradução de Protocolo:** Converte redes SST tradicionais com redes M3UA/SCTP baseadas em IP.
- Distribuição de Carga:** Distribui o tráfego entre múltiplos destinos usando roteamento baseado em prioridade.
- Gateway de Rede:** Conecta diferentes redes de sinalização e provedores de serviços.
- Ocultação de Topologia:** Pode reconhecer endereços para ocultar a topologia interna da rede.

Diagrama de Rede STP

Funções de Rede do STP Explicadas

ASP (Processo de Servidor de Aplicação)

- Função:** Cliente que se conecta a um SGP/STP remoto.
- Benefício:** Controla os dados.
- Caso de Uso:** Seu STP se conecta ao STP de uma rede parceira.

SGP (Processo de Gateway de Sinalização)

- Função:** Servidor que aceita conexões de ASPs.
- Benefício:** Controla os dados de entrada.
- Caso de Uso:** Redes parceiras se conectam ao seu STP.

AS (Servidor de Aplicação)

- Definição:** Aproximadamente igual a um ou mais ASPs.
- Objetivo:** Fornecer redundância e compartilhamento de carga.
- Caso de Uso:** Múltiplos ASPs atendendo ao mesmo destino.

Habilitando o Modo STP

O OmniSST pode operar em diferentes modos. Para usá-lo como um STP, você precisa habilitar o modo STP na configuração.

Mudando para o Modo STP

O `config/runtime.asx` do OmniSST contém três modos operacionais pré-configurados. Para habilitar o modo STP:

- Abra `config/runtime.asx`.
- Encontre as três seções de configuração (linhas 53-54):
 - Configuração 1: Modo STP (linhas 53-54).
 - Configuração 2: Modo HLR (linhas 57-58).
 - Configuração 3: Modo M3UA (linhas 125-126).
- Comente a configuração atualmente ativa (adicione # a cada linha).
- Descomente a configuração STP com o # das linhas 53-54.
- Personalize os parâmetros de configuração conforme necessário.
- Reinicie o aplicativo `asx -S asx`.

Core587 Stack

357 Events

M3UA

M3UA

Routing

Routing Test

Resources

Configuration

M3UA Status

Last updated: 2025-10-20 22:08:12 UTC

Refresh

Name	IPD	Status	ASP State	Assoc/SCTP	Local	Remote	RC	24h Uptime	Actions
"STP-CAMEL Gateway"	10.179.4.10	DOWN	down	down	10.179.4.10/2965	10.179.4.12/9	3	0%	<div></div>
"STP-HLR"	10.179.4.11	UP	active	established	10.179.4.10/2965	10.179.4.11/2965	2	100%	<div></div>

M3UA DETAILS

24-Hour Availability Timeline

Uptime: 100% Total Up: 40m Total Down: 20m

24 hrs

10 hrs

10 hrs

4 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

0 hrs

<

Configuração do Modo STP

A configuração completa do STP fica assim:

```
config complete(),
# tipo de modo: habilitar recursos STP apenas
m3ua_client_enabled: true,
hlr_mode_enabled: false,
work_mode_enabled: false,
```

- O protocolo SCTP (protocolo IP 132) deve ser permitido através dos firewalls
- A porta MJXA padrão é 2905 (padrão da indústria)
- Certifique-se de que há recursos suficientes do sistema para lidar com o tráfego de roteamento
- **Permissões de Roteamento:** Todas as rotas de rede geradas via o Interface Web ou API são armazenadas no banco de dados Mensa e sobrelivres a reinicialização
- **Mesclagem de Configuração:** As rotas de runtime, são as carregadas na inicialização e mescladas com as rotas de Mensa
- Após mudar de modo, o **canal** deve reiniciar a aplicação para que as alterações tenham efeito
- **Interface Web:** Consulte o [Guia da Interface Web](#) para obter mais informações
- **Acesso à API:** Consulte o [Guia de API](#) para documentação da API REST e acesso ao Swagger UI

Nota: Você pode habilitar tanto `map_client_n3ua` quanto `n3ua_stp` simultaneamente se precisar de conexões de saída e funcionalidade STP de entrada.

Melhor Prática: Use `rustine.exe` para configuração base e a Interface Web para alterações dinâmicas de rotas durante a operação.

```

# ping (continua)
# ping para:
# * Comando de saída para o STP Paroiro (apel: client)

peer id 1;
name "Partner STP_MST";
role client;
# Nome de saída
# * Identificar para o STP
# * Identificar para saída, server para entrada
# * Nome de saída
local_port 0;
remote_port 0, 1;
# S - atribuição de porta dinâmica
# 20 - no paramento
# 20 - no paramento
remote_port 2000;
routing control 1;
# Contato de rede para o STP
point_cost 100;
network_indicator: international

# * Comando para HLR Local (apel: client)

peer id 2;
name "Total HLR";
role local;
# Nome de saída
# * Identificar para o STP
# * Identificar para saída, server para entrada
local_port 0;
remote_port 0, 1;
# S - atribuição de porta dinâmica
# 20 - no paramento
# 20 - no paramento
remote_port 2000;
routing control 1;
# Contato de rede para o STP
point_cost 200;
network_indicator: international

# * Comando de entrada do MSC Remota (apel: server)
# Para o apel server, o STP aponta: comando de saída

peer id 3;
name "MSC Remota";
role server;
# Nome de saída
# * Identificar para o STP
# * Identificar para saída, server para entrada
local_port 0;
remote_port 0, 1;
# S - atribuição de porta dinâmica
# 20 - no paramento
# 20 - no paramento
remote_port 2000;
routing control 1;
# Contato de rede para o STP
point_cost 200;
network_indicator: international

```

```

    peer_id: 300,
    network_indicator: :international
  }
  # Conexão de entrada com porta de origem dinâmica (sem filtragem de porta)
  %
  peer_id: 4,
  name: "Dynamic_Client",
  role: :server,
  remote_ip: [10, 0, 0, 40], # IP de origem esperado
  remote_port: 4, # 0 = aceitar conexões de qualquer porta de origem
  routing_context: 4,
  point_code: 400,
  network_indicator: :international
}

```

OmniS57 Stack

SS7 Events

SS7 Client

MSUA

Routing

Routing Test

Resources

Configuration

MSUA Status

Last updated: 2025-10-20 23:58:12 UTC

Refresh

Name	IPD	Status	ASP Status	Assoc./SCTP	Local	Remote	RC	24h Uptime	Actions
"STP-CPEL_Selway"	-stp_peer	CONNECTED	down	down	10.179.4.10:2005	10.179.4.12:0	3		
"STP-ISA"	-stp_peer	UP	active	established	10.179.4.10:2005	10.179.4.10:2005	2	100%	

MSUA DETAILS

24-Hour Availability Timeline

Uptime: 100% Total Up: 48h Total Down: 23h 15m

10:00 10:00 10:00 10:00

Up Down

Basic Information

Name	STP-ISA	IPD	-stp_peer	Status	UP	Mode	server	ASP Status	active	Association Status	established
Routing Context	2										

Network Configuration

Local IP	10.179.4.10	Local Port	2005	Remote IP	10.179.4.12	Remote Port	2005
----------	-------------	------------	------	-----------	-------------	-------------	------

Additional Details

Peer ID	2	Role	server	Point Code	200
---------	---	------	--------	------------	-----

Raw Data (click to expand)

"STP-ISA"

-stp_peer	UP	active	established	10.179.4.10:2005	10.179.4.12:2005	1	100%
-----------	----	--------	-------------	------------------	------------------	---	------

"STP-Indication"

-stp_peer	UP	active	established	10.179.4.10:2005	10.179.4.10:2005	4	100%
-----------	----	--------	-------------	------------------	------------------	---	------

Parâmetros de Configuração do Par

Parâmetro	Tipo	Obrigatório	Descrição
peer_id	Integer	Sim	Identificador numérico único para o par
name	String	Sim	Nome legível por humanos para logs e monitoramento
role	Alenum	Sim	<:Local (indica se o par é servidor) ou <:remote (indica se o par é cliente)>
local_ip	Tuple	Sim (cliente)	Endereço IP local para vincular
local_port	Integer	Sim (cliente)	Porta local IP para vincular
remote_ip	Tuple	Sim	Endereço IP do par remoto
remote_port	Integer	Sim	Porta do par remoto IP para entrada - aceitar de qualquer porta de origem
routing_context	Integer	Sim	Identificador do contexto de roteamento MSUA
point_code	Integer	Sim	Código de ponto SS7 deste par
network_indicator	Alenum	Não	:international ou :national

Filtragem de Porta de Origem para Conexões de Entrada

Para conexões de entrada (papal: :server), o parâmetro remote_port controla a filtragem da porta de origem:

- Porta Específica** (ex: remote_port: 2005): Aceitar apenas conexões daquela porta de origem exata
 - Permite segurança adicional validando a porta de origem
 - Use quando o par remoto usar uma porta de origem fixa
- Qualquer Porta** (remote_port: 0): Aceitar conexões de qualquer porta de origem
 - Use quando o par remoto usa portas de origem dinâmicas/difusas
 - Apenas válida o endereço IP de origem
 - Mais flexível, mas igualmente menos seguro

Exemplo:

```

# Aceitar apenas de 10.5.100.200-2005 (porta específica)
%
peer_id: 1,
name: "Strict_Peer",
role: :server,
remote_ip: [10, 5, 100, 200],
remote_port: 2005,
# ... outras configurações
}

# Aceitar de 10.5.100.200 com qualquer porta de origem
%
peer_id: 2,
name: "Flexible_Peer",
role: :server,
remote_ip: [10, 5, 100, 200],
remote_port: 0, # Aceitar de qualquer porta de origem
# ... outras configurações
}

```

Suporte ao Protocolo MZPA

O OmniS57 suporta tanto os protocolos MSUA quanto MZPA para transporte de sinalização SS7.

O que é MZPA?

MZPA (Canais de Adaptação Peer-to-Peer da Indústria MTP) é um protocolo padronizado pelo ITU-T (RFC 4163) para transportar mensagens MTP3 sobre redes IP usando SCTP.

MSUA vs MZPA: Diferenças Chave

Recurso	MSUA	MZPA
Arquitetura	Cliente/Servidor (ASPS/DSC)	Peer-to-Peer
Caso de Uso	Gateway entre SS7 e IP	Link dentro de uma rede IP
Governança do Estado de Link	Nível de aplicação (ASPC/ASPN)	Então MTP3 (Abandono, Previa, Pronto)
Número de Inicializações	Um estabelecimento instantâneo	REINIT de 24 bits para reatuação
Implementação Típica	Gateway SS7 para IP, STP	Link de sinalização direta entre nós
RFC	RFC 4668	RFC 4163

Orientação para Seleção de Protocolo

Recomendação: Use MSUA por padrão. Use MZPA apenas quando especificamente necessário.

Quando Usar MSUA (Recomendado)

MSUA é o protocolo recomendado para a maioria das implantações:

- Implantações STP:** Implementações padrão de ponto de transferência de sinalização
- Funções de Gateway:** Conectando redes SS7 com sinalização baseada em IP
- Conexões de Elementos de Rede:** Conectando HLRs, MSCs, SMSCs e outros elementos de rede ao seu STP
- Gateway de Sinalização (SGSN):** Gateway centralizado para conexões de múltiplas Sinalizadores de Aplicação
- Implantações Interiores:** Arquiteturas centralizadas com controle centralizado
- Redes Multiplataforma:** Padrões da indústria amplamente suportados (RFC 4668)

Use MSUA para conectar elementos de rede (HLR, MSC, SMSC, VLR, etc.) ao seu STP.

Quando Usar MZPA (Apenas Casos Especiais)

MZPA deve ser usado apenas em cenários específicos:

- Links STP-to-STP:** Conexões diretas entre pontos entre Pontos de Transferência de Sinalização em uma rede multi-STP
- Substituição de TDM Legado:** Substituindo links tradicionais SS7 TDM quando o sistema remoto requer especificamente MZPA
- Compatibilidade MTP3 Necessária:** Ao conectar-se a sistemas legados que exigem precisamente do estado de link no estilo MTP3
- Requisito de Parceiro:** Quando um parceiro ou intermediário requer especificamente o protocolo MZPA

Importante: Não use MZPA para conectar elementos de rede (HLR, MSC, SMSC) ao seu STP - use MSUA em vez disso. MZPA é projetado para interconexões STP-to-STP onde ambos os lados operam como nós de roteamento.

Configurando Pares MZPA

Os pares MZPA são configurados da mesma forma que os pares MSUA, com um parâmetro adicional protocol:

Configuração de Par MZPA

Adicione pares MZPA à sua configuração (chamado peers) em config/routing.asx (sim, eles compartilham a mesma seção de configuração, apesar de serem protocolos diferentes):

Parâmetros Chave para MZPA:

Parâmetro	Valor	Descrição
protocol	mzpa	Especifica o protocolo MZPA (padrão é :chamado)
role	<:client ou :server>	Definição da conexão
local_ip	Inteiro	Porta SCTP local (a porta padrão do MZPA é 3103)
remote_ip	Inteiro	Porta SCTP remota (a porta padrão do MZPA é 3103)
point_code	Inteiro	Seu código de ponto
remote_port	Inteiro	Código de ponto do par remoto (especifico do MZPA)

Nota: O MZPA usa porta 3103 como padrão da indústria (diferente da porta 2005 do MSUA).

Estados de Link MZPA

Os links MZPA progressam através de vários estados durante a inicialização:

- Down:** Nenhum contato estabelecido
- Abandono:** Fase de inicialização inicial (~1 segundo)
- Previa:** Verificação da qualidade do link (~2 segundos)
- Pronto:** Link ativo e pronto para tráfego

A progressão do estado do link garante sinalização confiável antes que o tráfego seja trocado.

Gerenciando Pares MZPA via Interface Web

A página **Roteamento** na Interface Web fornece suporte total para gerenciar pares MZPA:

- Navegue até a página de Roteamento**
- Selecione o "Par"**
- Clique em "Adicionar Novo Par"**
- Insira "MZPA (RFC 4163)" no menu suspenso de Protocolo**
- Preencha a configuração de par:**
 - Nome do Par (identificador descritivo)
 - Protocolo: MZPA
 - Papel: cliente ou servidor
 - Código de Ponto (seu PC)
 - Endereço IP Remotamente
 - Portas locais/remotas (Especificamente 3103 para MZPA)
 - Indicador de Rede (internacional ou nacional)
- Clique em "Salvar Par"**

A tabela de pares exibe o tipo de protocolo com codificação de cores:

- Amarelo:** pares MSUA
- Verde:** pares MZPA

Comportamento de Roteamento MZPA

Os pares MZPA se integram perfeitamente ao sistema de roteamento do OmniS57:

- Rota de Código de Ponto:** Funcionam de forma idêntica para MZPA e MSUA
- Rota de Tabela Global:** Tratamento suportado em links MZPA

Você pode combinar rotas específicas com rotas mascaradas para um roteamento flexível:

Melhores Práticas

- ### Configurando o Roteamento de Título Global (GT)

O roteamento de Título Global permite o **roteamento baseado em conteúdo** usando números de telefone ou valores IMSI em vez de códigos de posto. Para tradução avançada de endereços de Título Global com base na parte chamadora/recebadora, consulte o [Guia de NAT de Título Global](#).

Pré-requisitos

- Habilitar roteamento GT: `enable_gt_routing: true` em `config/runtime.xml`

```
config :omiss7.
```

Lógica de Roteamento GT

O algoritmo de roteamento GT segue este processo de decisão:

Etapas de Roteamento:

1. **Correspondência do Prefixo Mais Longo:** O STP encontra todas as rotas GT onde o prefixo corresponde ao início do Título Global

- Exemplo: GT "447712345678" corresponde a "44" e "447", mas "447" vence (correspondência mais longa)

3. **Correspondência de TT/NP/NAI (Opcional):**
- Se **source** TT, **source** NPI ou **source** NAI forem especificados, as rotas devem corresponder a esses indicadores
 - Valores n/11 assumem coringas (correspondem a qualquer valor)
4. **Seleção Baseada em Especificidade:**
- Rotas com critérios de correspondência mais específicos vencerem sobre coringas
 - Ordem de prioridade: Comprimento do Prefixo GT = SSN > TT > NPI > NAI = Número de Prioridade
5. **Reescrita de Indicadores (Opcional):**
- Se **dest_1st**, **dest_1t**, **dest_1pi** ou **dest_nai** forem especificados, o STP reescreverá esses indicadores
 - Usado para normalização de prefixos e interconexão de rede
6. **Fallback para Códigos de Ponto:**
- Se nenhuma rota GT corresponder, o STP recorre ao roteamento de Código de Ponto usando o DPC

Roteamento Avançado: Tipo de Tradução, NPI e NAI

Além da correspondência de prefixo GT e SSN, o STP aponta roteamento e transformação com base nos indicadores de Título Global SCP:

- **Tipo de Tradução (TT)** Identifica o plano de numeração e o tipo de endereço
- **Indicador de Plano de Numeração (NPI)** Define o plano de numeração (ex: ISDN, Dados, Tóxico)
- **Indicador de Natureza do Endereço (NAI)** Especifica o formato do endereço (ex: Internacional, Nacional, Autônomo)

OmniSST Stack STP 1.0.5

SS7 Events

System Logs

SS7 Client

MSUA

Routing

Routing Test

Resources

Configuration

MSUA Routing Management

Peer 01Peer Code Routes 00Global Title Routes 01

Edit Global Title Route

Routing

Routing Test

Resources

Configuration

Peer 01

DTCP - Discard traffic [newer] keep[]

DTCP is enabled. Discard traffic to this Global Title used for processing routing request

DTCP is optional

Leave empty to preserve

Route to SSN when forwarding (empty = keep original)

Description

Index numbers

Source SSN (optional)

Leave empty for wildcard

Make an Global Peer SSN empty = n/11

Priority

1

Hide Advanced Routing (TT/NP/NAI)

Configure routing based on SCP Global Title indicators. Leave empty for wildcard matching

Source TT (Match)

0-100 or empty

Source SSN (Match)

0-100 or empty

Source NPI (Match)

0-10 or empty

Source NAI (Match)

0-100 or empty

Source of Address Indicator (empty = n/11)

Common Values Reference

TT=0-Unknown, 1=ISDN, 6-Mobile

NAI=0-Unknown, 1=National, 4=ISN

Update RouteCancel

GT Prefix	Source SSN	TT/NP/NAI	Peer ID	Peer	Dest SSN	Priority	Description	Actions
91	n/11	-	3	MSC South (3)	preserve	1	Index numbers	Remove Edit Delete
1234	n/11	-	1	STP West (1)	preserve	1	US numbers	Remove Edit Delete
447	n/11	-	2	STP East (2)	preserve	1	UK mobile numbers	Remove Edit Delete
44	n/11	-	2	STP East (2)	preserve	1	UK numbers	Remove Edit Delete

Common SSN Values

0 - HLR Home Location Register

6 - MSC Mobile Switching Center

10 - AUC (Authentication Center)

(n) - nonSCP (Special Control Function)

1 - HLR Mobile Location Register

6 - EIR (Equipment Identity Register)

1cc - Network

1cc - SSN

Correspondência (Indicadores de Origem)

As rotas podem corresponder aos indicadores de mensagens de entrada:

- **source** TT: Correspondência de mensagens com Tipo de Tradução específico
- **source** NAI: Correspondência de mensagens com Indicador de Plano de Numeração específico
- **source** NPI: Correspondência de mensagens com Indicador de Natureza do Endereço específico
- **Value** n/11 = coringa (corresponde a qualquer valor)

Transformação (Indicadores de Destino)

As rotas podem reescrever indicadores ao encaminhá-las:

- **dest_1t**: Transformar Tipo de Tradução para novo valor
- **dest_npi**: Transformar Indicador de Plano de Numeração para novo valor
- **dest_nai**: Transformar Indicador de Natureza do Endereço para novo valor
- **Value** n/11 = preservar o valor original (sem transformação)

Seleção Baseada em Especificidade

Quando várias rotas correspondem, a rota mais específica é selecionada usando esta ordem de prioridade:

1. Correspondência de prefixo GT mais longo
2. SSN específico sobre SSN coringa
3. TT específico sobre TT coringa
4. NPI específico sobre NPI coringa
5. NAI específico sobre NAI coringa
6. Menor número de prioridade

Exemplos de Configuração

```
config omitall;
enable_gtp_routing True;

# Exemplo 1: Correspondência e transformação de Tipo de Tradução
N;
gt_prefix: "447";
peer_id: 1;
source_tti: 4; # Correspondência TT=6 (Desconhecido)
dest_tt: 3; # Transformar para TT=3 (Nacional)
priority: 1;
description: "Número do Reino Unido: TT 6->3 transformação"
};

# Exemplo 2: Correspondência de NPI específico e transformação de NAI
N;
gt_prefix: "11";
peer_id: 2;
source_npi: 1; # Correspondência NPI=1 (ISDN/Teléfono)
source_nai: 4; # Correspondência NAI=4 (Internacional)
dest_nai: 3; # Transformar para NAI=3 (Nacional)
priority: 1;
description: "Número dos EUA: Internacional->Nacional NAI"
};

# Exemplo 3: Roteamento combinado de SSN e indicadores
N;
gt_prefix: "131";
source_ssn: 8; # Correspondência de tráfego de SMS
source_tti: 6; # Correspondência TT=6
dest_ssn: 6; # Reescrever SSN para HLR
dest_tt: 2; # Transformar para TT=2
dest_npi: 1; # Definir NPI=1 (ISDN)
dest_nai: 4; # Definir NAI=4 (Internacional)
peer_id: 3;
priority: 1;
description: "SMS Francês: Normalização completa"
};

# Exemplo 4: TT coringa, NPI específico
N;
gt_prefix: "440";
source_tti: n/11; # Correspondência de qualquer TT (coringa)
source_npi: 6; # Correspondência NPI=6 (Dados)
dest_npi: 1; # Transformar para NPI=1 (ISDN)
peer_id: 4;
priority: 1;
description: "Normalização da rede de dados island"
};
}
```

Valores Comuns de TT/NP/NAI

Tipo de Tradução (TT):

- 0 = Desconhecido
- 1 = Internacional
- 2 = Nacional
- 3 = Especifico da Rede

Indicador de Plano de Numeração (NPI):

- 0 = Desconhecido
- 1 = ISDN/Frêquência (E.164)
- 2 = Dados (X.21)
- 4 = Tóxico (E.20)
- 6 = Móvel Terrestre (E.212)

Indicador de Natureza do Endereço (NAI):

- 0 = Desconhecido
- 1 = Número de Assinante
- 2 = Reservado para Uso Nacional
- 3 = Número Significativo Nacional
- 4 = Número Internacional

Exemplo de Decisão de Roteamento

Para uma mensagem de entrada com:

- GTI "44712345678"
- SSN: 8
- TT: 0
- NPI: 1
- NAI: 4

Com estas rotas configuradas:

```
# Rota A: TT coringa
%gt_prefix: "447", peer_id: 1, priority: 1;

# Rota B: TT específico
%gt_prefix: "447", source_tti: 8, peer_id: 2, priority: 1;

# Rota C: TT específico + NPI
%gt_prefix: "447", source_tti: 8, source_npi: 1, peer_id: 3, priority: 1;
```

Resultado: A Rota C é selecionada (mais específica: corresponde a GT + TT + NPI)

A mensagem é encaminhada com indicadores transformados de acordo com os valores de **dest_tt**, **dest_npi**, **dest_nai** de Rota C.

Exemplos de Roteamento GT

GT Chamado	SSN de Origem	TT/NP/NAI	Rota Correspondente	Razão
44712345678	-	-	"447" - par 3	Correspondência de prefixo mais longo
441234567890	-	-	"44" - par 1	Correspondência de prefixo, sem rota mais específica
1212551234	-	-	"1" - par 2	Correspondência de prefixo para sinais do EUA
5558812345678	-	-	"555" (SSN 8) - par 4	Correspondência de GT + SSN, reescreve SSN para 6
5558812345678	-	-	"555" SSN coringa - par X	Correspondência de GT sem reescrita de SSN
441234567890	0	4	"44" (TT=0) - par 1	Correspondência de GT + TT, transforma TT para 3
1212551234	8	0	4 "1" (TT=0, NPI=1, NAI=4)	Mais específico: correspondência de GT+TT+NPI+SSN

Casos de Uso Comuns para Roteamento TT/NP/NAI

1. Normalização de Interconexão de Rede

- Redes diferentes podem usar convenções de indicadores diferentes
- Transformar indicadores no ponto de interconexão para garantir compatibilidade
- Exemplo: Rede precisa usar TT=2 para internacional, mas rede usa TT=4

2. Conversão de Protocolo

Converte entre planos de numeração ao rotar entre diferentes tipos de rede
Exemplo: Rotas de uma rede móvel (NPN=0) para PSTN (NPN=1)
3. Padronização de Formato de Endereços

Normalizar todo o tráfego de entrada para usar valores de NAI consistentes
Exemplo: Converter todos os formatos internacionais (NAI=0) para formato nacional (NAI=3) para roteamento doméstico
4. Roteamento Específico de Operadores

Rotas com base no tipo de tráfego para diferentes provedores de serviços
Exemplo: Tráfego rotas para o Operador A, Tráfego rotas para o Operador B
5. Integração de Sistemas Legados

Sistemas modernos podem usar valores de indicadores diferentes dos sistemas legados
Transformar os SIP para manter compatibilidade retroativa

Recursos de Gerenciamento de Rota

Desabilitando Rotas

As rotas podem ser desabilitadas temporariamente sem serem excluídas. Isso é útil para testes, manutenção ou gerenciamento de tráfego.

Flag Habilitada

Tanto as rotas de Código de Ponto quanto as rotas de Título Global suportam uma flag enabled opcional.

```
config omit17;  
alias routes: 1  
  # Rota ativa  
  #  
  # dest_pc: 100,  
  # peer_id: 1,  
  # priority: 1,  
  # network_indicator: international,  
  # enabled: true # Rota está ativa (padrão se omitido)  
  }  
  # Rota desabilitada (não avaliada durante o roteamento)  
  #  
  # dest_pc: 200,  
  # peer_id: 2,  
  # priority: 1,  
  # network_indicator: international,  
  # enabled: false # Rota está desabilitada  
  }  
  }  
alias gt_routes: 1  
  # Rota GT desabilitada  
  #  
  # gt_prefix: "44",  
  # peer_id: 1,  
  # priority: 1,  
  # description: "Número do Reino Unido - temporariamente desabilitada",  
  # enabled: false  
  }  
  }
```

Comportamento Padrão

- Se enabled não for especificado, as rotas padrão são enabled: true
- Rotas desabilitadas não completamente ignoradas durante a busca de rotas
- Use a Interface Web para alterar rotas ouclid sem editar a configuração

Causa de Uso

- Testando alterações no fluxo de tráfego
- Isolando falhas de manutenção temporárias
- Testes A/B de diferentes caminhos de roteamento
- Implementação gradual de novas rotas

Rotas DROP - Prevenindo Loops de Roteamento

Rotas DROP (com peer_id: 0) descartam silenciosamente o tráfego em vez de encaminhá-lo, isso previne loops de roteamento e permite filtragem avançada de tráfego.

Configurando Rotas DROP

```
config omit17;  
alias routes: 1  
  # Rota DROP para código de ponto específico  
  #  
  # dest_pc: 999,  
  # peer_id: 0, # peer_id=0 significa DROP  
  # priority: 1,  
  # network_indicator: international  
  }  
  }  
alias gt_routes: 1  
  # Rota DROP para prefixo GT  
  #  
  # gt_prefix: "999",  
  # peer_id: 0, # peer_id=0 significa DROP  
  # priority: 99, # peer_id=0 significa "Bloqueio intervalo de teste"  
  }  
  }
```

Como Funcionam as Rotas DROP

Quando uma mensagem corresponde a uma rota DROP:

- O encaminhamento de mensagem identifica peer_id: 0
- A mensagem é **silenciosamente descartada** (não encaminhada)
- Um **log de INFO** gerado "Rota DROP correspondente para DPC 999" ou "Rota DROP correspondente para GT 999"
- A busca de roteamento retorna [error, :dropped]

Importante: O tráfego descartado é registrado no nível INFO para monitoramento e solução de problemas.

Causa de Uso Comum: Lista Branca de Prefixos

Um dos usos mais poderosos das rotas DROP é a **lista branca de prefixos** - permitindo apenas números específicos dentro de um grande intervalo enquanto bloqueia todos os outros.

O Padrão

- Crie uma rota DROP para todo o prefixo com **número de prioridade alta** (ex: 99)
- Crie rotas de permissão específicas para números individuais com **números de prioridade baixos** (ex: 1)
- Como números de prioridade mais baixos são avaliados primeiro, rotas permissão correspondem rotas da rota DROP
- Qualquer número não explicitamente permitido é capturado pela rota DROP

Exemplo de Configuração

Você tem um prefixo GT 1234 que representa um intervalo de 10.000 números (1234000000 - 1234999999), mas você deseja restringir apenas 3 números específicos: 1234567890, 1234555000 e 123411222.

```
config omit17;  
alias gt_routes: 1  
  # Rota DROP com PRIORIDADE ALTA (avaliada por último)  
  #  
  # gt_prefix: "1234",  
  # peer_id: 0, # DROP  
  # priority: 99,  
  # description: "Bloqueio todo 1234, exceto números na lista branca"  
  }  
  # Rotas de permissão específicas com NÚMEROS DE PRIORIDADE BAIXOS (avaliadas primeiro)  
  #  
  # gt_prefix: "1234567890",  
  # peer_id: 1,  
  # priority: 1,  
  # description: "Número permitido 1"  
  }  
  #  
  # gt_prefix: "123455000",  
  # peer_id: 1,  
  # priority: 1,  
  # description: "Número permitido 2"  
  }  
  #  
  # gt_prefix: "123411222",  
  # peer_id: 1,  
  # priority: 1,  
  # description: "Número permitido 3"  
  }  
  }
```

Comportamento de Roteamento:

GT de Entrada	Rotas Correspondentes	Rota Selecionada	Ação
1234567890	o "1234567890" (prioridade 1), "1234567890" (mais específica, maior prioridade)	Roteado para o par 1	
123455000	o "1234" DROP (prioridade 99) o "123455000" (prioridade 3), "123455000" (mais específica, maior prioridade)	Roteado para o par 1	
123411222	o "1234" DROP (prioridade 99) o "123411222" (prioridade 1), "123411222" (mais específica, maior prioridade)	Roteado para o par 1	
1234999999	o "1234" DROP (prioridade 99)"1234" DROP (única correspondência)	Descartado e registrado	
1234000000	o "1234" DROP (prioridade 99)"1234" DROP (única correspondência)	Descartado e registrado	

- Resultado:**
- o Apenas 3 números específicos são roteados para o par 1
 - o Todos os outros números 1234* são silenciosamente descartados
 - o Todo o tráfego descartado é registrado para monitoramento

Logs Gerados:

[INFO] Rota DROP correspondente para GT 1234999999

[INFO] Rota DROP correspondente para DPC 1020

Rotas DROP para Códigos de Ponto

O mesmo padrão de lista branca funciona para roteamento de Código de Ponto:

```
config omit17;  
alias routes: 1  
  # DROP todo o intervalo /8 (64 códigos de ponto: 1000-1063)  
  #  
  # dest_pc: 1000,  
  # peer_id: 0,  
  # priority: 99,  
  # mask: 8,  
  # network_indicator: international  
  }  
  # Permitir PCs específicos  
  #  
  # dest_pc: 1010, peer_id: 1, priority: 1, network_indicator: international,  
  # dest_pc: 1020, peer_id: 1, priority: 1, network_indicator: international,  
  # dest_pc: 1030, peer_id: 1, priority: 1, network_indicator: international  
  }
```

Resultado: Apenas os PCs 1010, 1020 e 1030 são roteados. Todos os outros PCs no intervalo 1000-1063 são descartados.

Monitorando Rotas DROP

Verifique os Logs:

```
# Monitore tráfego descartado  
tail -f logs/app-log | grep "Rota DROP correspondente"
```

Saida esperada:

```
[INFO] Rota DROP correspondente para GT 1234999999  
[INFO] Rota DROP correspondente para DPC 1020
```

Via Interface Web:

- Navegue até a aba **Logs do Sistema**
- Filtre por nível INFO
- Pesquise por "Rota DROP correspondente"

Melhores Práticas:

- o Monitore logs regularmente para garantir que rotas DROP não estão bloqueando tráfego legítimo
- o Use sempre descrições descritivas para documentar por que as rotas são descartadas
- o Use números de prioridade altos (99) para rotas DROP para garantir que sejam rotas catch-all
- o Teste o comportamento da rota DROP antes de implantar em produção em produção
- o Configure alertas para eventos inesperados no tráfego descartado

Roteamento Avançado: Roteamento Baseado em SSN e Reescrita

Números de Subsistema (SSN)

Os Números de Subsistema identificam a chamada de aplicação:

- SSN 0:** HLR (Departamento de Localização do Usuário)
- SSN 1:** VLR (Departamento de Localização do Visitante)
- SSN 8:** SMSC (Centro de Mensagens Móveis) / SMS (Centro de SMS)
- SSN 9:** CMSC (Centro de Mensagens Móveis Gateway)

Exemplo de Roteamento Baseado em SSN

Rotas e tráfego de SMS para diferentes HLRs com base no prefixo do número:

```
alias gt_routes: 1  
  # Rota SMS para números do Reino Unido para HLR do Reino Unido, reescrevendo SSN de 8 (SMSC) para 6 (HLR)  
  #  
  # gt_prefix: "44",  
  # peer_id: 0, # Correspondência de SSN de entrada 8 (SMSC)  
  # peer_id: 1,  
  # dest_pc: 6,  
  # priority: 1,  
  # description: "SMS do Reino Unido para HLR"
```

```
}
# Envia tráfego de voz para números de Reino Unido (558 6) sem rescrita
%t_prefix: 149,
source: 558 6,
peer_id: 1,
dest: 558 6,
peer_id: 1,
description: "Tráfego de voz do Reino Unido"
}
```

Testando a Configuração de Roteamento STP

Após configurar pares e rotas, verifique sua configuração:

1. Verifique o Status do Par

Via Interface Web:

- Navegue até [Status de Status](#).
- Verifique a página de Status MSUA.
- Confirme que os pares mostram **Status: ATIVO**.

Via Console CLI:

```
# Obtenha todos os status dos pares
MSUA_STP.get_peers_status()

# Saída esperada:
# [
#   {peer_id: 1, name: "Partner_STP_Mest", status: active, point_code: 100, ...},
#   {peer_id: 2, name: "Local_HLR", status: active, point_code: 200, ...}
# ]
```

2. Teste o Roteamento de Código de Ponto

```
# Envie uma mensagem MSUA de teste para DPC 100
test_payload = <1> 2, 3, 00 # Payload fictício
MSUA_STP.route_by_pc(100, test_payload, 0)

# Verifique os logs para a decisão de roteamento
# Log esperado: "Enviando mensagem: DPC=... => DPC=100 via par 1"
```

Omni527 Stack

MSUA Routing Test

SS7 Events

SS7 Client

MSUA

Routing

Routing Test

Resources

Configuration

Message Parameters

Source Point Code (DPC)	Destination Point Code (DPC)
e.g. 123	e.g. 456
Calling Global Title	Called Global Title
e.g. 1234567890	e.g. 8076543210
Calling SSN (optional)	Called SSN (optional)
e.g. 6 HLR, 7 VLR, 8 MSC	e.g. 6 HLR, 7 VLR, 8 MSC

Test Routing

Common SSN Values

6 - HLR (Home Location Register)	7 - VLR (Visitor Location Register)
8 - MSC (Mobile Switching Center)	9 - ER (Equipment Identity Register)
10 - MSC (Mobile Switching Center)	11 - HLR
12 - gsmSCF (Service Control Function)	14 - SSN

3. Teste o Roteamento de Título Global

```
# Procure rota GT manualmente
MSUA_Routing.lookup_gw_by_gt("44772245678")

# Saída esperada:
# [pk, [msua_peer, 3, "MR_Mobile_Peer", ...], n13]

# Procure rota GT com SSN
MSUA_Routing.lookup_gw_by_gt("55881234567", 8)

# Saída esperada com rescrita de SSN:
# [pk, [msua_peer, 4, "MR_HLR_Peer", ...], 6]
```

4. Monitore as Métricas de Roteamento

Acesso às métricas de Prometheus em /metrics

Métricas chave:

```
# Mensagens recebidas por par
msua_stp_messages_received_total{peer_name="Partner_STP_Mest", point_code="100"} 1523

# Mensagens enviadas por par
msua_stp_messages_sent_total{peer_name="Local_HLR", point_code="200"} 1408

# Falhas de roteamento
msua_stp_routing_failures_total{reason="no_route"} 5
msua_stp_routing_failures_total{reason="no_peer_route"} 2
```

Métricas e Monitoramento STP

Métricas Disponíveis

Métricas de Tráfego por Par:

- msua_stp_messages_received_total: Total de mensagens recebidas de cada par
- msua_stp_messages_sent_total: Total de mensagens enviadas para cada par
- msua_stp_messages_sent_total



Guia da Interface Web

[← Voltar à Documentação Principal](#)

Este guia fornece documentação abrangente para o uso da **Interface Web** do OmniSS7 (interface Phoenix LiveView).

Índice

1. [Visão Geral](#)
 2. [Acessando a Interface Web](#)
 3. [Página de Gerenciamento de Roteamento](#)
 4. [Página de Assinantes Ativos](#)
 5. [Operações Comuns](#)
 6. [Comportamento de Atualização Automática](#)
-

Visão Geral

A Interface Web do OmniSS7 é uma aplicação **Phoenix LiveView** que fornece capacidades de monitoramento e gerenciamento em tempo real. As páginas disponíveis dependem do modo operacional ativo (STP, HLR ou SMSs).

Arquitetura da Interface Web

Configuração do Servidor

- **Protocolo:** HTTPS
- **Porta:** 443 (configurada em config/runtime.exs)
- **IP Padrão:** 0.0.0.0 (escuta em todas as interfaces)
- **Certificados:** Localizados em priv/cert/

URL de Acesso: https://[server-ip]:443

Acessando a Interface Web

Pré-requisitos

1. **Certificados SSL:** Certifique-se de que certificados SSL válidos estão presentes em priv/cert/:

- omnitouch.crt - Arquivo de certificado
- omnitouch.pem - Arquivo de chave privada

2. **Aplicação em Execução:** Inicie a aplicação com `iex -S mix`

3. **Firewall:** Certifique-se de que a porta 443 está aberta para tráfego HTTPS

Páginas Disponíveis por Modo

Página	Modo STP	Modo HLR	Modo SMSc	Descrição
Eventos SS7	?	?	?	Registro de eventos e captura de mensagens SCCP
Cliente SS7	?	?	?	Teste manual de operações MAP
M3UA	?	?	?	Status da conexão M3UA
Roteamento	?	?	?	Gerenciamento da tabela de roteamento M3UA
Teste de Roteamento	?	?	?	Teste e validação de rotas
Links HLR	?	?	?	Status da API HLR e gerenciamento de assinantes
Assinantes Ativos	?	?	?	Rastreamento em tempo real da localização de assinantes (HLR)
Links SMSc	?	?	?	Status da API SMSc e gerenciamento de filas
Assinantes SMSc	?	?	?	Rastreamento em tempo real de assinantes (SMSc)
Aplicação	?	?	?	Recursos do sistema e monitoramento
Configuração	?	?	?	Visualizador de configuração

Gerenciamento de Roteamento

Página: /routing

Modos: STP, SMSc

Atualização Automática: A cada 5 segundos

A página de Gerenciamento de Roteamento fornece uma interface com abas para gerenciar tabelas de roteamento M3UA.

Layout da Página

Aba de Pares

Gerencie conexões de pares M3UA (outros STPs, HLRs, MSCs, SMSCs).

Colunas da Tabela de Pares

Coluna	Descrição	Exemplo
ID	Identificador único do par	1
Nome	Nome legível por humanos do par	"STP_West"
Função	Função da conexão	client, server, stp
Código de Ponto	Código de ponto SS7 do par	100
Remoto	IP:Porta remota	10.0.0.10:2905
Status	Status da conexão	active, aspup, down
Ações	Botões Editar/Excluir	-

Adicionando um Par

1. **Clique** na aba Pares
2. **Preencha** os campos do formulário:
 - **ID do Par:** Gerado automaticamente se deixado em branco
 - **Nome do Par:** Nome descritivo (obrigatório)
 - **Função:** Selecione client, server ou stp
 - **Código de Ponto:** Código de ponto SS7 (obrigatório)
 - **IP Local:** Endereço IP do seu sistema
 - **Porta Local:** 0 para atribuição dinâmica de porta
 - **IP Remoto:** Endereço IP do par
 - **Porta Remota:** Porta do par (tipicamente 2905)
 - **Contexto de Roteamento:** ID do contexto de roteamento M3UA
 - **Indicador de Rede:** international ou national
3. **Clique** em "Adicionar Par"

Persistência: O par é imediatamente salvo no Mnesia e sobrevive a reinicializações.

Editando um Par

1. **Clique** no botão "Editar" na linha do par
2. **Modifique** os campos do formulário conforme necessário
3. **Clique** em "Atualizar Par"

Nota: Se você alterar o ID do Par, o par antigo será excluído e um novo será criado.

Excluindo um Par

1. **Clique** no botão "Excluir" na linha do par
2. **Confirme** a exclusão (todas as rotas que usam este par também serão removidas)

Indicadores de Status do Par

Status	Cor	Descrição
active	Verde	Par está conectado e roteando mensagens
aspup	Amarelo	ASP está ativo, mas ainda não está em operação
down	Vermelho	Par está desconectado

Aba de Rotas de Código de Ponto

Configure regras de roteamento com base em Códigos de Ponto de Destino.

Colunas da Tabela de Rotas

Coluna	Descrição	Exemplo
PC de Destino	Código de ponto alvo (formato zone.area.id)	1.2.3 (100)
Máscara	Máscara de sub-rede para correspondência de PC	/14 (exato), /8 (intervalo)
ID do Par	Par alvo para esta rota	1
Nome do Par	Nome do par alvo	"STP_West"
Prioridade	Prioridade da rota (1 = mais alta)	1
Rede	Indicador de rede	international
Ações	Botões Editar/Excluir	-

Adicionando uma Rota de Código de Ponto

1. **Clique** na aba "Rotas de Código de Ponto"
2. **Preencha** os campos do formulário:
 - **Código de Ponto de Destino:** Insira como zone.area.id (por exemplo, 1.2.3) ou inteiro (0-16383)
 - **Máscara:** Selecione máscara /14 para correspondência exata, valores menores para intervalos
 - **ID do Par:** Selecione o par alvo no dropdown
 - **Prioridade:** Insira prioridade (1 = mais alta, padrão)
 - **Indicador de Rede:** Selecione international ou national
3. **Clique** em "Adicionar Rota"

Formato do Código de Ponto: Você pode inserir códigos de ponto em dois formatos:

- **Formato 3-8-3:** zone.area.id (por exemplo, 1.2.3)
- **Formato Inteiro:** 0-16383 (por exemplo, 1100)

O sistema converte automaticamente entre formatos.

Entendendo Máscaras

Códigos de ponto são valores de 14 bits (0-16383). A máscara especifica quantos bits mais significativos devem corresponder:

Máscara	PCs Correspondidos	Caso de Uso
/14	1 (correspondência exata)	Roteamento para destino específico
/13	2 PCs	Pequeno intervalo
/8	64 PCs	Intervalo médio
/0	Todos os 16.384 PCs	Rota padrão/catch-all

Exemplos:

- PC 1000 /14 → Corresponde apenas ao PC 1000
- PC 1000 /8 → Corresponde ao PC 1000-1063 (64 PCs consecutivos)
- PC 0 /0 → Corresponde a todos os códigos de ponto (rota padrão)

Cartão de Referência de Máscara de Código de Ponto

A página da web inclui uma referência interativa mostrando todos os valores de máscara e seus intervalos.

Aba de Rotas de Título Global

Configure regras de roteamento com base em endereços de Título Global SCCP.

Requisito: O roteamento de Título Global deve ser habilitado na configuração:

```
config :omniss7,  
  enable_gt_routing: true
```

Colunas da Tabela de Rotas

Coluna	Descrição	Exemplo
Prefixo GT	Prefixo GT da parte chamada (vazio = fallback)	"1234", ""
SSN de Origem	Correspondência no SSN da parte chamada (opcional)	6 (HLR), any
ID do Par	Par alvo	1
Par	Nome do par	"HLR_West (1)"
SSN de Destino	Reescrever SSN ao encaminhar (opcional)	6, preserve
Prioridade	Prioridade da rota	1
Descrição	Descrição da rota	"Números dos EUA"
Ações	Botões Editar/Excluir	-

Adicionando uma Rota de Título Global

1. **Clique** na aba "Rotas de Título Global"
2. **Preencha** os campos do formulário:
 - **Prefixo GT:** Deixe vazio para rota de fallback, ou insira dígitos (por exemplo, "1234")
 - **SSN de Origem:** Opcional - filtrar pelo SSN da parte chamada
 - **ID do Par:** Selecione o par alvo
 - **SSN de Destino:** Opcional - reescrever SSN ao encaminhar
 - **Prioridade:** Prioridade da rota (1 = mais alta)
 - **Descrição:** Descrição legível por humanos
3. **Clique** em "Adicionar Rota"

Rotas de Fallback: Se o Prefixo GT estiver vazio, a rota atua como um catch-all para GTs que não correspondem a nenhuma outra rota.

Valores Comuns de SSN

A página inclui um cartão de referência com valores comuns de SSN:

SSN	Elemento de Rede
6	HLR (Registro de Localização do Lar)
7	VLR (Registro de Localização do Visitante)
8	MSC (Centro de Comutação Móvel)
9	EIR (Registro de Identidade de Equipamento)
10	AUC (Centro de Autenticação)
142	RANAP
145	gsmSCF (Função de Controle de Serviço)
146	SGSN

Reescrita de SSN

- **SSN de Origem:** Correspondência no SSN da Parte Chamadora em mensagens recebidas
- **SSN de Destino:** Se definido, reescreve o SSN da Parte Chamadora ao encaminhar
 - Vazio = preservar SSN original
 - Valor = substituir por este SSN

Caso de Uso: Roteie mensagens com SSN=6 (HLR) para um par e reescreva para SSN=7 (VLR) no lado de saída.

Persistência da Tabela de Roteamento

Todas as rotas são armazenadas no Mnesia e sobrevivem a reinicializações

da aplicação.

Como as Rotas Persistem

1. **Mudanças na Interface Web:** Todas as operações de adicionar/editar/excluir são imediatamente salvas no Mnesia
2. **Reinicialização da Aplicação:** As rotas são carregadas do Mnesia na inicialização
3. **Mesclagem do runtime.exs:** Rotas estáticas de config/runtime.exs são mescladas com rotas do Mnesia (sem duplicatas)

Prioridade da Rota

Quando várias rotas correspondem a um destino:

1. **Mais Específico Primeiro:** Valores de máscara mais altos (mais específicos) têm precedência
2. **Campo de Prioridade:** Números de prioridade mais baixos roteiam primeiro (1 = maior prioridade)
3. **Status do Par:** Apenas rotas para pares active são usadas

Assinantes Ativos

Página: /subscribers

Modo: Apenas HLR

Atualização Automática: A cada 2 segundos

Exibe rastreamento em tempo real de assinantes que enviaram solicitações de UpdateLocation.

Recursos da Página

Colunas da Tabela de Assinantes

Coluna	Descrição	Exemplo
IMSI	IMSI do assinante	"50557123456789"
Número VLR	Endereço GT VLR atual	"555123155"
Número MSC	Endereço GT MSC atual	"555123155"
Atualizado Em	Último timestamp de UpdateLocation	"2025-10-25 14:23:45 UTC"
Duração	Tempo desde o registro	"2h 15m 34s"

Resumo de Estatísticas

Quando assinantes estão presentes, um cartão de resumo exibe:

- **Total Ativo:** Total de assinantes registrados
- **VLRs Únicos:** Número de endereços VLR distintos
- **MSCs Únicos:** Número de endereços MSC distintos

Limpendo Assinantes

Botão Limpar Todos: Remove todos os assinantes ativos do rastreador.

Confirmação: Requer confirmação antes de limpar (não pode ser desfeito).

Caso de Uso: Limpar registros de assinantes obsoletos após manutenção ou teste de rede.

Atualização Automática

A página atualiza automaticamente a cada **2 segundos** para mostrar atualizações em tempo real dos assinantes.

Assinantes SMSc

Página: /smsc_subscribers

Modo: Apenas SMSc

Atualização Automática: A cada 2 segundos

Exibe rastreamento em tempo real de assinantes com base em mensagens alertServiceCenter recebidas dos HLRs, status de entrega de mensagens e rastreamento de falhas.

Recursos da Página

Colunas da Tabela de Assinantes

Coluna	Descrição	Exemplo
MSISDN	Número de telefone do assinante	"15551234567"
IMSI	IMSI do assinante	"001010123456789"
HLR GT	HLR GT que enviou alertServiceCenter	"15551111111"
Msgs Enviadas	Contagem de mensagens MT-FSM enviadas	5
Msgs Recebidas	Contagem de mensagens MO-FSM recebidas	2
Status	Ativo ou Falhou (código de cores)	● Ativo
Última Atualização	Timestamp da última atualização	"2025-10-30 14:23:45 UTC"
Duração	Tempo desde a última atualização	"15m 34s"

Indicadores de Status

- ● **Ativo** (Verde): Assinante é acessível, último alertServiceCenter recebido com sucesso
- ○ **Falhou** (Vermelho): Última tentativa de entrega falhou (erro SRI-for-SM ou assinante ausente)

Resumo de Estatísticas

Quando assinantes estão presentes, um cartão de resumo exibe:

- **Total Rastreado**: Total de assinantes rastreados
- **Ativos**: Número de assinantes com status ativo
- **Falhados**: Número de assinantes com status falhado
- **HLRs Únicos**: Número de HLRs distintos enviando alertas

Gerenciando Assinantes

Botão Remover: Remove assinante individual do rastreamento.

Botão Limpar Todos: Remove todos os assinantes rastreados.

Confirmação: Limpar Todos requer confirmação antes de limpar (não pode ser desfeito).

Caso de Uso:

- Remover entradas obsoletas após problemas de rede
- Limpar dados de teste após desenvolvimento
- Monitorar quais HLRs estão enviando alertas

Contadores de Mensagens

O rastreador incrementa automaticamente os contadores:

- **Mensagens Enviadas**: Incrementado quando SRI-for-SM é bem-sucedido e MT-FSM é enviado
- **Mensagens Recebidas**: Incrementado quando MO-FSM é recebido do assinante

Atualização Automática

A página atualiza automaticamente a cada **2 segundos** para mostrar atualizações em tempo real dos assinantes e status.

Operações Comuns

Pesquisa e Filtragem

Atualmente, a Interface Web não inclui funcionalidade de pesquisa/filtragem integrada. Para encontrar rotas específicas:

1. Use a função de busca do seu navegador (Ctrl+F / Cmd+F)
2. Pesquise por nomes de pares, códigos de ponto ou prefixos GT

Operações em Lote

Para realizar alterações em lote nas rotas:

1. **Opção 1:** Use a [API REST](#) para acesso programático
2. **Opção 2:** Edite `config/runtime.exs` e reinicie a aplicação
3. **Opção 3:** Use a Interface Web para alterações individuais de rotas

Exportar/Importar

Nota: A Interface Web atualmente não suporta exportação ou importação de tabelas de roteamento. As rotas são:

- Armazenadas em arquivos de banco de dados Mnesia
- Configuradas em `config/runtime.exs`

Para fazer backup das rotas:

1. **Mnesia:** Faça backup do diretório `Mnesia.{node_name}/`
2. **Config:** Controle de versão de `config/runtime.exs`

Comportamento de Atualização Automática

Diferentes páginas têm diferentes intervalos de atualização:

Página	Intervalo de Atualização	Razão
Gerenciamento de Roteamento	5 segundos	Mudanças de rota são infrequentes
Assinantes Ativos	2 segundos	Estado dos assinantes muda frequentemente
Status M3UA	Varia por página	Monitoramento do estado da conexão

Conexão WebSocket: Todas as páginas usam conexões WebSocket do Phoenix

LiveView para atualizações em tempo real.

Interrupção de Rede: Se a conexão WebSocket for perdida, a página tentará reconectar automaticamente.

Solução de Problemas

Página Não Carregando

1. **Verifique o Certificado HTTPS:** Certifique-se de que `priv/cert/omnitouch.crt` e `.pem` estão presentes
2. **Verifique a Porta 443:** Verifique se as regras do firewall permitem tráfego HTTPS
3. **Aplicação em Execução:** Confirme se a aplicação está em execução com `ix -S mix`
4. **Console do Navegador:** Verifique se há erros de certificado SSL (avisos de certificado autoassinado)

Rotas Não Persistindo

1. **Verifique o Armazenamento Mnesia:** Verifique `mnesia_storage_type:disc_copies` na configuração
2. **Diretório Mnesia:** Certifique-se de que o diretório Mnesia é gravável
3. **Verifique os Logs:** Procure por erros do Mnesia nos logs da aplicação

Atualização Automática Não Funcionando

1. **Conexão WebSocket:** Verifique o console do navegador para erros do WebSocket
 2. **Rede:** Verifique a conexão de rede estável
 3. **Recarregar a Página:** Tente atualizar a página (F5)
-

Documentação Relacionada

- [Guia STP](#) - Configuração detalhada de roteamento
 - [Guia HLR](#) - Gerenciamento de assinantes
 - [Guia API](#) - API REST para acesso programático
 - [Referência de Configuração](#) - Todos os parâmetros de configuração
-

Resumo

A Interface Web do OmniSS7 fornece gerenciamento intuitivo e em tempo real de tabelas de roteamento e rastreamento de assinantes:

- ❖ **Atualizações em Tempo Real** - Atualização automática mantém os dados atuais
- ❖ **Armazenamento Persistente** - Mnesia garante que as rotas sobrevivam a reinicializações
- ❖ **Interface Baseada em Funções** - Páginas se adaptam ao modo operacional (STP/HLR/SMSc)
- ❖ **Gerenciamento Interativo** - Adicione, edite, exclua rotas sem reinicialização
- ❖ **Monitoramento de Status** - Status de conexão e par em tempo real

Para operações avançadas ou automação, consulte o [Guia API](#).