# دليل واجهة برمجة التطبيقات REST

← العودة إلى الوثائق الرئيسية

يوفر هذا الدليل وثائق شاملة لواجهة برمجة التطبيقات **REST API** و **Swagger UI** الخاصة بـ OmniSS7.

## جدول المحتويات

1. نظرة عامة
2. تكوين خادم HTTP
3. Swagger UI
4. نقاط نهاية واجهة برمجة التطبيقات
5. المصادقة
6. تنسيقات الاستجابة
7. معالجة الأخطاء
8. المقاييس (بروميثيوس)
9. طلبات مثال

---

## نظرة عامة

يوفر OmniSS7 واجهة برمجة تطبيقات REST للوصول البرمجي إلى عمليات MAP (جزء تطبيق الهاتف المحمول). تتيح لك واجهة برمجة التطبيقات:

- إرسال طلبات MAP (SRI، SRI-for-SM، UpdateLocation، إلخ)
- استرجاع استجابات MAP
- مراقبة مقاييس النظام عبر بروميثيوس

## بنية واجهة برمجة التطبيقات

---

# تكوين خادم HTTP

## تفاصيل الخادم

| المعامل | القيمة | قابل للتكوين |
|---|---|---|
| **البروتوكول** | HTTP | لا |
| **عنوان IP** | 0.0.0.0 (جميع الواجهات) | عبر الكود فقط |
| **المنفذ** | 8080 | عبر الكود فقط |
| **النقل** | Plug.Cowboy | لا |

**رابط الوصول**: `http://[server-ip]:8080`

## تمكين/تعطيل خادم HTTP

سيبدأ HTTP تحكم فيما إذا كان خادم:

```
config :omniss7,
  start_http_server: true  # تعيين إلى false لتعطيل
```

**الإعداد الافتراضي**: `true` (مفعل)

**عند التعطيل**: لن يبدأ خادم HTTP، وستكون واجهة برمجة التطبيقات REST/Swagger UI غير متاحة.

---

# Swagger UI

تتضمن واجهة برمجة التطبيقات التفاعلية والاختبار **Swagger UI** لوثائق واجهة برمجة التطبيقات.

## الوصول إلى Swagger UI

**الرابط**: `http://[server-ip]:8080/swagger`

**الميزات**:

- وثائق واجهة برمجة التطبيقات التفاعلية
- وظيفة جربها لاختبار نقاط النهاية
- مخططات الطلب/الاستجابة
- أحمال مثال

## Swagger JSON

متاحة على OpenAPI مواصفة:

**الرابط**: `http://[server-ip]:8080/swagger.json`

**حالات الاستخدام**:

- الاستيراد إلى Postman أو عملاء واجهة برمجة التطبيقات الآخرين
- توليد مكتبات العملاء
- أتمتة وثائق واجهة برمجة التطبيقات

---

# نقاط نهاية واجهة برمجة التطبيقات

تتبع جميع نقاط نهاية عمليات MAP النمط: `POST /api/{operation}`

## ملخص نقاط النهاية

| نقطة النهاية | الطريقة | الغرض | المهلة |
|---|---|---|---|
| `/api/sri` | POST | إرسال معلومات التوجيه | 10s |
| `/api/sri-for-sm` | POST | إرسال معلومات التوجيه لـ SM | 10s |
| `/api/send-auth-info` | POST | إرسال معلومات المصادقة | 10s |
| `/api/MT-forwardSM` | POST | إعادة توجيه SMS إلى الهاتف المحمول | 10s |
| `/api/forwardSM` | POST | إعادة توجيه SMS | 10s |
| `/api/updateLocation` | POST | تحديث الموقع | 10s |
| `/api/prn` | POST | توفير رقم التجوال | 10s |
| `/metrics` | GET | مقاييس بروميثيوس | N/A |
| `/swagger` | GET | Swagger UI | N/A |
| `/swagger.json` | GET | مواصفة OpenAPI | N/A |

**ملاحظة**: جميع طلبات MAP تحتوي على **مهلة محددة مسبقًا مدتها 10 ثوانٍ**.

---

# SendRoutingInfo (SRI)

استرجاع معلومات التوجيه لإنشاء مكالمة لمشترك الهاتف المحمول.

**نقطة النهاية**: `POST /api/sri`

**جسم الطلب**:

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

**المعلمات**:

| الحقل | النوع | مطلوب | الوصف |
|---|---|---|---|
| `msisdn` | سلسلة | نعم | MSISDN الطرف المتصل |
| `gmsc` | سلسلة | نعم | عنوان بوابة MSC العالمي |

**الاستجابة** (200 OK):

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

**خطأ** (504 Gateway Timeout):

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

**مثال cURL**:

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

---

## SendRoutingInfoForSM (SRI-for-SM)

استرجاع معلومات التوجيه لتسليم SMS لمشترك الهاتف المحمول.

**نقطة النهاية**: `POST /api/sri-for-sm`

**جسم الطلب**:

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

**المعلمات**:

| الحقل | النوع | مطلوب | الوصف |
|---|---|---|---|
| `msisdn` | سلسلة | نعم | MSISDN الوجهة |
| `service_center` | سلسلة | نعم | عنوان مركز الخدمة العالمي |

**الاستجابة** (200 OK):

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

**مثال cURL**:

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

---

## SendAuthenticationInfo

طلب متجهات المصادقة لمشترك.

**نقطة النهاية**: `POST /api/send-auth-info`

**جسم الطلب**:

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

**المعلمات**:

| الحقل | النوع | مطلوب | الوصف |
|---|---|---|---|
| `imsi` | سلسلة | نعم | IMSI المشترك |
| `vectors` | عدد صحيح | نعم | عدد متجهات المصادقة لتوليدها |

**الاستجابة** (200 OK):

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

**مثال cURL**:

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

---

# MT-ForwardSM

تسليم SMS إلى مشترك.

**نقطة النهاية**: `POST /api/MT-forwardSM`

**جسم الطلب**:

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

**المعلمات**:

| الحقل | النوع | مطلوب | الوصف |
|---|---|---|---|
| `imsi` | سلسلة | نعم | IMSI المشترك الوجهة |
| `destination_service_centre` | سلسلة | نعم | GT لمركز الخدمة الوجهة |
| `originating_service_center` | سلسلة | نعم | GT لمركز الخدمة المنشئ |
| `smsPDU` | سلسلة | نعم | SMS TPDU بتنسيق سداسي عشري |

**ملاحظة**: يجب أن يكون `smsPDU` (كبيرة أو صغيرة) سلسلة مشفرة بتنسيق سداسي عشري.

**الاستجابة** (200 OK):

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

**مثال 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

إعادة توجيه رسالة SMS (MO-SMS من المشترك).

**نقطة النهاية**: `POST /api/forwardSM`

**جسم الطلب**: نفس MT-ForwardSM

**مثال 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

إخطار HLR بتغيير موقع المشترك (تسجيل VLR).

**نقطة النهاية**: `POST /api/updateLocation`

**جسم الطلب**:

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

**المعلمات**:

| الحقل | النوع | مطلوب | الوصف |
|---|---|---|---|
| `imsi` | سلسلة | نعم | IMSI المشترك |
| `vlr` | سلسلة | نعم | عنوان VLR العالمي |

**الاستجابة** (200 OK):

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

**ملاحظة**: في وضع HLR، يؤدي هذا إلى بدء تسلسل InsertSubscriberData (ISD) مع مهلة مدتها 10 ثوانٍ لكل ISD.

**مثال cURL**:

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

---

## ProvideRoamingNumber (PRN)

طلب MSRN (رقم التجوال لمحطة الهاتف المحمول) لتوجيه المكالمات إلى المشترك المتجول.

**نقطة النهاية**: `POST /api/prn`

**جسم الطلب**:

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

**المعلمات**:

| الحقل | النوع | مطلوب | الوصف |
|---|---|---|---|
| `msisdn` | سلسلة | نعم | MSISDN المشترك |
| `gmsc` | سلسلة | نعم | GT لبوابة MSC |
| `msc_number` | سلسلة | نعم | رقم MSC للمشترك |
| `imsi` | سلسلة | نعم | IMSI المشترك |

**الاستجابة** (200 OK):

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

**مثال cURL**:

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

---

# المصادقة

**المصادقة الحالية**: لا تتطلب واجهة برمجة التطبيقات.

**اعتبارات الأمان**:

- واجهة برمجة التطبيقات مخصصة للاستخدام في الشبكات الداخلية/الموثوقة
- ضع في اعتبارك استخدام قواعد جدار الحماية لتقييد الوصول
- بالنسبة لنشر الإنتاج، ضع في اعتبارك تنفيذ وسيط المصادقة

---

# تنسيقات الاستجابة

تستخدم جميع الاستجابات تنسيق **JSON**.

## استجابة النجاح

**حالة HTTP**: 200 OK

**الهيكل**:

```
{
  "result": {
    // بيانات الاستجابة الخاصة بالعملية
  }
}
```

## استجابة الخطأ

**حالة HTTP**:

- 400 Bad Request - جسم الطلب غير صالح
- 504 Gateway Timeout - انتهاء مهلة طلب MAP (10 ثوانٍ)
- 404 Not Found - نقطة نهاية غير صالحة

**الهيكل**:

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

أو

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

---

# معالجة الأخطاء

## الأخطاء الشائعة

| الخطأ | رمز HTTP | الوصف | الحل |
|---|---|---|---|
| JSON غير صالح | 400 | جسم الطلب ليس JSON صالحًا | JSON تحقق من بناء جملة |
| الحقول المفقودة | 400 | الحقول المطلوبة مفقودة | قم بتضمين جميع المعلمات المطلوبة |
| انتهاء المهلة | 504 | المهلة MAP تجاوز طلب المحددة 10 ثوا�ٍ | M3UA، تحقق من اتصال وتوفر HLR/VLR |
| غير موجود | 404 | نقطة نهاية غير صالحة | لنقطة URL تحقق من عنوان النهاية |

## سلوك انتهاء المهلة

تحتوي جميع طلبات MAP على **مهلة محددة مسبقًا مدتها 10 ثوانٍ**:

1. يتم إرسال الطلب إلى MapClient GenServer
2. ينتظر الاستجابة لمدة تصل إلى 10 ثوانٍ
3. إذا لم يتم تلقي استجابة → يتم إرجاع 504 Gateway Timeout
4. إذا تم تلقي استجابة → يتم إرجاع 200 OK مع النتيجة

**استكشاف أخطاء انتهاء المهلة**:

- تحقق من حالة اتصال M3UA (صفحة واجهة الويب → M3UA)
- تحقق من أن العنصر الشبكي (HLR/VLR/MSC) يمكن الوصول إليه
- تحقق من تكوين التوجيه
- راجع سجلات أحداث SS7 للبحث عن الأخطاء

---

# المقاييس (بروميثيوس)

تقوم واجهة برمجة التطبيقات بتعريض مقاييس بروميثيوس للمراقبة.

## نقطة نهاية المقاييس

**الرابط**: `http://[server-ip]:8080/metrics`

**التنسيق**: تنسيق نص بروميثيوس

**مثال على المخرجات**:

```
# HELP map_requests_total Total MAP requests
# 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 CAP requests
# TYPE cap_requests_total counter
cap_requests_total{operation="initialDP"} 87
cap_requests_total{operation="requestReportBCSMEvent"} 91

# HELP map_request_duration_milliseconds Duration of MAP
request/responses in 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 Number of pending MAP TID waiters
# TYPE map_pending_requests gauge
map_pending_requests 3
```

## المقاييس المتاحة

| المقياس | النوع | التسميات | الوصف |
|---|---|---|---|
| `map_requests_total` | عداد | `operation` | إجمالي عدد طلبات MAP حسب نوع العملية |
| `cap_requests_total` | عداد | `operation` | إجمالي عدد طلبات CAP حسب نوع العملية |
| `map_request_duration_milliseconds` | هيستوجرام | `operation` | مدة ��لطلب بالمللي ثانية |
| `map_pending_requests` | مقياس | - | عدد معاملات MAP المعلقة |

## تكوين بروميثيوس

أضف إلى `prometheus.yml` الخاص بك:

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

# طلبات مثال

## مثال بايثون

```python
import requests
import json

# طلب 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"نجاح: {result}")
elif response.status_code == 504:
    print("انتهاء المهلة - لا استجابة من الشبكة")
else:
    print(f"خطأ: {response.status_code} - {response.text}")
```

## مثال جافا سكريبت

```
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('نجاح:', response.data);
  } catch (error) {
    if (error.code === 'ECONNABORTED') {
      console.error('انتهاء المهلة - لا استجابة من الشبكة');
    } else {
      console.error('خطأ:', error.response?.data ||
error.message);
    }
  }
}

sendSRI();
```

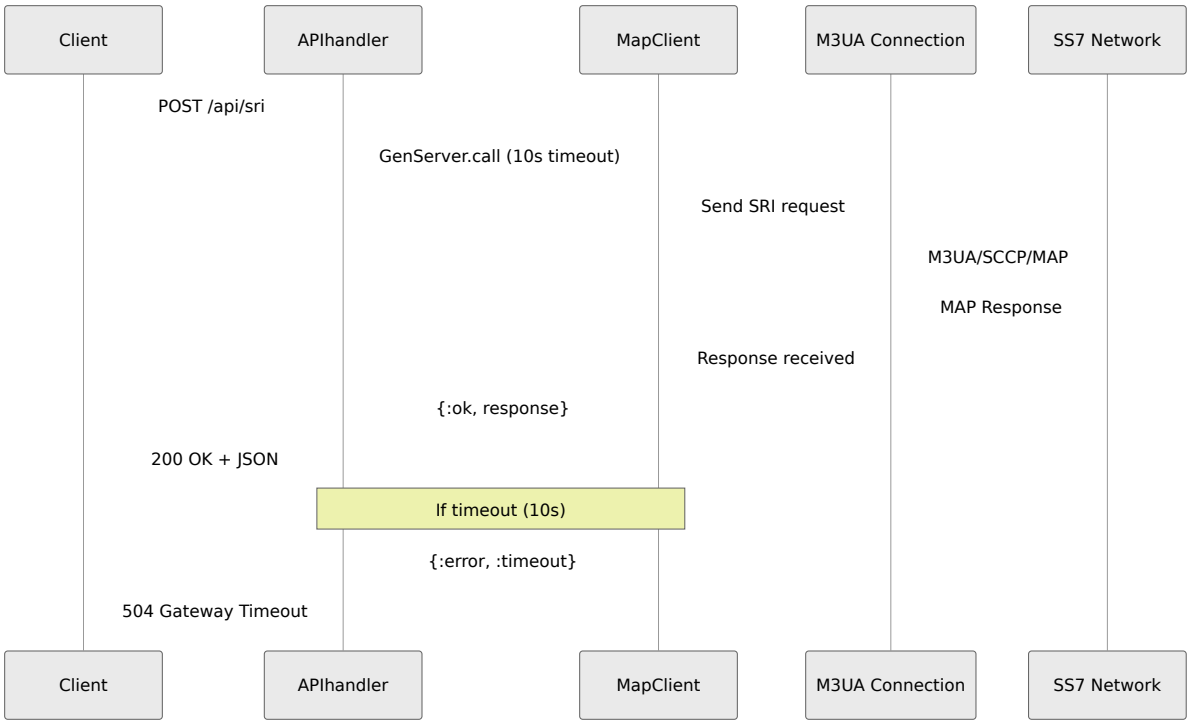
## مثال Bash/cURL

```
#!/bin/bash

# طلب 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 "نجاح: $body"
elif [ "$http_code" -eq 504 ]; then
  echo "انتهاء المهلة - لا استجابة من الشبكة"
else
  echo "خطأ $http_code: $body"
fi
```

---

# مخططات التدفق

## تدفق طلب واجهة برمجة التطبيقات




---

# الملخص

توفر واجهة برمجة التطبيقات REST API الخاصة بـ OmniSS7:

- **عمليات MAP** - دعم كامل لـ SRI، SRI-for-SM، UpdateLocation، تسليم SMS، المصادقة
- **Swagger UI** - وثائق واجهة برمجة التطبيقات التفاعلية والاختبار
- **مقاييس بروميثيوس** - المراقبة والرؤية
- **مهلات محددة مسبقًا** MAP - مهلة مدتها 10 ثوانٍ لجميع طلبات
- **خادم HTTP** - 8080 يعمل على المنفذ (قابل للتكوين عبر `start_http_server`)

للوصول إلى واجهة الويب، راجع دليل واجهة الويب.

للحصول على تفاصيل التكوين، راجع مرجع التكوين.

# مرجع تقني (ملحق)

← العودة إلى الوثائق الرئيسية

مرجع تقني لبروتوكولات SS7 وتنفيذ OmniSS7.

## مكدس بروتوكولات SS7

---

## رموز عمليات MAP

| العملية | رمز العملية | الغرض |
|---|---|---|
| updateLocation | 2 | تسجيل موقع المشترك |
| cancelLocation | 3 | إلغاء التسجيل من VLR |
| provideRoamingNumber | 4 | طلب MSRN |
| sendRoutingInfo | 22 | استعلام عن توجيه المكالمات |
| mt-forwardSM | 44 | تسليم SMS إلى المشترك |
| sendRoutingInfoForSM | 45 | استعلام عن توجيه SMS |
| mo-forwardSM | 46 | إعادة توجيه SMS من المشترك |
| sendAuthenticationInfo | 56 | طلب متجهات المصادقة |

---

# أنواع رسائل TCAP

- **BEGIN** - بدء المعاملة
- **CONTINUE** - منتصف المعاملة
- **END** - الاستجابة النهائية
- **ABORT** - إلغاء المعاملة

---

# عنونة SCCP

## تنسيقات العنوان العالمية

- **E.164** - رقم الهاتف الدولي (مثل: 447712345678)
- **E.212** - تنسيق IMSI (مثل: 234509876543210)
- **E.214** - تنسيق رمز النقطة

## أرقام النظام الفرعي (SSN)

- **SSN 6**: HLR
- **SSN 7**: VLR
- **SSN 8**: MSC/SMSC
- **SSN 9**: GMLC
- **SSN 10**: SGSN

---

# SMS TPDU

## أنواع الرسائل

- **SMS-DELIVER** (MT) - من الشبكة إلى الهاتف المحمول
- **SMS-SUBMIT** (MO) - من الهاتف المحمول إلى الشبكة
- **SMS-STATUS-REPORT** - حالة التسليم
- **SMS-COMMAND** - أمر عن بعد

### ترميزات الأحرف

- **GSM7** - أبجدية GSM 7 بت (160 SMS لكل حرف)
- **UCS2** - يونيكود 16 بت (70 SMS لكل حرف)
- **8-bit** - بيانات ثنائية (140 SMS لكل بايت)

---

## حالات M3UA

- **DOWN** - لا يوجد اتصال SCTP
- **CONNECTING** - الاتصال بـ SCTP
- **ASPUP_SENT** - في انتظار تأكيد ASPUP
- **INACTIVE** - ASP نشط ولكن غير مفعل
- **ASPAC_SENT** - في انتظار تأكيد ASPAC
- **ACTIVE** - جاهز للحركة

---

## رموز النقاط الشائعة في SS7

عادةً ما تكون رموز النقاط قيم 14 بت (ITU) أو 24 بت (ANSI).

**تنسيق المثال (ITU):**

- الشبكة: 3 بت
- العنقود: 8 بت
- العضو: 3 بت

---

## رموز خطأ SCCP

- **0** - لا يوجد ترجمة للعنوان
- **1** - لا يوجد ترجمة لعنوان محدد
- **2** - ازدحام النظام الفرعي
- **3** - فشل النظام الفرعي

- **4** - مستخدم غير مجهز
- **5** - فشل MTP
- **6** - ازدحام الشبكة
- **7** - غير مؤهل
- **8** - خطأ في نقل الرسالة

---

# رموز خطأ MAP

| الرمز | الخطأ | الوصف |
|---|---|---|
| 1 | unknownSubscriber | المشترك غير موجود في HLR |
| 27 | absentSubscriber | المشترك غير قابل للوصول |
| 34 | systemFailure | فشل الشبكة |
| 35 | dataMissing | البيانات المطلوبة غير متوفرة |
| 36 | unexpectedDataValue | قيمة معلمة غير صالحة |

---

# الوثائق ذات الصلة

- ← العودة إلى الوثائق الرئيسية
- دليل STP
- دليل عميل MAP
- دليل مركز SMS
- دلي�� HLR
- الميزات الشائعة

---

**OmniSS7** بواسطة خدمات الشبكة Omnitouch

# CAMEL - مُنشئ طلب ملخص التنفيذ

## نظرة عامة

تم إنشاء مكون LiveView جديد لبناء وإرسال طلبات CAMEL/CAP لأغراض الاختبار. يوفر هذا واجهة مستخدم تفاعلية لإنشاء InitialDP وعمليات CAMEL الأخرى.

## المكونات الجديدة

### 1. CAMEL LiveView مُنشئ طلب

**الميزات:**

- واجهة مستخدم تفاعلية تعتمد على النماذج لبناء طلبات CAMEL
- دعم لعدة أنواع من الطلبات:

    - **InitialDP** - (إشعار إعداد المكالمة) نقطة الكشف الأولية
    - **Connect** - ربط المكالمة بالوجهة
    - **ReleaseCall** - إنهاء/تحرير المكالمة
    - **RequestReportBCSMEvent** - طلب إشعارات الأحداث
    - **Continue** - مواصلة معالجة المكالمة
    - **ApplyCharging** - تطبيق حدود الشحن/المدة على المكالمات

**القدرات الرئيسية:**

- قائمة منسدلة لاختيار نوع الطلب
- حقول نموذج ديناميكية بناءً على نوع الطلب المحدد
- خيارات SCCP/M3UA (قسم قابل للطي) متقدمة
    - عناوين الأطراف المتصلة/المتصل
    - تكوين SSN (رقم النظام الفرعي)
    - إعدادات OPC/DPC (رمز النقطة)
- تاريخ الطلبات في الوقت الحقيقي (آخر 20 طلبًا)
- تتبع الجلسة عبر OTID
- ملاحظات النجاح/الخطأ
- تتبع حجم الطلب

**المسار:** `/camel_request`

# 2. سجل الأحداث المحسن مع دعم CAMEL

**الوظائف الجديدة:**

- `paklog_camel/2` - تسجيل رسائل CAMEL/CAP المخصصة
- `lookup_cap_opcode_name/1` - البحث عن رمز عملية CAP
- `find_cap_opcode/1` - استخراج رمز CAP من JSON
- `extract_cap_tids/1` - استخراج OTID/DTID من رسائل CAP
- `format_cap_to_json/1` - تحويل PDUs CAP إلى تنسيق JSON

**رموز عمليات CAP المدعومة:**

```
0  => "initialDP"
5  => "connect"
6  => "releaseCall"
7  => "requestReportBCSMEvent"
8  => "eventReportBCSM"
10 => "continue"
13 => "furnishChargingInformation"
35 => "applyCharging"
... (إجمالي العمليات 47)
```

**الميزات:**

- تسجيل JSON لجميع طلبات/استجابات CAMEL
- اكتشاف تلقائي لإجراءات TCAP (بدء/استمرار/إنهاء/إلغاء)
- استخراج عناوين SCCP
- معالجة الأخطاء للرسائل غير الصحيحة
- معالجة المهام في الخلفية (غير مت blocking)
- أحداث مسبوقة بـ "CAP:" لتسهيل التصفية

## 3. تحديث CapClient

**التغييرات:**

- إضافة استدعاءات `paklog_camel/2` للرسائل الواردة والصادرة
- تسجيل مزدوج: كل من MAP (`paklog`) و CAP (`paklog_camel`) للتوافق
- تسجيل الرسائل الصادرة في `sccp_m3ua_maker/2`
- تسجيل الرسائل الواردة في `handle_payload/1`

# التكوين

تمت إضافة صفحات LiveView الجديدة إلى تكوين وقت التشغيل:

```
# الملف: config/runtime.exs

config :control_panel,
  use_additional_pages: [
    {SS7.Web.EventsLive, "/events", "SS7 Events"},
    {SS7.Web.TestClientLive, "/client", "SS7 Client"},
    {SS7.Web.M3UAStatusLive, "/m3ua", "M3UA"},
    {SS7.Web.HlrLinksLive, "/hlr_links", "HLR Links"},
    {SS7.Web.CAMELSessionsLive, "/camel_sessions", "CAMEL
Sessions"},
    {SS7.Web.CAMELRequestLive, "/camel_request", "CAMEL Request
Builder"}
  ],
  page_order: ["/events", "/client", "/m3ua", "/hlr_links",
               "/camel_sessions", "/camel_request",
               "/application", "/configuration"]
```

# الاستخدام

## الوصول إلى مُنشئ الطلب

1. انتقل إلى: `https://your-server:8087/camel_request`
2. اختر نوع الطلب من القائمة المنسدلة
3. املأ المعلمات المطلوبة
4. اختياريًا قم بتوسيع "خيارات SCCP/M3UA المتقدمة" للتعديل الدقيق
5. انقر على "إرسال [RequestType] طلب"

## تدفق الطلب

### InitialDP (مكالمة جديدة)

1. تعيين مفتاح الخدمة (على سبيل المثال، 100)
2. تعيين رقم المتصل (A-Party)
3. تعيين رقم المتصل به (B-Party)
4. إرسال الطلب → يولد OTID جديد
5. يتم تخزين OTID في الجلسة للطلبات اللاحقة

**الطلبات اللاحقة (Connect، ReleaseCall، إلخ.)**

1. يجب أن يكون لديك OTID نشط من InitialDP
2. يستخدم الطلب تلقائيًا OTID المخزن
3. يظهر تحذير إذا لم يكن هناك OTID نشط

# معلمات الطلب

**InitialDP:**

- مفتاح الخدمة (عدد صحيح)
- رقم المتصل (بتنسيق ISDN)
- رقم المتصل به (بتنسيق ISDN)

**Connect:**

- رقم الوجهة (أين يتم توجيه المكالمة)

**ReleaseCall:**

- رمز السبب (16 = عادي، 17 = مشغول، 31 = غير محدد)

**RequestReportBCSMEvent:**

- أحداث BCSM (مفصولة بفواصل: oAnswer، oDisconnect، إلخ.)

**Continue:**

- لا توجد معلمات (تستخدم OTID النشط)

**ApplyCharging:**

- المدة (بالثواني، 1-864000) - الحد الأقصى لمدة المكالمة قبل الإجراء
- ما إذا كان يجب تحرير المكالمة عند انتهاء المدة - تحرير عند انتهاء المهلة (boolean)

# الخيارات المتقدمة

**توجيه SCCP:**

- للطرف المتصل (العنوان العالمي) GT

- GT للطرف المتصل
- SSN المتصل به (146 افتراضي = gsmSSF)
- SSN (146 افتراضي) المتصل

**رموز نقاط M3UA:**

- OPC (5013 افتراضي، رمز النقطة الأصلية)
- DPC (5011 افتراضي، رمز النقطة الوجهة)

# تسجيل JSON

جميع رسائل CAMEL مسجلة الآن بتنسيق JSON مع في سجل الأحداث:

- **الاتجاه:** وارد/صادر
- **إجراء TCAP:** بدء/استمرار/إنهاء/إلغاء
- **عملية CAP:** على سبيل المثال، "CAP:initialDP"، "CAP:connect"
- **توجيه SCCP:** معلومات الطرف المتصل/المتصل
- **TIDs:** OTID/DTID للتوافق
- **الرسالة الكاملة:** PDU CAP مشفرة بتنسيق JSON

## مثال على إدخال السجل

```
{
  "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 ... }"
}
```

# تاريخ الطلب

تظهر واجهة المستخدم آخر 20 طلبًا مع:

- الطابع الزمني
- نوع الطلب (مع شارة ملونة)
- OTID (أول 8 أحرف سداسية)
- الحالة (تم الإرسال/خطأ)
- حجم الرسالة بالبايت

# تتبع الجلسة

**لوحة معلومات الجلسة الحالية:**

- تعرض OTID النشط
- تظهر حجم البايت للطلب الأخير
- مرئية فقط عندما تكون الجلسة نشطة

# سير عمل الاختبار

1. **بدء ��كالمة جديدة:**
    - إرسال InitialDP → الحصول على OTID
    - يقوم النظام بإنشاء جلسة
2. **التحكم في المكالمة:**
    - إرسال RequestReportBCSMEvent → طلب الإشعارات
    - إرسال ApplyCharging → تعيين حد مدة المكالمة (على سبيل المثال، 290 ثانية)
    - إرسال Connect → توجيه إلى الوجهة
    - إرسال ReleaseCall → إنهاء أو
3. **عرض النتائج:**
    - تحقق من تاريخ الطلب

- مراقبة صفحة جلسات CAMEL
- مراجعة سجلات الأحداث مع بادئة "CAP:"

# ApplyCharging - التحكم في مدة المكالمة

## نظرة عامة

تتيح لك عملية ApplyCharging تعيين الحد الأقصى لمدة المكالمة وإمكانية تحرير المكالمة عند انتهاء تلك المدة. يستخدم هذا عادةً في سيناريوهات الشحن المدفوع مسبقًا أو فرض حدود زمنية على المكالمات.

## حالات الاستخدام

- **الشحن المدفوع مسبقًا**: تحديد مدة المكالمة بناءً على رصيد المشترك
- **الفوترة القائمة على الوقت**: فرض فترات شحن دورية
- **إدارة ا��موارد**: منع المكالمات من الاستمرار إلى أجل غير مسمى
- **تكامل OCS**: التنسيق مع أنظمة الشحن عبر الإنترنت للتحكم في ��لائتمان في الوقت الحقيقي

## المعلمات

### المدة (maxCallPeriodDuration)

- **النوع**: عدد صحيح (1-864000 ثانية)
- **الوصف**: الحد الأقصى لعدد الثواني التي يمكن أن تستمر فيها المكالمة قبل انتهاء المؤقت
- **أمثلة**:
  - `60` = 1 دقيقة
  - `290` = 4 دقائق و50 ثانية (قيمة اختبار شائعة)
  - `3600` = 1 ساعة
  - `86400` = 24 ساعة

### الإفراج عند انتهاء المهلة (releaseIfDurationExceeded)

- **النوع**: Boolean (true/false)
- **الافتراضي**: true
- **الوصف**: ماذا يحدث عند انتهاء المدة:
  - `true`: تحرير/فصل المكالمة تلقائيًا
  - `false`: إرسال إشعار ولكن إبقاء المكالمة نشطة (يسمح لـ gsmSCF باتخاذ إجراء)

## هيكل الرسالة

يتم ترميز رسالة ApplyCharging كـ TCAP Continue مع:

- **TCAP**: رسالة استمرار (تستخدم المعاملة الحالية)
- **Opcode**: 35 (applyCharging)
- **المعلمات**: ApplyChargingArg تحتوي على:
  - `aChBillingChargingCharacteristics`: معلومات الشحن القائمة على الوقت
    - `timeDurationCharging`: الحد الأقصى للم��ة وعلم الإفراج
  - `partyToCharge`: أي طرف يتم شحنه (افتراضي: sendingSideID)

## مثال على الاستخدام

**السيناريو**: مكالمة مدفوعة مسبقًا بحد 5 دقائق

1. إرسال **InitialDP** لبدء مراقبة المكالمة

```
مفتاح الخدمة: 100
المتصل: 447700900123
المتصل به: 447700900456
→ OTID: A1B2C3D4
```

2. إرسال **ApplyCharging** لتعيين حد 5 دقائق

```
المدة: 300 (ثانية)
الإفراج عند انتهاء المهلة: true
→ يستخدم OTID: A1B2C3D4
```

3. إرسال **Connect** لإكمال المكالمة

```
الـوجهة: 447700900456
→ يستخدم OTID: A1B2C3D4
```

4. بعد 5 دقائق (300 ثانية):
   - يتم تحرير المكالمة تلقائيًا بواسطة الشبكة
   - يتلقى gsmSCF إشعار فصل

# أفضل الممارسات

1. **دائمًا أرسل ApplyCharging قبل Connect**
   - يضمن أن الشحن نشط عند الاتصال بالمكالمة
   - يمنع أجزاء المكالمات غير المشحونة
2. **استخدم مع RequestReportBCSMEvent**
   - طلب أحداث `oAnswer` و `oDisconnect`
   - يسمح بتتبع مدة المكالمة ال��علية
   - يمكّن من إعادة تطبيق الشحن إذا لزم الأمر
3. **تعيين مدد معقولة**
   - قصيرة جدًا: عمليات شحن متكررة، تجربة مستخدم سيئة
   - طويلة جدًا: خطر فقدان الإيرادات على المكالمات المدفوعة مسبقًا
   - النموذجية: 60-300 ثانية للمدفوعة مسبقًا، أطول للمدفوعة لاحقًا
4. **التعامل مع انتهاء المهلة بشكل سلس**
   - كن مستعدًا للتعامل مع إشعارات انتهاء المؤقت، إذا كان `release=false`
   - تنفيذ منطق لتمديد المدة أو تحرير المكالمة

# معالجة الأخطاء

المشاكل الشائعة:

- أولاً InitialDP **نشط**: يجب إرسال **لا يوجد OTID**
- **مدة غير صالحة**: يجب أن تكون 1-864000 ثانية
- SSF ApplyCharging **دعم الشبكة**: قد لا تدعم بعض تنفيذات
- **دقة المؤقت**: عادةً ما تكون دقة مؤقت الشبكة 1 ثانية، ولكن قد تختلف

### المراقبة

تتبع عمليات ApplyCharging عبر:

- المرسلة ApplyCharging **تاريخ الطلب**: يظهر طلبات
- **سجل الأحداث**: البحث عن "CAP:applyCharging"
- **جلسات CAMEL**: مراقبة الجلسات النشطة مع تطبيق الشحن
- **تتبع TCAP**: تصحيح مشاكل الترميز/فك الترميز

## تفاصيل التنفيذ

### إدارة الحالة

- يقوم LiveView بتعيين حالة نموذج المسار
- يتم تخزين OTID في تعيين المقبس
- تاريخ الطلب محدود إلى 20 إدخال
- تم تعطيل التحديث التلقائي (إرسال يدوي فقط)

### إنشاء الطلب

- يستخدم وحدة `CapRequestGenerator` الحالية
- يبني هياكل TCAP/CAP الصحيحة
- يقوم بالترميز باستخدام ترميز `:TCAPMessages`
- يلف في SCCP عبر `CapClient.sccp_m3ua_maker/2`

### آلية الإرسال

- يرسل عبر M3UA إلى `:camelgw_client_asp`
- يستخدم سياق التوجيه 1

- تغليف SCCP/M3UA تلقائي

### معالجة الأخطاء

- تحقق من صحة النموذج مع ملاحظات المستخدم
- التعامل السلس مع OTID المفقود
- تظهر أخطاء التحليل في واجهة المستخدم
- يتم تسجيل فشل الترميز

## تحسينات مستقبلية

الإضافات المحتملة:

1. قوالب/إعدادات مسبقة للطلبات
2. ارتباط ال��ستجابة وعرضها
3. تصور تدفق المكالمات
4. تفاصيل الجلسة
5. تصدير تاريخ الطلب
6. اختبار التحميل (طلبات جماعية)
7. تصدير PCAP للرسائل المولدة
8. التحقق من معلمات CAP

## ملاحظات التكامل

- متوافق مع تسجيل MAP الحالي (`paklog`)
- يشارك قاعدة بيانات سجل الأحداث مع أحداث MAP
- يستخدم نفس بنية SCCP/M3UA
- يعمل مع CAMELSessionsLive للمراقبة
- يتكامل مع توجيه M3UA الحالي

## الملفات المعدلة

- `config/runtime.exs` - مُحدث

# التبعيات

- CapRequestGenerator الحالي
- CapClient لإرسال M3UA
- M3UA.Server لنقل الحزم
- EventLog لتسجيل الرسائل
- إطار عمل Phoenix LiveView
- لوحة التحكم لبنية واجهة المستخدم

# دليل تكوين بوابة CAMEL

## نظرة عامة

توفر خدمات (IN) إلى منصة شبكة ذكية OmniSS7 يحول **CAMEL (CAMELGW)** وضع **بوابة** التحكم في المكالمات والشحن في الوقت الفعلي باستخدام بروتوكول جزء تطبيق CAMEL (CAP).

## ما هو CAMEL؟

هو مجموعة من المعايير (تطبيقات مخصصة للمنطق المعزز لشبكة الهاتف المحمول) **CAMEL** المصممة للعمل على شبكة GSM الأساسية أو شبكة UMTS. يسمح للمشغلين بتقديم خدمات تتطلب التحكم في المكالمات في الوقت الفعلي، مثل:

- **المكالمات المدفوعة مسبقًا** - التحقق من الرصيد والشحن في الوقت الفعلي
- **خدمات الأسعار المميزة** - الفوترة الخاصة للخدمات ذات القيمة المضافة
- **تحكم في توجيه المكالمات** - توجيه ��لوجهة الديناميكي بناءً على الوقت/الموقع
- **الشبكات الخاصة الافتراضية** - خطط ترقيم الشركات

- **تصفية المكالمات** - السماح/حظر المكالمات بناءً على المعايير

## إصدارات بروتوكول CAP

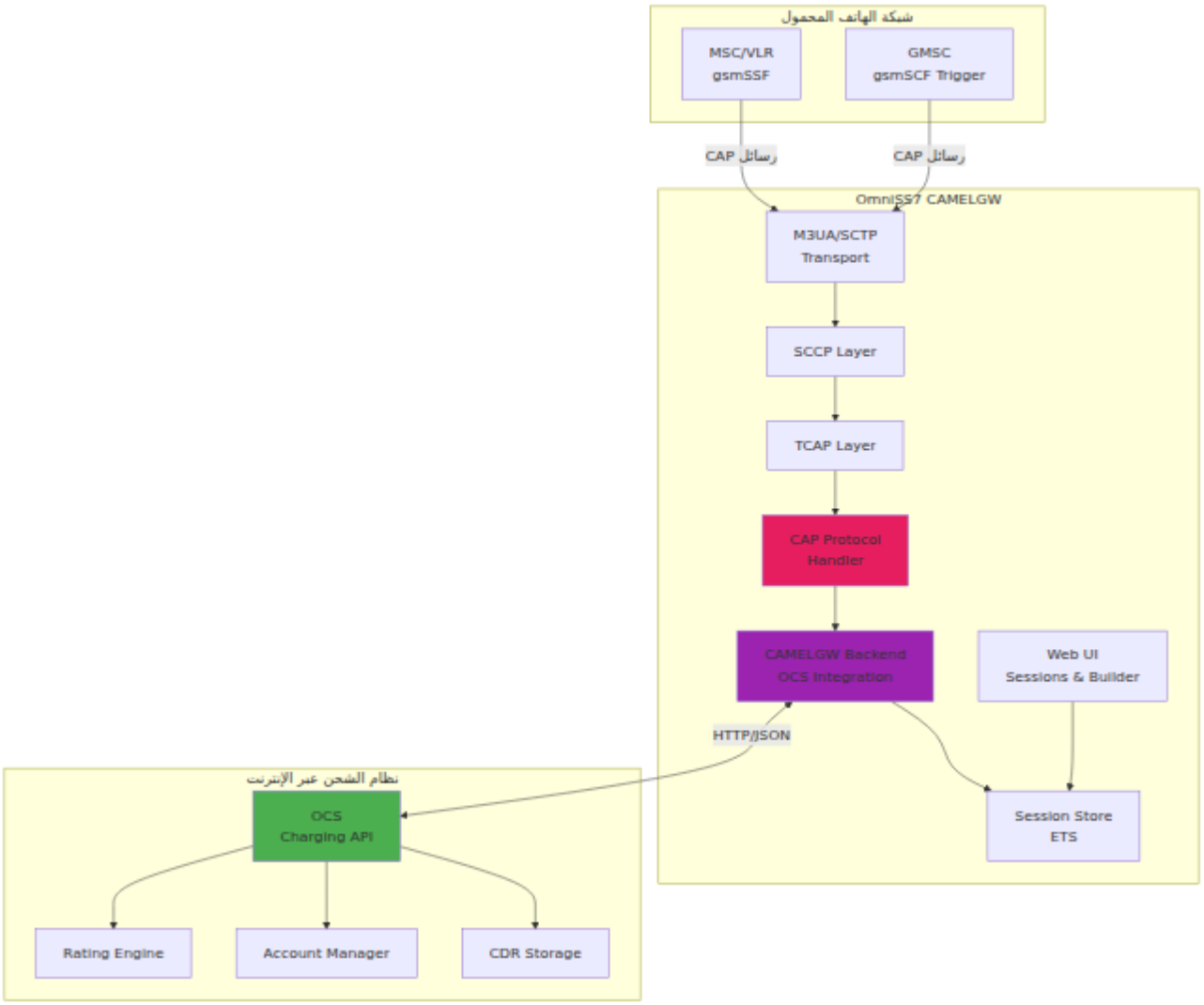
يدعم OmniSS7 CAMELGW عدة إصدارات من CAP:

| الإصدار | المرحلة | الميزات |
|---|---|---|
| **CAP v1** | المرحلة 1 من CAMEL | التحكم الأساسي في المكالمات، عمليات محدودة |
| **CAP v2** | المرحلة 2 من CAMEL | عمليات محسنة، دعم SMS |
| **CAP v3** | المرحلة 3 من CAMEL | عمليات إضافية، دعم GPRS |
| **CAP v4** | المرحلة 4 من CAMEL | ميزات متقدمة، دعم الوسائط المتعددة |

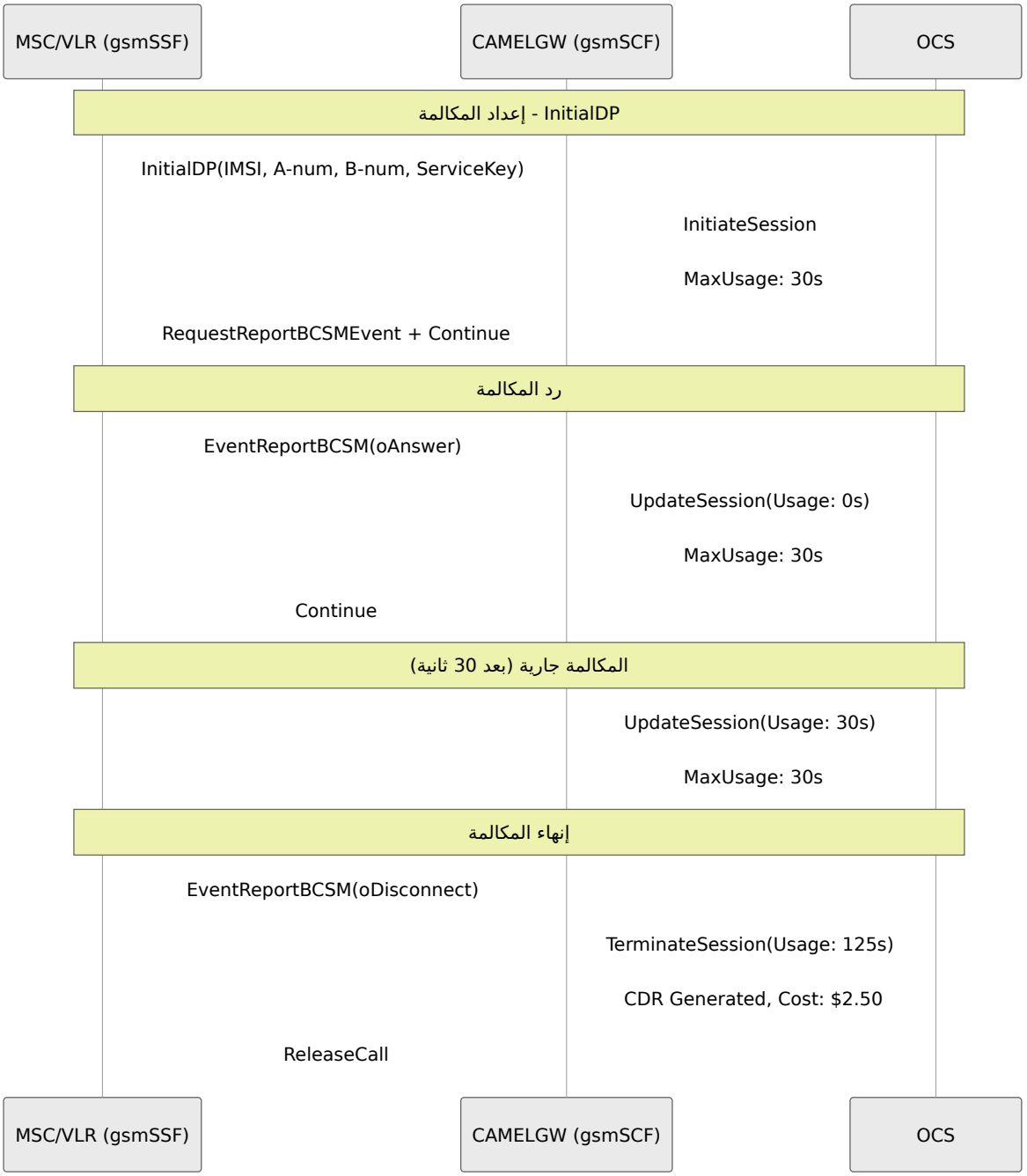
**الإعداد الافتراضي:** CAP v2 (الأكثر انتشارًا)

---

# الهيكلية


شبكة الهاتف المحمول
MSC/VLR
gsmSSF
GMSC
gsmSCF Trigger
رسائل CAP
رسائل CAP
OmniSS7 CAMELGW
M3UA/SCTP
Transport
SCCP Layer
TCAP Layer
CAP Protocol
Handler
CAMELGW Backend
OCS Integration
Web UI
Sessions & Builder
HTTP/JSON
Session Store
ETS
نظام الشحن عبر الإنترنت
OCS
Charging API
Rating Engine
Account Manager
CDR Storage

## مثال على تدفق المكالمات




---

# التكوين

## المتطلبات المسبقة

- تثبيت وتشغيل OmniSS7

- اتصال M3UA بـ MSC/GMSC (gsmSSF)
- اختياري، للشحن في الوقت) API مع نقطة نهاية (OCS) نظام الشحن عبر الإنترنت الفعلي)

## تفعيل وضع بوابة CAMEL

قم بتحرير `config/runtime.exs` وقم بتكوين قسم بوابة CAMEL:

```
config :omniss7,
  # علامات الوضع - تفعيل ميزات CAP/CAMEL
  cap_client_enabled: true,
  camelgw_mode_enabled: true,

  # تعطيل الأوضاع الأخرى
  map_client_enabled: false,
  hlr_mode_enabled: false,
  smsc_mode_enabled: false,

  # تكوين إصدار CAP/CAMEL
  # يجب استخدامه للطلبات والمحادثات الصادرة CAP يحدد أي إصدار
  # الخيارات: :v1, :v2, :v3, :v4
  cap_version: :v2,

  # تكامل OCS (للشحن في الوقت الفعلي)
  ocs_enabled: true,
  ocs_url: "http://your-ocs-server/api/charging",
  ocs_timeout: 5000,  # مللي ثانية
  ocs_auth_token: "your-api-token"  # إذا كان اختياري، يتطلب OCS
مصادقة

  # تكوين اتصال M3UA لـ CAMEL
  # الاتصال كـ ASP (عملية خادم التطبيق) لعمليات CAP
  cap_client_m3ua: %{
    mode: "ASP",
    callback: {CapClient, :handle_payload, []},
    process_name: :camelgw_client_asp,

    # نقطة النهاية المحلية (نظام CAMELGW)
    local_ip: {10, 179, 4, 13},
    local_port: 2905,

    # نقطة النهاية البعيدة (MSC/GMSC - gsmSSF)
    remote_ip: {10, 179, 4, 10},
    remote_port: 2905,

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

## تكوين صفحات واجهة الويب

تتضمن واجهة الويب صفحات متخصصة لعمليات CAMEL:

```
config :control_panel,
  use_additional_pages: [
    {SS7.Web.EventsLive, "/events", "SS7 Events"},
    {SS7.Web.TestClientLive, "/client", "SS7 Client"},
    {SS7.Web.M3UAStatusLive, "/m3ua", "M3UA"},
    {SS7.Web.CAMELSessionsLive, "/camel_sessions", "CAP
Sessions"},
    {SS7.Web.CAMELRequestLive, "/camel_request", "CAP Requests"}
  ],
  page_order: ["/events", "/client", "/m3ua", "/camel_sessions",
               "/camel_request", "/application", "/configuration"]
```

---

# عمليات CAP المدعومة

## العمليات الواردة (من gsmSSF → gsmSCF)

| العملية | Opcode | الوصف | لمعالج |
|---|---|---|---|
| **InitialDP** | 0 | نقطة الكشف الأولية - إشعار إعداد المكالمة | `handle_initial_dp/1` |
| **EventReportBCSM** | 6 | حدث نموذج حالة المكالمة الأساسي (رد، فصل، إلخ) | `handle_event_report_` |
| **ApplyChargingReport** | 71 | تقرير الشحن من gsmSSF | `handle_apply_chargin` |
| **AssistRequestInstructions** | 16 | طلب المساعدة من gsmSRF | `handle_assist_reques` |

### العمليات الصادرة (من gsmSCF → gsmSSF)

| العملية | Opcode | الوصف | |
|---|---|---|---|
| **Connect** | 20 | توصيل المكالمة برقم الوجهة | `CapRequestGenerator.` |
| **Continue** | 31 | متابعة معالجة المكالمة دون تعديل | `CapRequestGenerator.` |
| **ReleaseCall** | 22 | إنهاء/فصل المكالمة | `CapRequestGenerator.` |
| **RequestReportBCSMEvent** | 23 | طلب إشعار بأحداث المكالمات | `CapRequestGenerator.` |
| **ApplyCharging** | 35 | تطبيق الشحن على المكالمة | `CapRequestGenerator.` |

---

# ميزات واجهة الويب

## صفحة جلسات CAMEL

**الرابط:** `http://localhost/camel_sessions`

مراقبة في الوقت الفعلي لجلسات المكالمات النشطة في CAMEL:

**الميزات:**

- **قائمة الجلسات الحية** - تتجدد تلقائيًا كل 2 ثانية
- **تفاصيل الجلسة** - OTID، معرف المكالمة، الحالة، المدة
- **إصدار CAP** - يعرض إصدار البروتوكول (CAP v1/v2/v3/v4) المكتشف من InitialDP
- **معلومات المكالمة** - IMSI، رقم A، رقم B، مفتاح الخدمة
- **تتبع الحالة** - تم البدء، تم الرد، تم إنهاء
- **مؤقت المدة** - عرض مدة المكالمة في الوقت الفعلي

**أعمدة الجدول:**

- رقم المتصل، رقم المتصل عليه، مفتاح الخدمة، IMSI، معرف المكالمة، الحالة، الإصدار، OTID، المدة، وقت البدء

**حالات الجلسة:**

- □ **تم البدء** - تم استلام InitialDP، في انتظار الرد
- □ **تم الرد** - تم الرد على المكالمة، الشحن جارٍ
- □ **تم إنهاء** - انتهت المكالمة، تم إنشاء CDR

**كشف إصدار CAP:** يكتشف النظام تلقائيًا إصدار بروتوكول CAP من جزء الحوار في InitialDP ويعرضه في عمود الإصدار. يساعد ذلك في تحديد أي إصدار CAP يستخدمه كل MSC.

# منشئ طلب CAMEL

**الرابط:** `http://localhost/camel_request`

أداة تفاعلية لبناء وإرسال طلبات CAP:

**الميزات:**

- **محدد نوع الطلب** - InitialDP، Connect، ReleaseCall، إلخ.
- **حقول نموذج ديناميكية** - تتكيف مع نوع الطلب المحدد
- **خيارات SCCP/M3UA** - تكوين عنوان متقدم
- **تاريخ الطلبات** - آخر 20 طلبًا مع الحالة

- للطلبات المتابعة OTID يحتفظ بـ - **تتبع الجلسة**
- رسائل النجاح/الخطأ - **ردود فعل في الوقت الفعلي**

**أنواع الطلبات:**

1. **InitialDP** - بدء جلسة مكالمة جديدة
    - مفتاح الخدمة (عدد صحيح)
    - رقم المتصل (A-party)
    - رقم المتصل عليه (B-party)
2. **Connect** - توجيه المكالمة إلى الوجهة
    - رقم الوجهة
3. **ReleaseCall** - إنهاء المكالمة
    - رمز السبب (16=عادي، 17=مشغول، 31=غير محدد)
4. **RequestReportBCSMEvent** - طلب إشعارات الأحداث
    - الأحداث: oAnswer، oDisconnect، tAnswer، tDisconnect
5. **Continue** - متابعة المكالمة دون تعديل
    - لا تتطلب أي معلمات
6. **ApplyCharging** - تطبيق حدود مدة المكالمة
    - المدة (بالثواني، 1-864000)
    - الإفراج عند انتهاء المهلة (boolean)
    - للاستخدام التفصيلي CAMEL انظر دليل منشئ طلب

**خيارات SCCP المتقدمة:**

- عنوان الطرف المتصل العالمي
- عنوان الطرف المتصل عليه العالمي
- SSN المتصل عليه (146 :افتراضي = gsmSSF)
- SSN (146 :المتصل (افتراضي

**خيارات M3UA:**

- OPC (رمز النقطة الأصلية، افتراضي: 5013)
- DPC (رمز النقطة الوجهة، افتراضي: 5011)

---

# التكامل مع OCS

## دورة حياة المكالمة مع الشحن

### 1. بدء المكالمة (InitialDP)

عندما يرسل MSC InitialDP، يقوم CAMELGW:

1. **كشف إصدار CAP** - يفحص جزء الحوار لتحديد CAP v1/v2/v3/v4
2. **فك شفرة رسالة CAP** - يستخرج IMSI، أرقام المتصل/المتصل عليه
3. **يتصل بـ OCS** - API `InitiateSession`
4. **يتلقى التفويض** - MaxUsage (على سبيل المثال، 30 ثانية)
5. **يخزن الجلسة** - في SessionStore (جدول ETS) مع إصدار CAP
6. **يستجيب لـ MSC** - RequestReportBCSMEvent + Continue (باستخدام نفس إصدار CAP)

**مثال:**

```
# بيانات InitialDP المفككة
%{
  imsi: "310150123456789",
  calling_party_number: "14155551234",
  called_party_number: "14155556789",
  service_key: 1,
  msc_address: "19216800123",
  cap_version: :v2  # تم اكتشافه من الحوار
}

# استجابة OCS
{:ok, %{max_usage: 30}}  # 30 ثانية مصرح بها

# إدخال SessionStore
%{
  call_id: "CAMEL-4B000173",
  initial_dp_data: %{...},
  cap_version: :v2,  # مخزنة لتوليد الاستجابة
  start_time: 1730246400,
  state: :initiated
}
```

### 2. رد المكالمة (EventReportBCSM - oAnswer)

عندما يتم الرد على المكالمة:

1. **يتلقى حدث oAnswer** - من MSC
2. **يحدث OCS** - `UpdateSession` مع الاستخدام=0
3. **يب��أ حلقة الخصم** - يبدأ الشحن OCS
4. **يحدث حالة الجلسة** - `:answered` في SessionStore
5. **يواصل المكالمة** - يرسل Continue إلى MSC

### 3. التحديثات الدورية (اختياري)

للمكالمات الطويلة، اطلب رصيدًا إضافيًا:

```
# كل 30 ثانية
OCS.Client.update_session(call_id, %{}, current_usage)
```

إذا كانت MaxUsage أرسل ← فإن المشترك ليس لديه رصيد 0، تعيد ReleaseCall

**4. إنهاء المكالمة (EventReportBCSM - oDisconnect)**

عندما تنتهي المكالمة:

1. **يتلقى حدث oDisconnect** - من MSC
2. **يحسب إجمالي المدة** - من وقت بدء الجلسة
3. **ينهي جلسة OCS** - API `TerminateSession`
4. **تم إنشاء CDR** - بواسطة OCS مع التكلفة النهائية
5. **ينظف الجلسة** - يزيلها من SessionStore
6. **يرسل ReleaseCall** - يؤكد إنهاء المكالمة إلى MSC

# تحليل CDR

يتم إنشاء CDRs بواسطة OCS الخاص بك وعادة ما تتضمن:

**حقول CDR من CAMEL:**

- `Account` - IMSI أو رقم المتصل
- `Destination` - رقم الطرف المتصل عليه
- `OriginID` - معرف المكالمة الفريد (CAMEL-OTID)
- `Usage` - إجمالي مدة المكالمة (بالثواني)
- `Cost` - التكلفة المحسوبة
- `IMSI` - IMSI المشترك
- `CallingPartyNumber` - الطرف المتصل
- `CalledPartyNumber` - الطرف المتصل عليه
- `MSCAddress` - رمز نقطة MSC الخادمة
- `ServiceKey` - مفتاح خدمة CAMEL

---

# الاختبار

## الاختبار اليدوي باستخدام منشئ الطلب

1. **انتقل إلى منشئ الطلب:**

   ```
   http://localhost/camel_request
   ```

2. **أرسل InitialDP:**

   - اختر "InitialDP" من القائمة المنسدلة
   - مفتاح الخدمة: `100`
   - رقم المتصل: `14155551234`
   - رقم المتصل عليه: `14155556789`
   - انقر على "إرسال طلب InitialDP"
   - لاحظ OTID الذي تم إنشاؤه

3. **راقب الجلسة:**

   - افتح علامة تبويب جديدة: `http://localhost/camel_sessions`
   - شاهد الجلسة النشطة مع الحالة "تم البدء"

4. **محاكاة رد المكالمة:**

   - عد إلى منشئ الطلب
   - اختر "EventReportBCSM"
   - نوع الحدث: `oAnswer`
   - انقر على "إرسال طلب EventReportBCSM"
   - تتغير حالة الجلسة إلى "تم الرد"

5. **إنهاء المكالمة:**

   - اختر "ReleaseCall"
   - رمز السبب: `16` (عادي)
   - انقر على "إرسال طلب ReleaseCall"
   - تتغير حالة الجلسة إلى "تم إنهاء"

# حقيقي MSC الاختبار باستخدام

### تكوين خدمة CAMEL في MSC

على MSC/VLR الخاص بك، قم بتكوين خدمة CAMEL:

```
# مثال على تكوين MSC من Huawei
ADD CAMELSERVICE:
  SERVICEID=1,
  SERVICEKEY=100,
  GSMSCFADDR="55512341234",  # عنوان CAMELGW العالمي
  DEFAULTCALLHANDLING=CONTINUE;

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

### مراقبة السجلات

راقب سجلات CAMELGW لرسائل CAP الواردة:

```
# عرض السجلات في الوقت الفعلي
tail -f /var/log/omniss7/omniss7.log

# تصفية أحداث CAP
grep "CAP:" /var/log/omniss7/omniss7.log

# عرض سجل الأحداث (بتنسيق JSON)
curl http://localhost/api/events | jq '.[] | select(.map_event |
startswith("CAP:"))'
```

# اختبار التحميل

استخدم منشئ الطلب في حلقة لاختبار التحميل:

```
# طلب 100 إرسال 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": "1415555'$i'",
      "called_number": "14155556789"
    }'
  sleep 0.1
done
```

---

# المراقبة والعمليات

## مقاييس Prometheus

تقوم CAMELGW بعرض المقاييس على `http://localhost:8080/metrics`:

**المقاييس الخاصة بـ CAP:**

- `cap_requests_total{operation}` - إجمالي طلبات CAP حسب نوع العملية (مثل initialDP، requestReportBCSMEvent)

**مقاييس MAP/API الإضافية:**

- `map_requests_total{operation}` - إجمالي طلبات MAP حسب نوع العملية
- `map_request_duration_milliseconds{operation}` - هيستوغرام مدة الطلب
- `map_pending_requests` - عدد المعاملات MAP المعلقة

**مقاييس M3UA STP** (إذا تم تمكين وضع STP):

- `m3ua_stp_messages_received_total{peer_name,point_code}` - الرسائل المستلمة من الأقران
- `m3ua_stp_messages_sent_total{peer_name,point_code}` - الرسائل المرسلة إلى الأقران

- `m3ua_stp_routing_failures_total{reason}` - فشل التوجيه حسب السبب

**استعلامات المثال:**

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

# إجم��لي InitialDP المستلمة
curl http://localhost:8080/metrics | grep
'cap_requests_total{operation="initialDP"}'

# طلبات MAP المعلقة
curl http://localhost:8080/metrics | grep map_pending_requests
```

## فحوصات الصحة

```
# تحقق من اتصال M3UA
curl http://localhost/api/m3ua-status

# تحقق من اتصال OCS
curl http://localhost/api/ocs-status

# تحقق من الجلسات النشطة
curl http://localhost/api/camel/sessions/count
```

## تكوين السجلات

قم بتعديل مستوى السجل في `config/runtime.exs`:

```
config :logger,
  level: :info  # الخيارات: :debug، :info، :warning، :error

# تفعيل تسجيل تصحيح CAP
config :logger, :console,
  metadata: [:cap_operation, :otid, :call_id]
```

---

# استكشاف الأخطاء وإصلاحها

## المشكلة: عدم تلقي رسائل CAP

**الأعراض:** يعمل منشئ الطلب، لكن MSC لا يرسل InitialDP

**تحقق من:**

1. حالة ارتباط M3UA: `curl http://localhost/api/m3ua-status`
2. تكوين خدمة CAMEL في MSC (مفتاح الخدمة، عنوان gsmSCF)
3. توجيه SCCP (يجب أن يوجه العنوان العالمي إلى CAMELGW)
4. قواعد جدار الحماية (السماح بمنفذ SCTP 2905)

**الحل:**

```
# تحقق من اتصال M3UA
tcpdump -i eth0 sctp

# تحقق مما إذا كان MSC يمكنه الوصول إلى CAMELGW
ss -tuln | grep 2905
```

## المشكلة: أخطاء OCS

**الأعراض:** `INSUFFICIENT_CREDIT` أو أخطاء المهلة

**تحقق من:**

1. يمكن الوصول إليه OCS: `curl http://your-ocs-server/api/health`
2. الحساب لديه رصيد في OCS
3. تم تكوين خطة التصنيف في OCS
4. الاتصال الشبكي بـ OCS
5. رمز المصادقة صالح (إذا لزم الأمر)

**الحل:**

- تحقق من تكوين عنوان OCS في `runtime.exs`
- تحقق من سجلات OCS للأخطاء

- اختبار واجهة برمجة تطبيقات OCS يدويًا باستخدام curl
- تحقق من قواعد جدار الحماية للسماح بالاتصال

## المشكلة: الجلسة غير موجودة

**الأعراض:** يفشل EventReportBCSM مع "الجلسة غير موجودة"

**السبب:** عدم تطابق OTID أو انتهاء صلاحية الجلسة

**الحل:**

1. تحقق من OTID في السجلات
2. تحقق من مهلة الجلسة (الافتراضي: لا انتهاء)
3. تأكد من تطابق OTID في الرسائل Continue/End مع DTID

```
# تحقق من الجلسات النشطة
iex> CAMELGW.SessionStore.list_sessions()
```

## المشكلة: أخطاء فك التشفير

**الأعراض:** `InitialDP` في السجلات `فشل في فك تشفير`

**السبب:** عدم تطابق إصدار CAP أو رسالة مشوهة

**الحل:**

1. تحقق من تطابق تكوين إصدار CAP مع MSC
2. تحقق من صحة ترميز ASN.1
3. التقاط PCAP وتحليله باستخدام Wireshark

```
# التقاط رسائل CAP
tcpdump -i eth0 -w cap_trace.pcap sctp port 2905

# تحليل باستخدام Wireshark (تصفية: m3ua)
wireshark cap_trace.pcap
```

---

# التكوين المتقدم

## إصدارات CAP متعددة

دعم إصدارات CAP المختلفة لكل مفتاح خدمة:

```
config :omniss7,
  cap_version_map: %{
    100 => :v2,  # يستخدم 100 الخدمة مفتاح CAP v2
    200 => :v3,  # يستخدم 200 الخدمة مفتاح CAP v3
    300 => :v4   # يستخدم 300 الخدمة مفتاح CAP v4
  },
  cap_version: :v2  # الإعداد الافتراضي
```

---

# الملخص

يمكن وضع بوابة CAMEL OmniSS7 مع كمنصة شبكة ذكية كاملة من العمل:

 **دعم كامل لبروتوكول CAP** (v1/v2/v3/v4)
 **شحن في الوقت الفعلي** عبر تكامل OCS
 **عمليات التحكم في المكالمات** (Connect، Release، Continue)
 **إدارة الجلسات** مع تخزين ETS
 **اختبار تفاعلي** عبر واجهة الويب لمنشئ الطلب
 **مراقبة حية** لجلسات المكالمات النشطة
 **توليد CDR** للفوترة والتحليلات
 **أداء موثوق به** وجاهز للإنتاج

للحصول على مزيد من المعلومات:

- وثائق منشئ طلب CAMEL
- المرجع الفني - عمليات CAP

---

**المنتج:** بوابة OmniSS7 CAMEL
**إصدار الوثائق:** 1.0
**آخر تحديث:** 26-10-2025

# دليل الميزات الشائعة

← العودة إلى الوثائق الرئيسية

يغطي هذا الدليل الميزات الشائعة لجميع أوضاع تشغيل OmniSS7.

## جدول المحتويات

1. نظرة عامة على واجهة الويب
2. وثائق API
3. المراقبة والقياسات
4. أفضل الممارسات
5. تعدد العناوين في SCTP لزيادة موثوقية الشبكة

---

## نظرة عامة على واجهة الويب

واجهة الويب متاحة عبر عنوان خادم الويب المكون لديك.

## التنقل الرئيسي

- في الوقت الحقيقي وسجلات الرسائل SS7 **الأحداث** - أحداث إشارة
- **التطبيق** - حالة التطبيق ومعلومات وقت التشغيل
- **التكوين** - عارض تكوين النظام
- **حالة M3UA** - اتصالات نظير M3UA (وضع STP)
- **قائمة انتظار SMS** - رسائل SMS الصادرة (وضع SMSc)

## الوصول إلى واجهة الويب

1. افتح متصفح الويب الخاص بك
2. انتقل إلى اسم المضيف المكون (مثل `http://localhost`)
3. عرض لوحة معلومات حالة النظام

## وثائق API Swagger

وثائق API التفاعلية:

```
http://your-server/swagger
```

## تكوين واجهة الويب

تكوين في `config/runtime.exs`:

```
config :control_panel,
  # ترتيب الصفحات في قائمة التنقل
  page_order: ["/events", "/application", "/configuration"],

  # إعدادات خادم الويب
  web: %{
    listen_ip: "0.0.0.0",    # IP للربط (0.0.0.0 لجميع الواجهات)
    port: 80,                 # منفذ HTTP (443 لـ HTTPS)
    hostname: "localhost",    # اسم المضيف للخادم لتوليد URL
    enable_tls: false,        # تعيين true لتمكين HTTPS
    tls_cert: "cert.pem",    # مسار ملف شهادة TLS
    tls_key: "key.pem"       # مسار ملف المفتاح الخاص بـ TLS
  }
```

**معلمات التكوين:**

| المعلمة | النوع | الافتراضي | الوصف |
|---|---|---|---|
| `page_order` | قائمة | `["/events", "/application", "/configuration"]` | ترتيب الصفحات في قائمة التنقل |
| `listen_ip` | سلسلة | `"0.0.0.0"` | عنوان IP لربط خادم الويب |
| `port` | عدد صحيح | `80` | منفذ HTTP (استخدم 443 لـ HTTPS) |
| `hostname` | سلسلة | `"localhost"` | اسم المضيف للخادم لتوليد URL |
| `enable_tls` | منطقي | `false` | تمكين HTTPS مع TLS |
| `tls_cert` | سلسلة | `"cert.pem"` | مسار شهادة TLS (عند تمكين TLS) |
| `tls_key` | سلسلة | `"key.pem"` | مسار المفتاح الخاص بـ TLS (عند تمكين TLS) |

## تكوين المسجل

تكوين مستوى التسجيل في `config/runtime.exs`:

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

**مستويات السجل:**

- `:debug` - معلومات تصحيح مفصلة
- `:info` - رسائل معلومات عامة

- `:warning` - رسائل تحذير لمشاكل محتملة
- `:error` - رسائل خطأ فقط

---

# وثائق API

## عنوان URL الأساسي لـ API

```
http://your-server/api
```

## رموز الاستجابة

- **200** - نجاح
- **400** - طلب غير صالح
- **504** - مهلة البوابة

## مواصفات OpenAPI

```
http://your-server/swagger.json
```

---

# المراقبة والقياسات

## نقطة نهاية قياسات Prometheus

```
http://your-server/metrics
```

## فئات القياسات الرئيسية

**قياسات M3UA/SCTP:**

- تغييرات حالة ارتباط SCTP
- انتقالات حالة ASP في M3UA
- وحدات البيانات البروتوكول المرسلة/المستلمة

**قياسات M2PA:**

- انتقالات حالة الرابط (DOWN → ALIGNMENT → PROVING → READY)
- الرسائل والبايتات المرسلة/المستلمة لكل رابط
- أخطاء محددة للرابط (SCTP، فك التشفير، تشفير)

**قياسات STP:**

- الرسائل المستلمة/المرسلة لكل نظير
- فشل التوجيه حسب السبب
- توزيع الحركة عبر الأقران

**قياسات عميل MAP:**

- طلبات MAP حسب نوع العملية
- مخططات مدة الطلب
- مقياس المعاملات المعلقة

**قياسات CAP:**

- طلبات CAP حسب نوع العملية
- عمليات بوابة CAMEL

**قياسات SMSc:**

- عمق قائمة الانتظار
- معدلات التسليم
- الرسائل الفاشلة

# تكامل Grafana

قياسات OmniSS7 متوافقة مع Prometheus وGrafana.

---

# أفضل الممارسات

## توصيات الأمان

1. **عزل الشبكة**
    - نشر في VLAN مخصص
    - قواعد جدار الحماية لتقييد الوصول
    - السماح لـ SCTP فقط من عناوين معروفة
2. **أمان واجهة الويب**
    - تمكين TLS للإنتاج
    - استخدام وكيل عكسي مع مصادقة
    - تقييد الوصول إلى عناوين IP الإدارية
3. **أمان API**
    - تنفيذ تحديد المعدل
    - استخدام مفاتيح API أو OAuth
    - تسجيل جميع الطلبات للتدقيق

## تحسين الأداء

1. **حدود TPS**
    - تكوين TPS مناسب
    - مراقبة تحميل النظام
    - ضبط مخازن SCTP
2. **تحسين قاعدة البيانات**
    - إضافة فهارس
    - أرشفة الرسائل القديمة
    - مراقبة مجموعة الاتصالات
3. **ضبط M3UA**

- ضبط فترات نبض SCTP
- تكوين قيم المهلة
- استخدام روابط متعددة للموثوقية

---

# لزيادة موثوقية SCTP تعدد العناوين في الشبكة

## ما هو تعدد العناوين في SCTP؟

**تعدد العناوين في SCTP** هو ميزة مدمجة في بروتوكول SCTP تسمح لاتصال واحد M3UA بالربط إلى عدة عناوين IP على نفس واجهة الشبكة أو عبر واجهات شبكة مختلفة. وهذا يوفر فشل تلقائي وزيادة موثوقية في طبقة النقل.

**الفوائد الرئيسية:**

- **فشل تلقائي** SCTP: إذا فشل مسار شبكة واحد، يقوم بالتبديل تلقائيًا إلى مسار بديل دون قطع الاتصال
- **فشل بدون تكوين** SCTP: لا حاجة إلى منطق على مستوى التطبيق - يتولى مراقبة المسار والفشل
- **موثوقية محسّنة**: البقاء على قيد الحياة في حالات فشل الشبكة، وفشل المفاتيح، أو فشل NIC
- **توازن الحمل** SCTP: يمكن لـ توزيع الحركة عبر مسارات متعددة (يعتمد على التنفيذ)

## كيف يعمل

عند تكوين عدة عناوين IP لاتصال M3UA، يقوم SCTP:

1. **الربط بجميع IPs**: يقوم المقبس بالربط بجميع عناوين IP المكونة في الوقت نفسه
2. **مراقبة المسارات** SCTP: يرسل باستمرار حزم نبض القلب على جميع المسارات لمراقبة صحتها
3. **الكشف عن الفشل** SCTP: إذا فشلت نبضات القلب على المسار الرئيسي، يقوم بوضع علامة عليه على أنه غير متاح
4. **فشل تلقائي**: يتم التبديل الفوري للحركة إلى مسار احتياطي دون تدخل التطبيق

5. **استعادة المسار**: عندما يتعافى المسار الفاشل، يقوم SCTP باكتشافه ويضع علامة عليه كمتاح مرة أخرى

# التكوين

يتم تكوين تعدد العناوين في SCTP **قائمة من عناوين IP** بدلاً من مجموعة **IP** عن طريق توفير واحدة.

**IP واحد (تقليدي)**

```
# IP واحد - لا يوجد تعدد عناوين
local_ip: {10, 179, 4, 10}
```

**عدة IPs (تم تمكين تعدد العناوين)**

```
# عدة IPs - تم تمكين تعدد العناوين
# IP الأول هو الرئيسي، والـ IPs اللاحقة هي مسارات احتياطية
local_ip: [{10, 179, 4, 10}, {10, 179, 4, 11}]
```

# أمثلة التكوين

**المثال 1: نظير STP مع تعدد العناوين**

```
# تكوين نظير في وضع STP
config :omniss7,
  m3ua_peers: [
    %{
      peer_id: 1,
      name: "Partner_STP_Redundant",
      role: :client,
      # تعدد العناوين: الربط إلى عنوانين IP محليين للموثوقية
      local_ip: [{213, 57, 23, 200}, {213, 57, 23, 201}],
      local_port: 0,
      # النظير البعيد يدعم أيضًا تعدد العناوين
      remote_ip: [{213, 57, 23, 100}, {213, 57, 23, 101}],
      remote_port: 2905,
      routing_context: 1,
      point_code: 100,
      network_indicator: :international
    }
  ]
```

**المثال 2: عميل MAP مع تعدد العناوين**

```
# وضع عميل MAP مع تعدد العناوين
config :omniss7,
  map_client_enabled: true,
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :hlr_client_asp,
    # تعدد العناوين: عنوانان محليان للفشل
    local_ip: [{10, 0, 0, 100}, {10, 0, 0, 101}],
    local_port: 2905,
    # STP البعيد مع دعم تعدد العناوين
    remote_ip: [{10, 0, 0, 1}, {10, 0, 0, 2}],
    remote_port: 2905,
    routing_context: 1
  }
```

**المثال 3: مستمع STP مع تعدد العناوين**

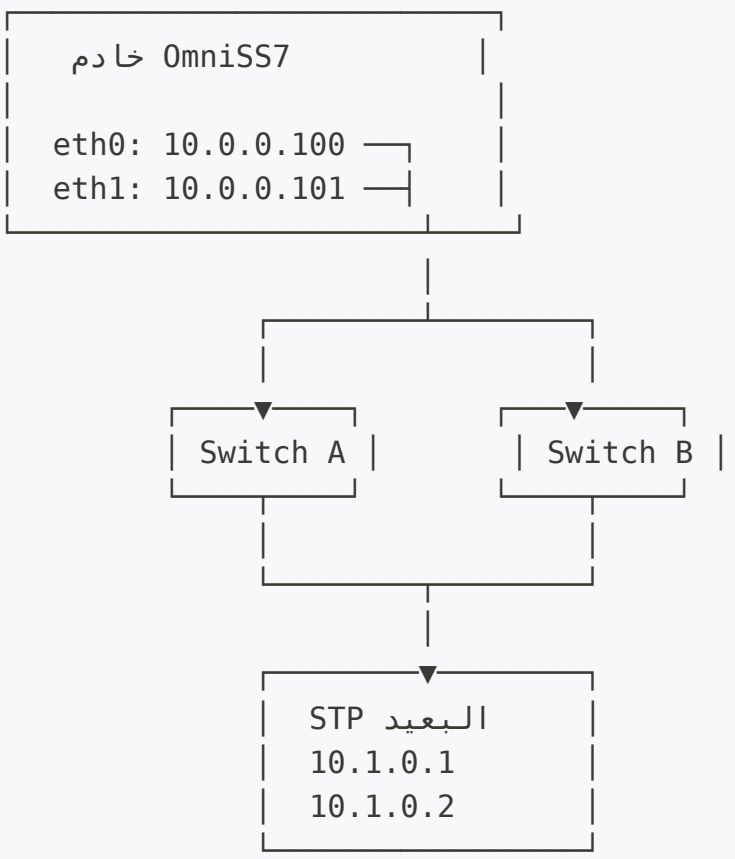
```
# خادم STP مستقل مع تعدد العناوين
config :omniss7,
  m3ua_stp: %{
    enabled: true,
    # الاستماع IPs للاتصالات الواردة ��لى عدة
    local_ip: [{172, 16, 0, 10}, {172, 16, 0, 11}],
    local_port: 2905,
    point_code: 100
  }
```

**المثال 4: تكوين مختلط (متوافق مع الإصدارات السابقة)**

```
# مزيج من الأقران ذات IP واحد ومتعدد العناوين
config :omniss7,
  m3ua_peers: [
    # نظير قديم - IP واحد
    %{
      peer_id: 1,
      name: "Legacy_STP",
      role: :client,
      local_ip: {10, 0, 0, 1},      # مجموعة IP واحدة
      local_port: 0,
      remote_ip: {10, 0, 0, 10},
      remote_port: 2905,
      routing_context: 1,
      point_code: 100
    },
    # نظير جديد - تعدد العناوين
    %{
      peer_id: 2,
      name: "Redundant_STP",
      role: :client,
      local_ip: [{10, 0, 0, 2}, {10, 0, 0, 3}],  # قائمة IP
      local_port: 0,
      remote_ip: [{10, 0, 0, 20}, {10, 0, 0, 21}],
      remote_port: 2905,
      routing_context: 2,
      point_code: 200
    }
  ]
```

# سيناريوهات الطوبولوجيا الشبكية

### السيناريو 1: بطاقات NIC (نشر شائع) مزدوجة



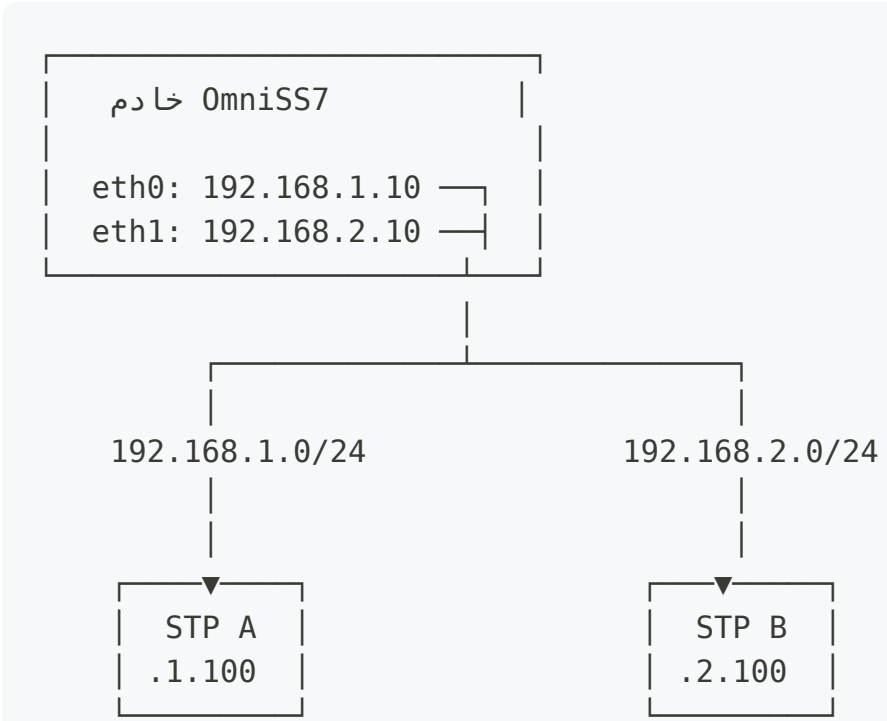

**التكوين:**

```
local_ip: [{10, 0, 0, 100}, {10, 0, 0, 101}]   # NIC كلا بطاقتي
remote_ip: [{10, 1, 0, 1}, {10, 1, 0, 2}]      # النظير البعيد
```

**الفوائد:**

- البقاء على قيد الحياة في حالة فشل بطاقة NIC واحدة
- البقاء على قيد الحياة في حالة فشل مفتاح واحد
- فشل تلقائي ف�� أقل من 1 ثانية

### السيناريو 2: عدة شبكات فرعية




**التكوين:**

```
local_ip: [{192, 168, 1, 10}, {192, 168, 2, 10}]
remote_ip: [{192, 168, 1, 100}, {192, 168, 2, 100}]
```

**الفوائد:**

- البقاء على قيد الحياة في حالة فشل الشبكة الفرعية
- إمكانية وجود موثوقية جغرافية
- مسارات توجيه مست��لة

# المراقبة والتسجيل

عند تمكين تعدد العناوين، سترى رسائل سجل تشير إلى التكوين:

**تعدد العناوين الناجح**

```
[info] عميل SCTP عنوان 2 بـ الربط تم :متعدد العناوين IP محلي
[info] تم تمكين تعدد العناوين لمستمع STP: تم الربط بـ 2 عنوان IP
محلي
```

**أحداث فشل المسار**

```
[warning] [MULTIHOMING] المسار 10.0.0.100 غير متاح للنظير
Partner_STP (assoc_id=1)
[info] [MULTIHOMING] المسار 10.0.0.101 هو الآن الرئيسي للنظير
Partner_STP (assoc_id=1)
[info] [MULTIHOMING] المسار 10.0.0.100 متاح الآن للنظير Partner_STP
(assoc_id=1)
```

# عرض واجهة الويب

تعرض واجهة الويب تلقائيًا معلومات تعدد العناوين:

**صفحة حالة M3UA:**

- **IP واحد**: يظهر كـ `10.0.0.100`
- **عدة IPs**: يظهر كـ `10.0.0.100 (1+)` أو `10.0.0.100 (2+)`
- **عرض التفاصيل**: يظهر جميع IPs احتياطية/مع تسميات رئيسية

# أفضل الممارسات

**1. تصميم الشبكة**

- **استخدم بطاقات NIC مختلفة** لتحقيق أقصى موثوقية
- **مفاتيح مختلفة** للبقاء على قيد الحياة في حالات فشل المفاتيح
- **شبكات فرعية مختلفة** إذا كان ذلك ممكنًا لتنوع التوجيه
- **نفس مركز البيانات في البداية** - اختبار قبل الفصل الجغرافي

**2. تخطيط عنوان IP**

- **الأول هو الرئيسي IP** - تأكد من أنه على المسار الأكثر موثوقية
- **ترتيب مهم** - قم بإدراج IPs بترتيب التفضيل
- **توجيه متسق** - استخدم أنظمة توجيه مشابهة لتسهيل استكشاف الأخطاء

**3. اختبار الفشل**

```
# تعطيل الواجهة الرئيسية لاختبار الفشل
sudo ip link set eth0 down

# مراقبة السجلات للفشل
tail -f /var/log/omniss7.log | grep MULTIHOMING

# إعادة تمكين الواجهة
sudo ip link set eth0 up
```

**4. يجب أن يدعم كلا الجانبين تعدد العناوين**

- **مثالي**: يستخدم كل من المحلي والبعيد عدة IPs
- **مقبول**: يستخدم جانب واحد فقط تعدد العناوين
- **ملاحظة**: تكون الموثوقية أفضل عندما يدعم كلا الطرفين ذلك

**5. تكوين جدار الحماية**

```
# السماح لـ SCTP على جميع IPs متعددة العناوين
iptables -A INPUT -p sctp --dport 2905 -s 10.0.0.0/24 -j ACCEPT
iptables -A INPUT -p sctp --dport 2905 -s 10.1.0.0/24 -j ACCEPT
```

# استكشاف الأخطاء

**المشكلة: عدم عمل تعدد العناوين**

**الأعراض**: يتم استخدام IP الرئيسي فقط، ولا يوجد فشل

**التحقق:**

1. تحقق من دعم Erlang لـ SCTP: `erl -eval 'gen_sctp:open(9999, [binary, {ip, {127,0,0,1}}]).'`
2. تحقق من وحدة SCTP في النواة: `lsmod | grep sctp`
3. إذا لزم الأمر قم بتحميل SCTP: `sudo modprobe sctp`
4. تحقق من تكوين كلا IPs على النظام: `ip addr show`

**المشكلة: المسار لا يفشل**

**الأعراض**: تم وضع علامة على المسار الرئيسي كـ "معطل" ولكن الحركة لا تتبدل

**التحقق:**

1. تحقق من إعدادات نبض SCTP
2. تحقق من أن جدول التوجيه يحتوي على مسارات لجميع المسارات
3. تحقق من أن جدار الحماية يسمح لـ IPs على جميع SCTP
4. راجع سجلات مراقبة مسار SCTP

**المشكلة: تذبذب المسار المتكرر**

**الأعراض**: تتبدل المسارات باستمرار بين UP ��DOWN

**التحقق:**

1. عدم استقرار الشبكة - تحقق من الروابط الفيزيائية
2. نبض SCTP سريع جدًا - قد يحتاج إلى ضبط
3. جدار الحماية يقوم بإسقاط نبضات SCTP
4. مشاكل MTU على مسار واحد

# اعتبارات الأداء

- **أقل تأثير**: نبضات SCTP صغيرة وغير متكررة
- **لا تغييرات على التطبيق**: تعدد العناوين شفاف لطبقة التطبيق
- **فشل سريع**: عادةً أقل من 1 ثانية للكشف والفشل
- **استعادة تلقائية**: لا حاجة لتدخل يدوي

# التوافق

- **متوافق مع الإصدارات السابقة**: لا يزال تنسيق مجموعة IP الواحدة يعمل
- **نشر مختلط**: يمكن مزج الأقران ذات IP واحد ومتعدد العناوين
- **جميع الأوضاع مدعومة**: يعمل في أوضاع MAP Client وSMSc وHLR وSTP
- **متطلبات Erlang**: يتطلب Erlang مع دعم SCTP مضمن

# المراقبة والتنبيه

**المقاييس الرئيسية:**

- حالة اتصال M3UA

- معدل نجاح طلبات MAP
- أوقات استجابة API
- عمق قائمة الرسائل

**حدود التنبيه:**

- معطل�� < 1 دقيقة M3UA
- معدل مهلة MAP > 10%
- عمق قائمة الانتظار > 1000
- معدل خطأ API > 5%

---

# مرجع التكوين الكامل

## جميع معلمات التكوين

يوفر هذا القسم مرجعًا كاملاً لجميع معلمات التكوين المتاحة عبر جميع أوضاع التشغيل.

**تكوين المسجل (`:logger`)**

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

---

**تكوين واجهة الويب (`: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"
  }
```

| المعلمة | النوع | مطلوب | الافتراضي | الوصف |
|---|---|---|---|---|
| `page_order` | قائمة من السلاسل | لا | `["/events", "/application", "/configuration"]` | ترتيب صفحات قائمة التنقل |
| `web.listen_ip` | سلسلة | نعم | `"0.0.0.0"` | عنوان IP لربط خادم الويب |
| `web.port` | عدد صحيح | نعم | `80` | رقم منفذ HTTP/HTTPS |
| `web.hostname` | سلسلة | نعم | `"localhost"` | اسم المضيف للخادم |
| `web.enable_tls` | منطقي | لا | `false` | تمكين HTTPS |
| `web.tls_cert` | سلسلة | إذا تم تمكين TLS | `"cert.pem"` | مسار شهادة TLS |
| `web.tls_key` | سلسلة | إذا تم تمكين TLS | `"key.pem"` | مسار المفتاح الخاص بـ TLS |

---

### تكوين M3UA STP (`:omniss7`)

```
config :omniss7,
  m3ua_stp: %{
    enabled: false,
    local_ip: {127, 0, 0, 1},
    local_port: 2905
  },
  enable_gt_routing: true,
  m3ua_peers: [...],
  m3ua_routes: [...],
  m3ua_gt_routes: [...]
```

| المعلمة | النوع | مطلوب | الافتراضي | الوصف |
|---|---|---|---|---|
| `m3ua_stp.enabled` | منطقي | نعم | `false` | تمكين وضع STP عند بدء التشغيل |
| `m3ua_stp.local_ip` | مجموعة | نعم | `{127, 0, 0, 1}` | IP للربط للـ M3UA الوارد |
| `m3ua_stp.local_port` | عدد صحيح | نعم | `2905` | منفذ SCTP لـ M3UA |
| `enable_gt_routing` | منطقي | لا | `false` | تمكين توجيه العنوان العالمي |

**معلمات نظير M3UA:**

| المعلمة | النوع | مطلوب | الوصف |
|---|---|---|---|
| `peer_id` | عدد صحيح | ن��م | معرف نظير فريد |
| `name` | سلسلة | نعم | اسم وصفي للنظير |
| `role` | ذرة | نعم | `:server` أو `:client` |
| `local_ip` | مجموعة أو قائمة | إذا كان `:client` | IP(s) المحلي للربط. فردي: `{10, 0, 0, 1}` أو قائمة: `[{10, 0, 0, 1}, {10, 0, 0, 2}]` |
| `local_port` | عدد صحيح | إذا كان `:client` | المنفذ المحلي (0 للديناميكي) |
| `remote_ip` | مجموعة أو قائمة | نعم | IP(s) للنظير البعيد. فردي: `{10, 0, 0, 10}` أو قائمة: `[{10, 0, 0, 10}, {10, 0, 0, 11}]` |
| `remote_port` | عدد صحيح | إذا كان `:client` | منفذ النظير البعيد |
| `routing_context` | عدد صحيح | نعم | سياق توجيه M3UA |
| `point_code` | عدد صحيح | نعم | رمز نقطة SS7 |
| `network_indicator` | ذرة | لا | `:international` أو `:national` |

**معلمات توجيه M3UA:**

| المعلمة | النوع | مطلوب | الوصف |
|---|---|---|---|
| `dest_pc` | عدد صحيح | نعم | رمز نقطة الوجهة |
| `peer_id` | عدد صحيح | نعم | النظير الذي سيتم التوجيه من خلاله |
| `priority` | عدد صحيح | نعم | أولوية التوجيه (أقل = أولوية أع��ى) |
| `network_indicator` | ذرة | لا | `:national` أو `:international` |

**معلمات توجيه GT لـ M3UA:**

| المعلمة | النوع | مطلوب | الوصف |
|---|---|---|---|
| `gt_prefix` | سلسلة | نعم | بادئة العنوان العالمي للمطابقة |
| `peer_id` | عدد صحيح | نعم | النظير الوجهة |
| `priority` | عدد صحيح | نعم | أولوية التوجيه |
| `description` | سلسلة | لا | وصف التوجيه للتسجيل |
| `source_ssn` | عدد صحيح | لا | المصدر SSN المطابقة فقط إذا تطابقت |
| `dest_ssn` | عدد صحيح | لا | الوجهة إلى هذه القيمة SSN إعادة كتابة |

---

**تكوين عميل MAP (`:omniss7`)**

```
config :omniss7,
  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
  }
```

| المعلمة | النوع | مطلوب | الافتراضي |
| --- | --- | --- | --- |
| `map_client_enabled` | منطقي | نعم | `false` |
| `map_client_m3ua.mode` | سلسلة | نعم | `"ASP"` |
| `map_client_m3ua.callback` | مجموعة | نعم | `{MapClient, :handle_payload, []}` |
| `map_client_m3ua.process_name` | ذرة | نعم | `:map_client_asp` |
| `map_client_m3ua.local_ip` | مجموعة | نعم | - |
| `map_client_m3ua.local_port` | عدد صحيح | نعم | `2905` |
| `map_client_m3ua.remote_ip` | مجموعة | نعم | - |
| `map_client_m3ua.remote_port` | عدد صحيح | نعم | `2905` |

| المعلمة | النوع | مطلوب | الافتراضي |
| --- | --- | --- | --- |
| `map_client_m3ua.routing_context` | عدد صحيح | نعم | - |

---

### تكوين مركز SMS (`:omniss7`)

```
config :omniss7,
  auto_flush_enabled: false,
  auto_flush_interval: 10_000,
  auto_flush_dest_smsc: nil,
  auto_flush_tps: 10
```

| المعلمة | النوع | مطلوب | الافتراضي | الوصف |
| --- | --- | --- | --- | --- |
| `auto_flush_enabled` | منطقي | لا | `false` | تمكين التفريغ التلقائي لقائمة انتظار SMS |
| `auto_flush_interval` | عد�� صحيح | لا | `10000` | فترة استقصاء القائمة (بالملي ثانية) |
| `auto_flush_dest_smsc` | سلسلة/لا شيء | لا | `nil` | تصفية بواسطة الوجهة SMSC (nil = الكل) |
| `auto_flush_tps` | عدد صحيح | لا | `10` | الحد الأقصى للمعاملات في الثانية |

---

### تكوين واجهة API HTTP (`:omniss7`)

يستخدم خلفية SMS الآن واجهة API HTTP بقاعدة البيانات المباشرة بدلاً من الاتصالات.

```
config :omniss7,
  smsc_api_base_url: "https://10.5.198.200:8443",
  frontend_name: "omni-smsc01"  # اختياري: الافتراضي هو
hostname_SMSc
```

**معلمات API:**

| المعلمة | النوع | مطلوب | الافتراضي | ف |
|---|---|---|---|---|
| `smsc_api_base_url` | سلسلة | نعم | `"https://10.5.198.200:8443"` | ن U ي هة A ية SI |
| `frontend_name` | سلسلة | لا | `"{hostname}_SMSc"` | ف هة ية ل |

**نقاط النهاية المستخدمة في API:**

- `POST /api/frontends` - تسجيل هذه النسخة الأمامية مع الخلفية
- `POST /api/messages_raw` - إدراج رسائل SMS جديدة
- `GET /api/messages` - استرجاع قائمة الرسائل (مع رأس `smsc`)
- `PATCH /api/messages/{id}` - وضع علامة على الرسالة على أنها تم تسليمها
- `PUT /api/messages/{id}` - تحديث حالة الرسالة
- `POST /api/events` - إضافة تتبع الأحداث
- `GET /api/status` - نقطة نهاية التحقق من الصحة

**تسجيل الواجهة الأمامية:**

الخلفية عند بدء التشغيل ويعيد التسجيل كل 5 API يقوم النظام تلقائيًا بتسجيل نفسه مع واجهة دقائق. يتضمن التسجيل:

- اسم الواجهة الأمامية ونوعها (SMSc)
- اسم المضيف
- وقت التشغيل بالثواني
- تفاصيل التكوين (بتنسيق JSON)

**ملاحظات التكوين:**

- يتم تعطيل التحقق من SSL بشكل افتراضي للشهادات الموقعة ذاتيًا
- تنتهي مهلة طلبات HTTP بعد 5 ثوانٍ
- جميع الطوابع الزمنية بتنسيق ISO 8601
- تستخدم واجهة API JSON لأجسام الطلب/الاستجابة

---

# الوثائق ذات الصلة

- ← العودة إلى الوثائق الرئيسية
- دليل STP
- دليل عميل MAP
- دليل مركز SMS
- دليل HLR

---

**OmniSS7** بواسطة Omnitouch Network Services

# مرجع التكوين

← العودة إلى الوثائق الرئيسية

يوفر هذا المستند مرجعًا شاملاً لجميع معلمات تكوين OmniSS7.

## جدول المحتويات

1. نظرة عامة
2. علامات وضع التشغيل
3. معلمات وضع HLR
4. معلمات وضع SMSc
5. معلمات وضع STP
6. معلمات NAT لعنوان العنوان العالمي
7. معلمات اتصال M3UA
8. معلمات خادم HTTP
9. معلمات قاعدة البيانات
10. القيم الثابتة

---

## نظرة عامة

يتم إدارة تكوين OmniSS7 عبر `config/runtime.exs`. يدعم النظام ثلاثة أوضاع تشغيل:

- **وضع STP** - نقطة نقل الإشارة للتوجيه
- **وضع HLR** - سجل الموقع المنزلي لإدارة المشتركين
- **وضع SMSc** - مركز الرسائل القصيرة لتسليم الرسائل

**ملف التكوين**: `config/runtime.exs`

---

# علامات وضع التشغيل

تحكم في الميزات المفعلة.

| المعلمة | النوع | الافتراضي | الوصف | الأوضاع |
|---|---|---|---|---|
| `map_client_enabled` | Boolean | `false` | تمكين عميل MAP والاتصال بـ M3UA | الكل |
| `hlr_mode_enabled` | Boolean | `false` | تمكين الميزات الخاصة بـ HLR | HLR |
| `smsc_mode_enabled` | Boolean | `false` | تمكين الميزات الخاصة بـ SMSc | SMSc |

**مثال**:

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

---

# معلمات وضع HLR

تكوين وضع HLR (سجل الموقع المنزلي).

## HLR تكوين واجهة برمجة تطبيقات

| المعلمة | النوع | الافتراضي | مطلوب | صف |
|---|---|---|---|---|
| `hlr_api_base_url` | String | - | **نعم** | قطة نهاية<br>هة برمجة<br>ا تطبيقات<br>ل) الخلفية<br>ى SSL من<br>شكل ثابت |
| `hlr_service_center_gt_address` | String | - | **نعم** | ن العنوان<br>العالمي لـ<br>تم إرجاعه<br>استجابات<br>UpdateL |
| `smsc_service_center_gt_address` | String | - | **نعم** | عنوان GT<br>تم SMSc<br>رجاعه في<br>استجابات<br>for-SM |

**مثال**:

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

## تحويل MSISDN ↔ IMSI

تكوين لتوليد IMSI الاصطناعية من MSISDNs. للحصول على شرح تقني مفصل لخوارزمية التحويل، راجع تحويل MSISDN ↔ IMSI في دليل HLR.

| المعلمة | النوع | الافتراضي | مطلوب | الوصف |
|---|---|---|---|---|
| `hlr_imsi_plmn_prefix` | String | `"50557"` | لا | بادئة PLMN (MCC+MNC) لتوليد IMSI الاصطناعية |
| `hlr_msisdn_country_code` | String | `"61"` | لا | بادئة رمز الدولة للتحويل العكسي IMSI→MSISDN |
| `hlr_msisdn_nsn_offset` | Integer | `0` | لا | الإزاحة في MSISDN حيث يبدأ NSN (عادةً طول رمز الدولة) |
| `hlr_msisdn_nsn_length` | Integer | `9` | ا�� | طول رقم المشترك الوطني لاستخراجه من MSISDN |

**مثال** (رمز دولة مكون من رقمين):

```
config :omniss7,
  hlr_imsi_plmn_prefix: "50557",      # MCC 505 + MNC 57
  hlr_msisdn_country_code: "99",      # مثال على رمز دولة مكون من رقمين
  hlr_msisdn_nsn_offset: 2,           # تخطي رمز الدولة المكون من رقمين
  hlr_msisdn_nsn_length: 9            # استخراج NSN مكون من 9 أرقام
```

**مثال** (رمز دولة مكون من 3 أرقام):

```
config :omniss7,
  hlr_imsi_plmn_prefix: "50557",      # MCC 505 + MNC 57
  hlr_msisdn_country_code: "999",     # مثال على رمز دولة مكون من
3 أرقام
  hlr_msisdn_nsn_offset: 3,           # تخطي رمز الدولة المكون من
3 أرقام
  hlr_msisdn_nsn_length: 8            # استخراج NSN مكون من 8
أرقام
```

**مهم**: قم بتعيين `nsn_offset` إلى طول رمز دولتك لاستخراج NSN بشكل صحيح. على سبيل المثال:

- رمز الدولة "9" (رقم واحد) → `nsn_offset: 1`
- رمز الدولة "99" (رقمان) → `nsn_offset: 2`
- رمز الدولة "999" (3 أرقام) → `nsn_offset: 3`

## تكوين InsertSubscriberData (ISD)

للحصول على شرح UpdateLocation. أثناء VLRs تكوين بيانات توفير المشتركين المرسلة إلى HLR. في دليل InsertSubscriberData وتدفق الرسائل، راجع تكوين ISD مفصل لتسلسل

| المعلمة | النوع | الافتراضي | مطلوب | |
|---|---|---|---|---|
| `isd_network_access_mode` | Atom | `:packetAndCircuit` | لا | شبكة<br>`:pa`<br>`:pa`<br>`:ci` |
| `isd_send_ss_data` | Boolean | `true` | لا | رسال<br>ميلية |
| `isd_send_call_barring` | Boolean | `true` | لا | رسال<br>لمات |

**مثال**:

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

## تكوين CAMEL

تكوين لتوجيه المكالمات الذكية المعتمد على CAMEL. للحصول على شرح مفصل لتكامل CAMEL ومفاتيح الخدمة، راجع تكوين CAMEL في دليل HLR.

| المعلمة | النوع | الافتراضي | وب |
|---|---|---|---|
| `camel_service_key` | Integer | `11_110` | لا |
| `camel_trigger_detection_point` | Atom | `:termAttemptAuthorized` | لا |
| `camel_gsmscf_gt_address` | String | (المتصل GT يستخدم) | لا |

**مثال**:

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

## بادئات VLR المنزلية

تكوين لتمييز المشتركين المحليين عن المشتركين المتجولين. للحصول على شرح مفصل لاكتشاف المنزل/التجوال وعمليات PRN، راجع التعامل مع المشتركين المتجولين في دليل HLR.

| المعلمة | النوع | الافتراضي | مطلوب | الوصف |
|---|---|---|---|---|
| `home_vlr_prefixes` | List | `["5551231"]` | لا | بادئات GT لـ VLR التي تعتبر شبكة "منزلية" |

**مثال**:

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

---

# معلمات وضع SMSc

تكوين لوضع مركز الرسائل القصيرة.

## تكوين واجهة برمجة تطبيقات SMSc

| المعلمة | النوع | الافتراضي | مطلوب |
|---|---|---|---|
| `smsc_api_base_url` | String | - | **نعم** |
| `smsc_name` | String | `"{hostname}_SMSc"` | لا |
| `smsc_service_center_gt_address` | String | - | **نعم** |

**مثال**:

```
config :omniss7,
  smsc_api_base_url: "https://10.179.3.219:8443",
  smsc_name: "ipsmgw",
  smsc_service_center_gt_address: "55512341112"
```

**ملاحظة**: يحدث التسجيل في الواجهة الأمامية كل **5 دقائق** (ثابت) عبر وحدة `SMS.FrontendRegistry`.

## تكوين التفريغ التلقائي

| المعلمة | النوع | الافتراضي | مطلوب | الوصف |
|---|---|---|---|---|
| `auto_flush_enabled` | Boolean | `true` | لا | تمكين معالجة قائمة انتظار SMS تلقائيًا |
| `auto_flush_interval` | Integer | `10_000` | لا | فترة معالجة قائمة الانتظار بالمللي ثانية |
| `auto_flush_dest_smsc` | String | - | **نعم** | اسم SMSc الوجهة للتفريغ التلقائي |
| `auto_flush_tps` | Integer | `10` | لا | معدل معالجة الرسائل (معاملات/ثانية) |

**مثال**:

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

---

# معلمات وضع STP

تكوين لوضع نقطة نقل إشارة M3UA. للحصول على تكوين توجيه مفصل وأمثلة، راجع دليل تكوين STP.

## خادم STP مستقل

| المعلمة | النوع | الافتراضي | مطلوب | الوصف |
|---|---|---|---|---|
| `m3ua_stp.enabled` | Boolean | `false` | لا | تمكين خادم M3UA STP المستقل |
| `m3ua_stp.local_ip` | Tuple أو List | `{127, 0, 0, 1}` | لا | عنوان IP (عناوين) للاستماع للاتصالات. IP واحد: `{10, 0, 0, 1}` أو عدة IPs لتعدد الإرسال SCTP: `[{10, 0, 0, 1}, {10, 0, 0, 2}]` |
| `m3ua_stp.local_port` | Integer | `2905` | لا | المنفذ للاستماع عليه |
| `m3ua_stp.point_code` | Integer | - | **نعم** (إذا تم تمكينه) | رمز نقطة SS7 الخاص بـ STP هذا |

**مثال (IP واحد)**:

```
config :omniss7,
  m3ua_stp: %{
    enabled: true,
    local_ip: {10, 179, 4, 10},
    local_port: 2905,
    point_code: 100
  }
```

**مثال (تعدد الإرسال SCTP)**:

```
config :omniss7,
  m3ua_stp: %{
    enabled: true,
    # عدة IPs للموثوقية
    local_ip: [{10, 179, 4, 10}, {10, 179, 4, 11}],
    local_port: 2905,
    point_code: 100
  }
```

> **ملاحظة**: للحصول على معلومات مفصلة حول تكوين تعدد الإرسال SCTP والفوائد، راجع في الدليل الشائع SCTP تعدد الإرسال.

### توجيه العنوان العالمي

| المعلمة | النوع | الافتراضي | مطلوب | الوصف |
|---|---|---|---|---|
| `enable_gt_routing` | Boolean | `false` | لا | تمكين توجيه GT بالإضافة إلى توجيه PC |

**مثال**:

```
config :omniss7,
  enable_gt_routing: true
```

---

## معلمات NAT لعنوان العنوان العالمي

يسمح NAT لعنوان العنوان العالمي بوجود GTs استجابة مختلفة بناءً على بادئة الطرف المتصل. للحصول على شرح تفصيلي وأمثلة، راجع دليل NAT لعنوان العنوان العالمي.

| المعلمة | النوع | الافتراضي | مطلوب | الوصف |
|---|---|---|---|---|
| `gt_nat_enabled` | Boolean | `false` | لا | تمكين/تعطيل ميزة NAT لعنوان العنوان العالمي |
| `gt_nat_rules` | List of Maps | `[]` | **نعم** (إذا تم تمكينه) | قائمة من تحويلات GT بادئة إلى |

خريطة تحتوي على `gt_nat_rules` **تنسيق القاعدة**: يجب أن تكون كل قاعدة في:

- `calling_prefix`: بادئة سلسلة لمطابقة GT المتصل
- `response_gt`: العنوان العالمي المستخدم في الاستجابات

**مثال**:

```
config :omniss7,
  gt_nat_enabled: true,
  gt_nat_rules: [
    # يبدأ بـ "8772"، استجب بـ "55512341112" عند الاتصال من GT
    %{calling_prefix: "8772", response_gt: "55512341112"},
    # يبدأ بـ "8773"، استجب بـ "55512341111" عند الاتصال من GT
    %{calling_prefix: "8773", response_gt: "55512341111"},
    # الافتراضي الاحتياطي (بادئة فارغة تطابق الكل)
    %{calling_prefix: "", response_gt: "55512311555"}
  ]
```

**انظر أيضًا**: دليل NAT للاستخدام التفصيلي والأمثلة لعنوان العنوان العالمي.

---

# معلمات اتصال M3UA

تكوين اتصال M3UA لوضع عميل MAP. للحصول على الاستخدام التفصيلي والأمثلة، راجع دليل عميل MAP.

| المعلمة | النوع | الافتراضي | مطلوب | |
|---|---|---|---|---|
| `map_client_m3ua.mode` | String | - | **نعم** | لاتصال<br>أو `"SG` |
| `map_client_m3ua.callback` | Tuple | - | **نعم** | ستدعاء<br>`{MapC`<br>`:hand`<br>`[]}` |
| `map_client_m3ua.process_name` | Atom | - | **نعم** | نسجيل |
| `map_client_m3ua.local_ip` | Tuple<br>أو List | - | **نعم** | عناوين)<br>. واحد:<br>`1}` أو<br>لإرسال<br>SCTP:<br>`0, 1}`<br>`0, 2}` |
| `map_client_m3ua.local_port` | Integer | `2905` | **نعم** | S منفذ |
| `map_client_m3ua.remote_ip` | Tuple<br>أو List | - | **نعم** | عناوين)<br>STP/S<br>`,0 ,0`<br>`,10}]`<br>`,10}`<br>`[{11` |
| `map_client_m3ua.remote_port` | Integer | `2905` | **نعم** | S منفذ |
| `map_client_m3ua.routing_context` | Integer | - | **نعم** | ي توجيه<br>M3UA |

**مثال (واحد IP)**:

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

**مثال (تعدد الإرسال SCTP)**:

```
config :omniss7,
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :hlr_client_asp,
    # عدة IPs محلية للموثوقية
    local_ip: [{10, 179, 4, 11}, {10, 179, 4, 12}],
    local_port: 2905,
    # عدة IPs بعيدة للموثوقية STP
    remote_ip: [{10, 179, 4, 10}, {10, 179, 4, 20}],
    remote_port: 2905,
    routing_context: 1
  }
```

> **ملاحظة**: للحصول على معلومات مفصلة حول تكوين تعدد الإرسال SCTP والفوائد، راجع الدليل الشائع SCTP تعدد الإرسال في.

---

# معلمات خادم HTTP

تكوين لخادم HTTP لواجهة برمجة التطبيقات REST.

| المعلمة | النوع | الافتراضي | مطلوب | الوصف |
|---|---|---|---|---|
| `start_http_server` | Boolean | `true` | لا | تمكين/تعطيل خادم HTTP (المنفذ 8080) |

**القيم الثابتة** (غير قابلة للتكوين):

- **IP**: 0.0.0.0 (جميع الواجهات)
- **المنفذ**: 8080
- **النقل**: Plug.Cowboy

**مثال**:

```
config :omniss7,
  start_http_server: true  # تعيين إلى false لتعطيل
```

**نقاط نهاية واجهة برمجة التطبيقات**:

- واجهة برمجة التطبيقات REST: `http://[server-ip]:8080/api/*`
- واجهة Swagger: `http://[server-ip]:8080/swagger`
- مقاييس Prometheus: `http://[server-ip]:8080/metrics`

راجع دليل واجهة برمجة التطبيقات للحصول على التفاصيل.

---

# معلمات قاعدة البيانات

تكوين لاستمرارية قاعدة بيانات Mnesia.

| المعلمة | النوع | الافتراضي | مطلوب | الوصف |
|---|---|---|---|---|
| `mnesia_storage_type` | Atom | `:disc_copies` | لا | نوع تخزين Mnesia: `:disc_copies` أو `:ram_copies` |

**مثال**:

```
config :omniss7,
  mnesia_storage_type: :disc_copies  # الإنتاج
  # mnesia_storage_type: :ram_copies  # للاختبار فقط
```

**أنواع التخزين**:

- `:disc_copies` - **موصى به للإنتاج** - (يستمر بعد إعادة التشغيل) تخزين قرص دائم
- `:ram_copies` - للاختبار فقط - (يفقد عند إعادة التشغيل) في الذاكرة فقط

**جداول Mnesia**:

- `m3ua_peer` - اتصالات نظير M3UA
- `m3ua_route` - طرق رمز النقطة
- `m3ua_gt_route` - طرق العنوان العالمي

**الموقع**: دليل `Mnesia.{node_name}/`

---

# القيم الثابتة

القيم التالية **مثبتة في الشيفرة المصدرية** ولا يمكن تغييرها عبر التكوين.

## مهلات

| الحل | التأثير | القيمة |
|---|---|---|
| تعديل الشيفرة المصدرية | تنتهي بعد 10 MAP جميع عمليات ثواني | مهلة طلب MAP: **10 ثواني** |
| تعديل الشيفرة المصدرية | تنتهي بعد 10 ثواني ISD كل رسالة | مهلة ISD: **10 ثواني** |

## خادم HTTP

| الحل | التأثير | القيمة |
|---|---|---|
| تعديل الشيفرة المصدرية | الخادم يستمع على جميع الواجهات | HTTP IP: **0.0.0.0** |
| تعديل الشيفرة المصدرية | تعمل على REST واجهة برمجة التطبيقات المنفذ 8080 | HTTP Port: **8080** |

## التحقق من SSL

| الحل | التأثير | القيمة |
|---|---|---|
| تعديل الشيفرة المصدرية | معطل SSL التحقق من دائمًا | واجهة برمجة تطبيقات HLR SSL: **معطلة** |
| تعديل الشيفرة المصدرية | معطل SSL التحقق من دائمًا | واجهة برمجة تطبيقات SMSc SSL: **معطلة** |

## فترات التسجيل

| الحل | التأثير | القيمة |
| --- | --- | --- |
| تعديل الشيفرة المصدرية | يسجل مع الخلفية كل 5 SMSc دقائق | التسجيل في الواجهة الأمامية: **5 دقائق** |

## تحديث تلقائي لواجهة الويب

| الفترة | الصفحة |
| --- | --- |
| 5 ثواني | إدارة التوجيه |
| 2 ثواني | المشتركين النشطين |

---

# أمثلة التكوين

## تكوين HLR الحد الأدنى

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

## تكوين SMSc الحد الأدنى

```
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
  }
```

## مع خادم مستقل STP

```
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
  }
```

---

# الملخص

## إجمالي معلمات التكوين: 32

**حسب الفئة**:

- وضع التشغيل: 3 معلمات
- وضع HLR: 13 معلمة
- وضع SMSc: 7 معلمات

- وضع STP: 5 معلمات
- اتصال M3UA: 8 معلمات
- خادم HTTP: 1 معلمة
- قاعدة البيانات: 1 معلمة

(يجب تعيينها) **المعلمات المطلوبة**:

- `hlr_api_base_url` (وضع HLR)
- `hlr_service_center_gt_address` (وضع HLR)
- `smsc_api_base_url` (وضع SMSc)
- `smsc_service_center_gt_address` (وضع SMSc/HLR)
- جميع معلمات `map_client_m3ua.*`
- `m3ua_stp.point_code` (إذا تم تمكين STP)

---

# الوثائق ذات الصلة

- **دليل HLR** - تكوين خاص بـ HLR
- **دليل SMSc** - تكوين خاص بـ SMSc
- **دليل STP** - تكوين توجيه STP
- **دليل واجهة برمجة التطبيقات** REST - مرجع واجهة برمجة التطبيقات
- **دليل واجهة الويب** - وثائق الواجهة الويب

# دليل ترجمة عنوان الشبكة العالمية (NAT)

## نظرة عامة

ترجمة عنوان الشبكة العالمية (GT NAT) هي ميزة تسمح لـ OmniSS7 بالاستجابة بعناوين GT للجهة المتصلة بها أو مزيج من كلاهما. GT للجهة المتصلة أو بادئة GT عالمية مختلفة بناءً على بادئة GT الصحيح بناءً على الشبكة أو النظير الذي يتصل و/أو أي GT تم الاتصال به. هذه الميزة ضرورية عند العمل مع عدة عناوين عالمية وتحتاج إلى ضمان أن تستخدم الاستجابات

## ما الجديد (GT NAT المحسن)

تم تحسين ميزة GT NAT بقدرات جديدة قوية:

### الميزات الجديدة

1. **مطابقة بادئة الجهة المتصلة**: يمكن الآن للقواعد المطابقة على `called_prefix` بالإضافة إلى `calling_prefix`
2. **المطابقة المجمعة**: يمكن للقواعد المطابقة على كلا ال��ادئين المتصلين والمستقبلين في نفس الوقت
3. **أولوية قائمة على الوزن**: تستخدم القواعد الآن حقل `weight` (أقل = أولوية) بدلاً من مجرد طول البادئة (أعلى
4. **مطابقة مرنة**: يمكنك الآن إنشاء قواعد مع:
   - بادئة متصلة فقط
   - بادئة متصلة بها فقط
   - كلا البادئين المتصلين والمستقبلين
   - لا شيء (قاعدة بديلة/عامة)

### تنسيق القاعدة الجديدة

**الحقول المطلوبة:**

- `weight`: أولوية عددية (أقل = أولوية أعلى)
- `response_gt`: GT للاستجابة به

**الحقول الاختيارية (يوصى بوجود واحد على الأقل للمطابقة المحددة):**

- `calling_prefix`: المطابقة على بادئة GT للجهة المتصلة
- `called_prefix`: المطابقة على بادئة GT للجهة المتصلة بها

**مثال:**

```
gt_nat_rules: [
  # قاعدة محددة مع كلا البادئين - أعلى أولوية
  %{calling_prefix: "8772", called_prefix: "555", weight: 1,
response_gt: "111111"},

  # قواعد محددة - أولوية متوسطة
  %{calling_prefix: "8772", weight: 10, response_gt: "222222"},
  %{called_prefix: "555", weight: 10, response_gt: "333333"},

  # بديل عام - أقل أولوية
  %{weight: 100, response_gt: "999999"}
]
```

# حالات الاستخدام

## التشغيل عبر شبكات متعددة

عندما يكون لديك عدة شبكات نظيرة وكل منها يتوقع استجابات من GT محدد:

- **الشبكة A** تتصل بـ GT `111111` وتتوقع استجابات من `111111` الخاص بك
- **الشبكة B** تتصل بـ GT `222222` وتتوقع استجابات من `222222` الخاص بك

يمكن لمثيل واحد من GT NAT، ستحتاج إلى مثيلات منفصلة أو توجيه معقد. مع GT NAT، بدون التعامل مع هذا بذكاء OmniSS7.

## سيناريوهات التجوال

مع اتفاقيات تجوال SMSc أو HLR عند العمل كـ:

- مشتركو **الشبكة المنزلية** يستخدمون GT `555000`
- **الشريك المتجول 1** يستخدم GT `555001`
- **الشريك المتجول 2** يستخدم GT `555002`

يضمن GT NAT الصحيح الذي تم تكوينه للتوجيه إليه GT أن يتلقى كل شريك استجابات من.

## الاختبار والترحيل

أثناء ترحيل الشبكات أو الاختبار:

- ترحيل حركة المرور تدريجياً من GT القديم إلى GT الجديد
- الحفاظ على كلا GTs خلال فترة الانتقال
- توجيه الاستجابات بناءً على GT الذي استخدمه المتصل

# كيف يعمل

## تدفق ترجمة العنوان

1. **طلب وارد**: يتلقى OmniSS7 رسالة SCCP مع:
   - GT `55512341112` الجهة المتصلة: (GT الخاص بك)
   - GT `877234567` الجهة المتصلة بها: (GT الخاص بهم)
2. **بحث GT NAT**: يتحقق النظام من GT `877234567` المتصل مقابل قواعد البادئة المكونة
3. **مطابقة البادئة**: يجد أطول بادئة مطابقة (على سبيل المثال، `8772` تطابق `877234567`)
4. **اختيار GT للاستجابة**: يستخدم `response_gt` من القاعدة المطابقة (على سبيل المثال، `55512341112`)
5. **إرسال الاستجابة**: تستخدم استجابة SCCP:

- GT `877234567` :الجهة المتصلة (معكوس - GT الخاص بهم)
- GT `55512341112` :الجهة المتصلة بها (GT المعاد توجيهه)

## أنواع الاستجابة المتأثرة

يطبق GT NAT على عدة طبقات من كومة SS7:

**طبقة SCCP (جميع الاست��ابات)**

- عناوين GT للجهة المتصلة/المتصل بها في جميع رسائل الاستجابة
- تأكيدات ISD (InsertSubscriberData)
- استجابات UpdateLocation
- استجابات الأخطاء

**طبقة MAP (محددة بالعملية)**

- **استجابات SRI-for-SM**: `networkNode-Number` (عنوان GT لـ SMSc)
- **UpdateLocation**: `hlr-Number` في الاستجابات
- **InsertSubscriberData**: GT HLR في رسائل ISD

# التكوين

## التكوين الأساسي

أضف إلى `config/runtime.exs`:

```
config :omniss7,
  # تفعيل GT NAT
  gt_nat_enabled: true,

  # تعريف قواعد GT NAT
  gt_nat_rules: [
    # القاعدة 1: المكالمات من البادئة "8772" تحصل على استجابة من
55512341112""
    %{calling_prefix: "8772", response_gt: "55512341112"},

    # القاعدة 2: المكالمات من البادئة "8773" تحصل على استجابة من
55512341111""
    %{calling_prefix: "8773", response_gt: "55512341111"},

    # القاعدة الافتراضية (بادئة فارغة تطابق كل شيء)
    %{calling_prefix: "", response_gt: "55512311555"}
  ]
```

## معلمات التكوين

للحصول على مرجع تكوين كامل، راجع معلما في مرجع التكوين GT NAT.

| المعلمة | النوع | مطلوب | الوصف |
|---|---|---|---|
| `gt_nat_enabled` | Boolean | نعم | تفعيل/تعطيل ميزة GT NAT |
| `gt_nat_rules` | قائمة من الخرائط | نعم (إذا تم التفعيل) | قائمة بقواعد مطابقة البادئة |

## تنسيق القاعدة

كل قاعدة هي خريطة تحتوي على المفاتيح التالية:

```
%{
  calling_prefix: "8772",      # (اختياري) بادئة للمطابقة مع GT المتصل
  called_prefix: "555",        # (اختياري) بادئة للمطابقة مع GT المتصل به
  weight: 10,                  # (مطلوب) قيمة الأولوية (أقل = أولوية أعلى)
  response_gt: "55512341112"   # (مطلوب) GT للاستخدام في الاستجابات
}
```

**حقول القاعدة:**

- **`calling_prefix`** (اختياري): بادئة سلسلة للمطابقة مع GT المتصل الوارد
  - تتم المطابقة بواسطة `String.starts_with?/2`
  - أو `""` السلسلة الفارغة `nil` تعمل كبديل (متصل GT تطابق أي)
  - يمكن حذفها لمطابقة أي متصل GT
- **`called_prefix`** (اختياري): بادئة سلسلة للمطابقة مع GT المتصل به الوارد
  - تتم المطابقة بواسطة `String.starts_with?/2`
  - أو `""` السلسلة الفارغة `nil` تعمل كبديل (متصل به GT تطابق أي)
  - يمكن حذفها لمطابقة أي متصل به GT
- **`weight`** (مطلوب): قيمة أولوية عددية
  - وزن أقل = أولوية أعلى (يتم معالجتها أولاً)
  - يجب أن تكون 0 =<
  - تستخدم كمعيار فرز أساسي لقواعد المطابقة
- **`response_gt`** (مطلوب): عنوان GT العالمي للاستخدام في الاستجابات
  - يجب أن تكون سلسلة رقم صالحة E.164
  - يجب أن تتطابق مع أحد GTs المكونة لديك

**يجب تحديد واحد على الأقل من `called_prefix` أو `calling_prefix` للمطابقة المحددة. يمكن حذف كلاهما لقاعدة بديلة/عامة.**

# منطق مطابقة القاعدة

يتم تقييم القواعد بواسطة **الوزن أولاً (تصاعدي)، ثم بواسطة خصوصية البادئة المجمعة**:

**خوارزمية المطابقة:**

1. تصفية القواعد حيث تتطابق جميع البادئات المحددة
   - إذا تم تعيين `calling_prefix`، يجب أن تتطابق مع GT المتصل
   - إذا تم تعيين `called_prefix`، يجب أن تتطابق مع GT المتصل به
   - إذا تم تعيين كلاهما، يجب أن تتطابق كلاهما
   - إذا لم يتم تعيين أي منهما، تعمل القاعدة كبديل
2. فرز القواعد المطابقة بواسطة:
   - **أساسي**: الوزن (تصاعدي - القيم الأقل أولاً)
   - **ثانوي**: طول البادئة المجمعة (تنازلي - الأطول = الأكثر تحديدًا)
3. إرجاع أول قاعدة مطابقة

**أمثلة:**

```
# أمثلة القواعد
gt_nat_rules: [
  # الوزن 1: أعلى أولوية - تطابق كلا البادئين
  %{calling_prefix: "8772", called_prefix: "555", weight: 1,
response_gt: "111111"},

  # الوزن 10: أولوية متوسطة - قواعد محددة
  %{calling_prefix: "8772", weight: 10, response_gt: "222222"},  #
المتصل فقط
  %{called_prefix: "555", weight: 10, response_gt: "333333"},    #
المتصل به فقط

  # الوزن 100: أقل أولوية - بديل عام
  %{weight: 100, response_gt: "444444"}  # تطابق كل شيء
]

# أمثلة المطابقة:
# المتصل: "877234567"، المتصل به: "555123" -> "111111" (الوزن 1،
كلاهما يتطابق)
# المتصل: "877234567"، المتصل به: "999999" -> "222222" (الوزن 10،
المتصل فقط)
# المتصل: "999999999"، المتصل به: "555123" -> "333333" (الوزن 10،
المتصل به فقط)
# المتصل: "999999999"، المتصل به: "888888" -> "444444" (الوزن 100،
بديل)
```

# أمثلة

## المثال 1: شريكان في الشبكة

**السيناريو**: أنت تدير SMSc مع شريكين في الشبكة. كل منهما يتوقع استجابات من GT مختلف.

```
config :omniss7,
  gt_nat_enabled: true,

  # GT الافتراضي لـ SMSc (يستخدم عندما يكون GT NAT معطلاً أو لا تطابق أي قاعدة)
  smsc_service_center_gt_address: "5551000",

  # قواعد GT NAT للشركاء
  gt_nat_rules: [
    # الشريك A (بادئة 4412) يتوقع استجابات من GT 5551001
    %{calling_prefix: "4412", weight: 10, response_gt: "5551001"},

    # الشريك B (بادئة 4413) يتوقع استجابات من GT 5551002
    %{calling_prefix: "4413", weight: 10, response_gt: "5551002"},

    # استخدام: GT الافتراضي لـ SMSc القياسي (بديل عام)
    %{weight: 100, response_gt: "5551000"}
  ]
```

**تدفق الحركة:**

```
طلب وارد SRI-for-SM 44121234567 من:
  GT 5551001 :المتصل به (GT الخاص بك الذي يستخدمه الشريك A)
  GT 44121234567 :المتصل (GT الخاص بالشريك A)

بحث GT NAT:
  "44121234567" 4412" تطابق بادئة"
  GT 5551001" :للاستجابة المحددة"

استجابة SRI-for-SM 44121234567 إلى:
  GT (معكوس) 44121234567 :المتصل
  GT (معاد توجيهه) 5551001 :المتصل
  networkNode-Number: 5551001 (في استجابة MAP)
```

# المثال 2: HLR مع GTs إقليمية

**السيناريو**: HLR وطني مع GTs مختلفة لكل منطقة.

```
config :omniss7,
  gt_nat_enabled: true,
  hlr_service_center_gt_address: "555000",  # GT الافتراضي لـ HLR

  gt_nat_rules: [
    # VLRs (5551 بادئة) في المنطقة الشمالية
    %{calling_prefix: "5551", weight: 10, response_gt: "555100"},

    # VLRs (5552 بادئة) في المنطقة الجنوبية
    %{calling_prefix: "5552", weight: 10, response_gt: "555200"},

    # VLRs (5553 بادئة) في المنطقة الغربية
    %{calling_prefix: "5553", weight: 10, response_gt: "555300"},

    # (بديل عام) الافتراضي للمناطق الأخرى
    %{weight: 100, response_gt: "555000"}
  ]
```

## المثال 3: سيناريو الترحيل

**السيناريو**: الترحيل من GT القديم إلى GT الجديد تدريجياً.

```
config :omniss7,
  gt_nat_enabled: true,
  hlr_service_center_gt_address: "123456789",  # GT القديم
(افتراضي)

  gt_nat_rules: [
    # (قامت بتحديث تكويناتها بالفعل) الشبكات المهاجرة
    %{calling_prefix: "555", weight: 10, response_gt:
"987654321"},  # GT الجديد
    %{calling_prefix: "666", weight: 10, response_gt:
"987654321"},  # GT الجديد

    # (بديل عام) GT القديم لا يزال يستخدم الجميع
    %{weight: 100, response_gt: "123456789"}  # GT القديم
  ]
```

## المثال 4: مطابقة بادئة الجهة المتصلة (جديد)

**السيناريو**: لديك عدة GTs لخدمات مختلفة، وتريد الاستجابة بـ GT الصحيح بناءً على GT الذي تم الاتصال به.

```
config :omniss7,
  gt_nat_enabled: true,

  gt_nat_rules: [
    # الرسائل القصيرة الخاص بك GT عندما يتصلون بـ (5551xxx)، استجب بتلك GT
    %{called_prefix: "5551", weight: 10, response_gt: "555100"},

    # الصوت الخاص بك GT عندما يتصلون بـ (5552xxx)، استجب بتلك GT
    %{called_prefix: "5552", weight: 10, response_gt: "555200"},

    # البيانات الخاص بك GT عندما يتصلون بـ (5553xxx)، استجب بتلك GT
    %{called_prefix: "5553", weight: 10, response_gt: "555300"},

    # بديل عام
    %{weight: 100, response_gt: "555000"}
  ]
```

**تدفق الحركة:**

```
طلب وارد إلى GT 555100 :المتصل به (الرسائل القصيرة الخاص بك GT)
GT (أي متصل) 441234567 :المتصل

بحث GT NAT:
  GT 5551" تطابق بادئة "555100" :المتصل به"
  GT 555100" :للاستجابة المحددة"

المتصل: 555100 (تطابق ما اتصلوا به) GT تستخدم الاستجابة
```

## المثال 5: المطابقة المجمعة للبادئات المتصلة والمستقبلين (متقدم)

**السيناريو**: يتصل شركاء مختلفون بـ GTs مختلفة، وتريد تحكمًا دقيقًا.

```
config :omniss7,
  gt_nat_enabled: true,

  gt_nat_rules: [
    # الرسائل القصيرة الخاصة بك - أعلى أولوية GT يتصل بـ A الشريك
(الوزن 1)
    %{calling_prefix: "4412", called_prefix: "5551", weight: 1,
response_gt: "555101"},

    # الرسائل القصيرة الخاصة بك - أعلى أولوية GT يتصل بـ B الشريك
(الوزن 1)
    %{calling_prefix: "4413", called_prefix: "5551", weight: 1,
response_gt: "555102"},

    # الرسائل القصيرة الخاصة بك - أولوية متوسطة GT أي شخص يتصل بـ
(الوزن 10)
    %{called_prefix: "5551", weight: 10, response_gt: "555100"},

    # أولوية متوسطة (الوزن 10) - GT يتصل بأي A الشريك
    %{calling_prefix: "4412", weight: 10, response_gt: "555200"},

    # بديل عام - أولوية منخفضة (الوزن 100)
    %{weight: 100, response_gt: "555000"}
  ]
```

**أمثلة المطابقة:**

```
# الرسائل القصيرة GT يتصل بـ A الشريك
"المتصل: "441234567"، المتصل به: "555100
"تطابق القاعدة ذات الوزن 1 (كلا البادئين) → 555101" →

# الصوت GT يتصل بـ A الشريك
"المتصل: "441234567"، المتصل به: "555200
"تطابق القاعدة ذات الوزن 10 (المتصل فقط) → 555200" →

# الرسائل القصيرة GT متصل غير معروف يتصل بـ
"المتصل: "999999999"، المتصل به: "555100
"تطابق القاعدة ذات الوزن 10 (المتصل به فقط) → 555100" →

# الصوت GT متصل غير معروف يتصل بـ
"المتصل: "999999999"، المتصل به: "555200
"تطابق القاعدة ذات الوزن 100 (بديل) → 555000" →
```

# أوضاع التشغيل

يعمل GT NAT عبر جميع أوضاع التشغيل لـ OmniSS7:

## وضع HLR

يؤثر GT NAT على:

- استجابات UpdateLocation (GT HLR في الاستجابة)
- رسائل InsertSubscriberData (GT HLR كجهة متصلة)
- استجابات SendAuthenticationInfo
- استجابات Cancel Location

لمزيد من المعلومات حول عمليات HLR، راجع دليل تكوين HLR.

**التكوين:**

```
config :omniss7,
  hlr_mode_enabled: true,
  hlr_service_center_gt_address: "5551234567",  # GT الافتراضي لـ
HLR

  gt_nat_enabled: true,
  gt_nat_rules: [
    %{calling_prefix: "331", weight: 10, response_gt:
"5551234568"},  # فرنسا
    %{calling_prefix: "44", weight: 10, response_gt:
"5551234569"},   # المملكة المتحدة
    %{weight: 100, response_gt: "5551234567"}  # بديل عام
  ]
```

## وضع SMSc

يؤثر GT NAT على:

- استجابات SRI-for-SM (الحقل `networkNode-Number`) - انظر تفاصيل SRI-for-SM
- تأكيدات MT-ForwardSM

لمزيد من المعلومات حول عمليات SMSc، راجع دليل تكوين SMSc.

**التكوين:**

```
config :omniss7,
  smsc_mode_enabled: true,
  smsc_service_center_gt_address: "5559999",  # GT الافتراضي لـ
SMSc

  gt_nat_enabled: true,
  gt_nat_rules: [
    %{calling_prefix: "1", weight: 10, response_gt: "5559991"},
# أمريكا الشمالية
    %{calling_prefix: "44", weight: 10, response_gt: "5559992"},
# المملكة المتحدة
    %{calling_prefix: "86", weight: 10, response_gt: "5559993"},
# الصين
    %{weight: 100, response_gt: "5559999"}  # بديل عام
  ]
```

## وضع بوابة CAMEL

يؤثر GT NAT على:

- جميع استجابات مستوى SCCP (GT gsmSCF كجهة متصلة)
- استجابات عمليات CAMEL/CAP (InitialDP، EventReportBCSM، إلخ.)
- تأكيدات RequestReportBCSMEvent
- استجابات ApplyCharging
- استجابات Continue

**التكوين:**

```
config :omniss7,
  camelgw_mode_enabled: true,
  camel_gsmscf_gt_address: "55512341112",  # GT الافتراضي لـ gsmSCF

  gt_nat_enabled: true,
  gt_nat_rules: [
    %{calling_prefix: "555", weight: 10, response_gt:
"55512341111"},  # الشبكة A
    %{calling_prefix: "666", weight: 10, response_gt:
"55512311555"},  # الشبكة B
    %{weight: 100, response_gt: "55512341112"}  # بديل عام
  ]
```

**حالة الاستخدام:** عند العمل كـ gsmSCF (وظيفة التحكم في الخدمة) لعدة شبكات، قد يتوقع كل gsmSSF استجابات من GT gsmSCF محدد. يضمن GT NAT استخدام GT الصحيح بناءً على أي GT يتصل gsmSSF.

# السجلات واستكشاف الأخطاء وإصلاحها

## تفعيل سجلات GT NAT

يتضمن GT NAT تسجيلًا تلقائيًا لجميع الترجمات:

```
# في السجلات، سترى:
[info] GT NAT [استجابة SRI-for-SM]: GT 877234567 المتصل -> GT
55512341112 الاستجابة
[info] GT NAT [UpdateLocation ISD]: GT 331234567 المتصل -> GT
55512341111 الاستجابة
[info] GT NAT [استجابة MAP BEGIN]: GT 441234567 المتصل -> GT
55512311555 الاستجابة
```

يظهر حقل السياق المكان الذي تم فيه تطبيق NAT:

- `"استجابة SRI-for-SM"` - في معالج SRI-for-SM
- `"UpdateLocation ISD"` - في رسائل InsertSubscriberData
- `"UpdateLocation END"` - في استجابة UpdateLocation END
- `"استجابة MAP BEGIN"` - استجابات MAP BEGIN العامة
- `"ISD ACK"` - تأكيد ISD
- `"استجابة خطأ HLR"` - استجابة خطأ من HLR
- `"استجابة CAMEL"` - استجابات عمليات CAMEL/CAP (gsmSCF)

# التحقق

يتحقق النظام من تكوين GT NAT عند بدء التشغيل:

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

# تحقق مما إذا كان مفعلًا
iex> GtNat.enabled?()
true

# احصل على جميع القواعد
iex> GtNat.get_rules()
[
  %{calling_prefix: "8772", weight: 10, response_gt:
"55512341112"},
  %{calling_prefix: "8773", weight: 10, response_gt:
"55512341111"}
]
```

## اختبار GT NAT

اختبر منطق GT NAT برمجيًا:

```
# اختبار الترجمة مع GT المتصل فقط (called_gt هو nil)
iex> GtNat.translate_response_gt("877234567", nil, "default_gt")
"55512341112"

# اختبار الترجمة مع كل من GT المتصل و GTالمتصل به
iex> GtNat.translate_response_gt("877234567", "555123",
"default_gt")
"55512341112"

# اختبار مع السجلات (GT المتصل به هو nil)
iex> GtNat.translate_response_gt_with_logging("877234567", nil,
"default_gt", "test")
# السجلات: GT NAT [test]: GT 877234567 المتصل -> GT الاستجابة
55512341112
"55512341112"

# اختبار مع السجلات (كلا GTs)
iex> GtNat.translate_response_gt_with_logging("877234567",
"555123", "default_gt", "test")
# السجلات: GT NAT [test]: GT 877234567 المتصل، GT 555123 المتصل به
-> GT 55512341112 الاستجابة
"55512341112"

# اختبار عدم المطابقة (يعود إلى الافتراضي)
iex> GtNat.translate_response_gt("999999999", "888888",
"default_gt")
"default_gt"
```

# استكشاف الأخطاء وإصلاحها

## المشكلة: GT NAT لا يعمل

### التحقق 1: هل هو مفعل؟

```
iex> Application.get_env(:omniss7, :gt_nat_enabled)
true  # يجب أن يكون true
```

### التحقق 2: هل تم تكوين القواعد؟

```
iex> Application.get_env(:omniss7, :gt_nat_rules)
[%{calling_prefix: "8772", response_gt: "55512341112"}, ...]  # يجب أن تعيد قائمة
```

في السجلات لمعرفة ما إذا كانت "GT NAT" ابحث عن **التحقق 3: تحقق من السجلات** الترجمات تحدث.

## المشكلة: GT خاطئ في الاستج��بات

**الأعراض**: تستخدم الاستجابات عنوان GT غير المتوقع

**السبب**: قد تكون مطابقة بادئة القاعدة واسعة جدًا أو أن القاعدة الافتراضية تلتقط الحركة

**الحل**: مراجعة أوزان القواعد والبادئات:

```
# سيء: بديل مع وزن منخفض (يلتقط كل شيء أولاً)
gt_nat_rules: [
  %{weight: 1, response_gt: "111111"},              # هذا يطابق كل شيء أولاً!
  %{calling_prefix: "8772", weight: 10, response_gt: "222222"}  # لا تصل أبدًا
]

# جيد: قواعد محددة مع وزن أقل، بديل مع وزن أعلى
gt_nat_rules: [
  %{calling_prefix: "8772", weight: 10, response_gt: "222222"},  # محدد، وزن منخفض
  %{weight: 100, response_gt: "111111"}  # بديل، وزن عالي (بديل)
]
```

## المشكلة: لم يتم تطبيق GT NAT على نوع رسالة محدد

**الأعراض**: تستخدم بعض الاستجابات GT المعاد توجيهه، والبعض الآخر لا

**التغطية الحالية:**

- ✅ SCCP (جميع الاستجابات) المتصل GT
- ✅ SRI-for-SM (networkNode-Number) استجابات

- □ رسائل UpdateLocation ISD (GT HLR)
- □ استجابات UpdateLocation END
- □ تأكيدات ISD
- □ استجابات MAP BEGIN

إذا لم يكن نوع رسالة محدد يستخدم GT NAT، فقد لا يتم تنفيذه بعد. تحقق من الشيفرة المصدرية أو اتصل بالدعم.

# اعتبارات الأداء

## أداء البحث

يستخدم GT NAT مطابقة بادئة بسيطة مع تعقيد O(n) حيث n هو عدد القواعد.

**نصائح الأداء:**

- احتفظ بعدد القواعد أقل من 100 لأفضل أداء
- استخدم بادئات محددة لتقليل عدد القواعد
- يجب أن تكون القاعدة الافتراضية (بادئة فارغة) في النهاية

**اختبار الأداء (نظام نموذجي):**

- 10 1 > :قواعدµs لكل بحث
- 50 5 > :قاعدةµs لكل بحث
- 100 10 > :قاعدةµs لكل بحث

## استخدام الذاكرة

تتطلب كل قاعدة حوالي 100 بايت من الذاكرة:

- 10 1 ≈ قواعد KB
- 100 10 ≈ قاعدة KB

# أفضل الممارسات

## 1. دائمًا تضمين قاعدة بديلة عامة

```
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"}  # دائمًا يجب أن يكون لديك بديل بوزن عالي
]
```

## 2. استخدم بادئات وأوزان ذات معنى

```
# جيد: بادئات واضحة ومحددة مع أوزان مناسبة
%{calling_prefix: "331", weight: 10, response_gt: "..."}  # فرنسا
%{calling_prefix: "44", weight: 10, response_gt: "..."}   # المملكة المتحدة

# سيء: بادئات واسعة جدًا أو أوزان مربكة
%{calling_prefix: "3", weight: 5, response_gt: "..."}     # العديد من البلدان
%{calling_prefix: "331", weight: 100, response_gt: "..."}  # الوزن يجب أن يكون أقل للقواعد المحددة
```

## 3. وثق قواعدك

```
gt_nat_rules: [
  # شبكة المملكة المتحدة (نطاق GT: 4412xxxxxxx) - الشريك XYZ
  # الوزن 10: أولوية الشريك القياسية
  %{calling_prefix: "4412", weight: 10, response_gt: "5551001"},

  # شبكة فرنسا (نطاق GT: 33123xxxxxx) - الشريك ABC
  # الوزن 10: أولوية الشريك القياسية
  %{calling_prefix: "33123", weight: 10, response_gt: "5551002"}
]
```

## 4. اختبر قبل النشر

```
# اختبر في iex قبل النشر
iex> GtNat.translate_response_gt("44121234567", nil, "default")
"5551001"  # النتيجة المتوقعة

# اختبار مع GT المتصل به
iex> GtNat.translate_response_gt("44121234567", "555123",
"default")
"5551001"  # النتيجة المتوقعة
```

## 5. راقب السجلات

قم بتمكين تسجيل مستوى INFO لرؤية جميع ترجمات GT NAT في الإنتاج.

# التكامل مع ميزات أخرى

## وضع STP

يعمل GT NAT بشكل مستقل عن توجيه STP. تقوم STP بتوجيه بناءً على رموز النقاط وعناوين GT الوجهة، بينما يتعامل GT NAT مع عناوين الاستجابة.

لمزيد من المعلومات حول توجيه STP، راجع دليل تكوين STP.

## تكامل CAMEL

GT NAT **مكامل تمامًا** مع عمليات CAMEL/CAP:

**طبقة SCCP:**

- GT الجهة المتصلة في جميع استجابات CAMEL
- GT gsmSSF الوارد يتم تطبيقه تلقائيًا بناءً على

**التكوين:**

- `camel_gsmscf_gt_address` - GT الافتراضي لـ gsmSCF (اختياري)
- الجهة المتصلة من الطلب الوارد GT إذا لم يتم تكوينه، يستخدم

- الافتراضي بناءً على بادئة الجهة ��لمتصلة GT NAT تتجاوز قواعد

**مثال:**

```
# عندما يتصل gsmSSF 555123456 بـ gsmSCF الخاص بك
# الوارد: المتصل به=55512341112، المتصل=555123456
# بحث GT NAT: "555" -> GT 55512341111"=الاستجابة"
# الاستجابة: المتصل به=555123456، المتصل=55512341111
```

## توازن الحمل

يمكن دمج GT NAT مع توازن الحمل M3UA لإدارة حركة المرور المتقدمة.

# دليل الترحيل

## تفعيل GT NAT على النظام الحالي

1. **تحضير التكوين**

```
# أضف إلى runtime.exs (احتفظ به معطل في البداية)
config :omniss7,
  gt_nat_enabled: false,  # ابدأ معطلاً
  gt_nat_rules: [
    # قواعدك هنا مع الأوزان
    %{calling_prefix: "877", weight: 10, response_gt:
"111111"},
    %{weight: 100, response_gt: "999999"}  # بديل عام
  ]
```

2. **اختبار التكوين**

```
# تحقق من تجميع التكوين
mix compile

# اختبار في iex
iex -S mix
iex> GtNat.validate_config()
```

3. **تفعيل في البيئة التجريبية**

```
gt_nat_enabled: true  # تغيير إلى true
```

4. **راقب السجلات**

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

5. **نشر في الإنتاج**

    - نشر خلال نافذة الصيانة
    - راقب الساعات الـ 24 الأولى عن كثب
    - يجب أن يكون لديك خطة للعودة (تعيين `gt_nat_enabled: false`)

# الدعم

للمشاكل أو الأسئلة:

- تحقق من السجلات للحصول على رسائل "GT NAT"
- تحقق من التكوين باستخدام `GtNat.validate_config()`
- راجع قسم استكشاف الأخطاء وإصلاحها في هذا الدليل
- اتصل بدعم OmniSS7 مع مقتطفات السجلات

# دليل تكوين HLR

← العودة إلى الوثائق الرئيسية

يوفر هذا الدليل تكوينًا لاستخدام OmniSS7 مع **سجل الموقع المنزلي (HLR/HSS)** كـ **OmniHSS** كقاعدة بيانات المشتركين الخلفية.

# تكامل OmniHSS

**OmniHSS**، وهو خادم **SS7 كواجهة إشارة OmniSS7 في HLR يعمل وضع** تتفاعل مع مشترك منزلي كامل الميزات (HSS) في الخلفية. تفصل هذه البنية بين الاهتمامات:

- **OmniSS7 (واجهة HLR)**: تتعامل مع جميع إشارات بروتوكول SS7/MAP، وتوجيه SCCP، والتواصل الشبكي
- **OmniHSS (خلفية HSS)**: تدير بيانات المشتركين، والمصادقة، والتزويد، والميزات المتقدمة

## لماذا OmniHSS؟

يوفر OmniHSS إدارة مشتركة على مستوى الناقل مع ميزات تشمل:

- **دعم متعدد IMSI**: يمكن لكل مشترك أن يكون لديه عدة IMSIs مرتبطة برقم MSISDN واحد للجوال الدولي، وتبديل الشبكات، وتزويد eSIM
- **مصادقة مرنة**: دعم لكل من خوارزميات المصادقة Milenage (3G/4G/5G) وCOMP128 (2G)
- **تتبع الجلسات الدائرية والحزم**: تتبع مستقل لتسجيلات الشبكة CS (الدائرة) وPS (الحزمة)
- **تزويد متقدم**: ملفات تعريف الخدمة القابلة للتخصيص، والخدمات التكميلية، وبيانات اشتراك CAMEL
- **تصميم يعتمد على API**: واجهة برمجة تطبيقات HTTP REST للتكامل مع أنظمة الفوترة، وإدارة علاقات العملاء، وأنظمة التزويد
- **تحديثات في الوقت الحقيقي**: تتبع الموقع، وإدارة الجلسات، وتوليد متجهات المصادقة

تُخزن جميع بيانات المشتركين، وبيانات الاعتماد الخاصة بالمصادقة، وتكوينات الخدمة وتُدار في OmniHSS. يستعلم OmniSS7 عن OmniHSS عبر مكالمات API HTTPS للرد على عمليات SendRoutingInfo وSendAuthenticationInfo وUpdateLocation مثل MAP.

هو **واجهة إشارة فقط**. يتم التعامل مع جميع منطق إدارة OmniSS7 في HLR **مهم**: وضع OmniHSS. المشتركين، وخوارزميات المصادقة، وقواعد التزويد، وعمليات قاعدة البيانات بواسطة لمزيد من المعلومات حول تزويد OmniSS7. يغطي هذا الدليل تكوين بروتوكول SS7/MAP في المشتركين، وتكوين المصادقة، وملفات تعريف الخدمة، والعمليات الإدارية، **يرجى الرجوع إلى وثائق OmniHSS**.

## دعم متعدد IMSI

**يدعم OmniHSS بشكل أصلي تكوينات متعددة IMSI**، مما يسمح لمشترك واحد (معرف برقم MSISDN) أن يكون لديه عدة IMSIs. وهذا يمكّن:

- **ملفات تعريف التجوال الدولي**: IMSIs مختلفة لمناطق مختلفة لتقليل تكاليف التجوال
- **ملف تعريف متعدد eSIM**: ملفات تعريف شبكة متعددة على جهاز واحد يدعم eSIM
- **تبديل الشبكات**: تبديل سلس بين الشبكات دون تغيير MSISDN
- **تنسيق شريحة مزدوجة**: التنسيق عبر عدة شرائح فعلية أو افتراضية
- **الاختبار والتطوير**: عدة IMSIs اختبارية تشير إلى نفس المشترك

**كيف يعمل:**

- لكل IMSI l��ب بيانات اعتماد مصادقة خاصة (Ki، OPc، الخوارزمية)
- يمكن أن يكون لكل IMSI تسجيلات جلسات دائرية وحزم مستقلة
- يمكن مشاركة خدمات المشتركين وملفات التعريف أو تخصيصها لكل IMSI
- يستعلم OmniSS7 عن OmniHSS حسب IMSI، وتعيد OmniHSS بيانات المشترك المناسبة
- يمكن أن تتعقب أنظمة الفوترة الاستخدام حسب IMSI مع ربط جميع IMSIs بحساب واحد

**مثال على سيناريو متعدد IMSI:**

```
رقم MSISDN 4567-123-555-1+ :للمشترك
├─ IMSI 1: 310260123456789 (مصادقة - شبكة منزلية أمريكية Milenage)
├─ IMSI 2: 208011234567890 (مصادقة - ملف تجوال فرنسا Milenage)
└─ IMSI 3: 440201234567891 (مصادقة - ملف تجوال المملكة المتحدة
COMP128)
```

الثلاثة بشكل مستقل لتسجيل الشبكة، لكنها جميعها تنتمي إلى نفس IMSIs يمكن استخدام جميع إلى المشترك وتضمن المصادقة والتزويد المناسبين IMSI تعيين OmniHSS حساب المشترك. تدير لكل IMSI.

# جدول المحتويات


1. تكامل OmniHSS
2. دعم متعدد IMSI
3. ما هو وضع HLR؟
4. تمكين وضع HLR
5. قاعدة بيانات المشتركين
6. متجهات المصادقة

7. تحديثات الموقع
8. تكامل CAMEL
9. معالجة المشتركين المتجولين
10. عمليات HLR

- تعيين حقول الاستجابة
  - SendRoutingInfo (SRI)
  - UpdateLocation / ISD
  - SendRoutingInfoForSM
- ملخص مصدر الحقول

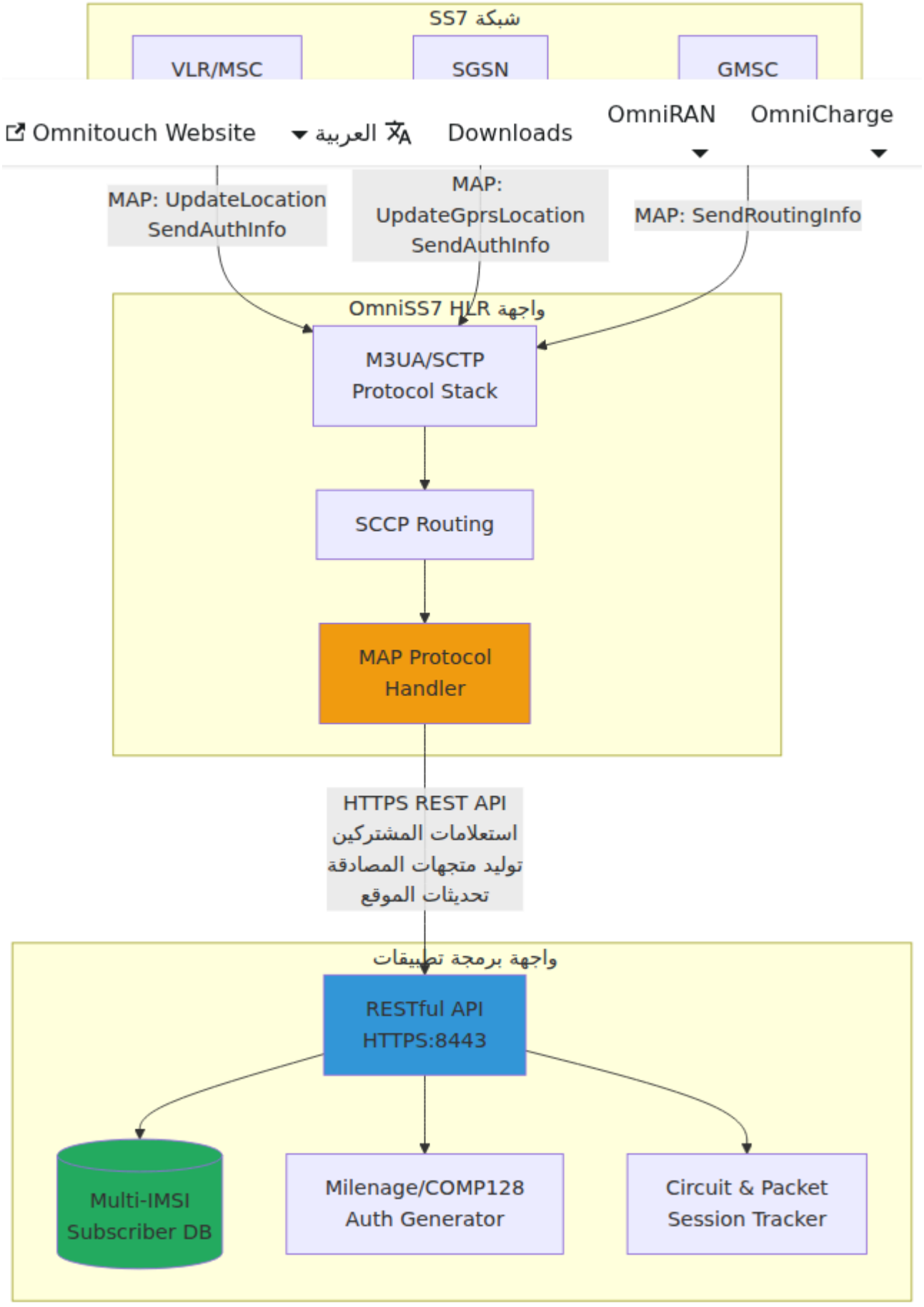
---

# ما هو وضع HLR؟

**وضع HLR** يمكّن OmniSS7 من العمل كسجل موقع منزلي لـ:

- **إدارة المشتركين**: تخزين وإدارة بيانات المشتركين
- **المصادقة**: توليد متجهات المصادقة للوصول إلى الشبكة
- **تتبع الموقع**: معالجة تحديثات الموقع من VLRs
- **معلومات التوجيه**: توفير معلومات التوجيه للمكالمات والرسائل القصيرة

# HLR بن


شبكة SS7
VLR/MSC
SGSN
GMSC
Omnitouch Website
العربية
Downloads
OmniRAN
OmniCharge
MAP: UpdateLocation
SendAuthInfo
MAP:
UpdateGprsLocation
SendAuthInfo
MAP: SendRoutingInfo
واجهة OmniSS7 HLR
M3UA/SCTP
Protocol Stack
SCCP Routing
MAP Protocol
Handler
HTTPS REST API
استعلامات المشتركين
توليد متجهات المصادقة
تحديثات الموقع
واجهة برمجة تطبيقات
RESTful API
HTTPS:8443
Multi-IMSI
Subscriber DB
Milenage/COMP128
Auth Generator
Circuit & Packet
Session Tracker

# تمكين وضع HLR

تحتاج إلى تمكين وضع HLR، في أوضاع مختلفة. لاستخدامه كـ OmniSS7 يمكن أن يعمل HLR في التكوين.

## التحويل إلى وضع HLR

يحتوي ملف `config/runtime.exs` في OmniSS7 ع��ى ثلاثة أوضاع تشغيل مسبقة التكوين. لتمكين وضع HLR:

1. **افتح** `config/runtime.exs`
2. **ابحث عن** أقسام التكوين الثلاثة (الأسطر 53-174):
   - وضع التكوين 1: STP (الأسطر 53-85)
   - وضع التكوين 2: HLR (الأسطر 87-123)
   - وضع التكوين 3: SMSc (الأسطر 125-174)
3. **علق** التكوين النشط حاليًا (أضف `#` إلى كل سطر)
4. **قم بإلغاء تعليق** تكوين HLR (قم بإزالة `#` من الأسطر 87-123)
5. **خصص** معلمات التكوين حسب الحاجة
6. **أعد تشغيل** التطبيق: `iex -S mix`

## تكوين وضع HLR

يبدو تكوين HLR الكامل كما يلي:

```
config :omniss7,
  # تمكين ميزات - علامات الوضع HLR فقط
  map_client_enabled: true,
  hlr_mode_enabled: true,
  smsc_mode_enabled: false,

  # تكوين واجهة برمجة تطبيقات OmniHSS الخلفية
  hlr_api_base_url: "https://10.180.2.140:8443",

  # عنوان GT لمركز خدمة HLR لعمليات الرسائل القصيرة
  hlr_service_center_gt_address: "1234567890",

  # تكوين تعيين MSISDN ↔ IMSI
  # انظر: قسم MSISDN ↔ IMSI Mapping للحصول على التفاصيل
  hlr_imsi_plmn_prefix: "50557",
  hlr_msisdn_country_code: "61",
  hlr_msisdn_nsn_offset: 0,
  hlr_msisdn_nsn_length: 9,

  # تكوين InsertSubscriberData
  # وضع الوصول إلى الشبكة: :packetAndCircuit، :packetOnly، أو
:circuitOnly
  isd_network_access_mode: :packetAndCircuit,

  # إرسال ISD #2 (بيانات الخدمات التكميلية)
  isd_send_ss_data: true,

  # إرسال ISD #3 (بيانات حظر المكالمات)
  isd_send_call_barring: true,

  # تكوين CAMEL (لردود SendRoutingInfo)
  # مفتاح الخدمة لبدء خدمة CAMEL
  camel_service_key: 11_110,

  # نقطة اكتشاف الزناد CAMEL
  # الخيارات: :termAttemptAuthorized، :tBusy، :tNoAnswer، :tAnswer
  camel_trigger_detection_point: :termAttemptAuthorized,

  # بادئات VLR المنزل
  # قائمة بادئات عنوان VLR منزلية "شبكة" التي تعتبر
  # إذا بدأ VLR استخدم رد ،هذه البادئات بأحد للمشترك SRI القياسي
  # خلاف ذلك، يكون المشترك متجولًا ونحتاج إلى إرسال PRN للحصول على
MSRN
```

```
  home_vlr_prefixes: ["123456"],

  # تكوين اتصال M3UA
  # الاتصال كـ ASP لتلقي عمليات MAP (UpdateLocation، SendAuthInfo، إلخ.)
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :hlr_client_asp,
    # نقطة النهاية المحلية (نظام HLR)
    local_ip: {10, 179, 4, 11},
    local_port: 2905,
    # نقطة النهاية البعيدة STP
    remote_ip: {10, 179, 4, 10},
    remote_port: 2905,
    routing_context: 1
  }
```

## معلمات التكوين للتخصيص

للحصول على مرجع كامل لجميع معلمات التكوين، انظر مرجع التكوين.

| المعلمة | النوع | الافتراضي |
| --- | --- | --- |
| `hlr_api_base_url` | سلسلة | *مطلوب* |
| `hlr_service_center_gt_address` | سلسلة | *مطلوب* |
| `smsc_service_center_gt_address` | سلسلة | *مطلوب* |
| `hlr_smsc_alert_gts` | قائمة | `[]` |
| `hlr_alert_location_expiry_seconds` | عدد صحيح | `172800` |
| `hlr_imsi_plmn_prefix` | سلسلة | `"50557"` |
| `hlr_msisdn_country_code` | سلسلة | `"61"` |
| `hlr_msisdn_nsn_offset` | عدد صحيح | `0` |

| المعلمة | النوع | الافتراضي |
| --- | --- | --- |
| | | |
| `hlr_msisdn_nsn_length` | عدد صحيح | `9` |
| `isd_network_access_mode` | ذرة | `:packetAndCircuit` |
| `isd_send_ss_data` | بولياني | `true` |
| `isd_send_call_barring` | بولياني | `true` |
| `camel_service_key` | عدد صحيح | `11_110` |
| `camel_trigger_detection_point` | ذرة | `:termAttemptAuthorized` |
| `home_vlr_prefixes` | قائمة | `["5551231"]` |
| `local_ip` | مجموعة | *مطلوب* |
| `local_port` | عدد صحيح | `2905` |

| المعلمة | النوع | الافتراضي |
|---|---|---|
| `remote_ip` | مجموعة | *مطلوب* |
| `remote_port` | عدد صحيح | `2905` |
| `routing_context` | عدد صحيح | `1` |

## HLR ماذا يحدث عند تمكين وضع

عند `hlr_mode_enabled: true`، ستظهر واجهة الويب:

- **SS7 أحداث** - تسجيل الأحداث
- **SS7 عميل** - اختبار عمليات MAP
- **M3UA** - حالة الاتصال
- **روابط HLR** - إدارة المشتركين ← محدد *لـ* + HLR حالة واجهة برمجة تطبيقات *HLR*
- **الموارد** - مراقبة النظام
- **التكوين** - عارض التكوين

**SMSc** ستكون علامات التبويب **التوجيه** و**اختبار التوجيه** و**روابط** مخفية.

## ملاحظات هامة

- **التكوين المطلوب**: المعلمة `hlr_service_center_gt_address` هي **إلزامية**. سيفشل التطبيق في البدء إذا لم يتم تكوينها.
- **خلفية OmniHSS**: يجب أن تكون واجهة برمجة تطبيقات OmniHSS الخلفية قابلة للوصول عند المكون `hlr_api_base_url`
- **مهلة طلب API**: جميع طلبات واجهة برمجة تطبيقات OmniHSS لديها **مهلة ثابتة مدتها 5 ثوانٍ**
- **مهلة طلب MAP**: جميع طلبات MAP (SRI، UpdateLocation، SendAuthInfo، إلخ.) لديها **مهلة ثابتة مدتها 10 ثوانٍ**

- **مهلة ISD**: كل رسالة InsertSubscriberData (ISD) في تسلسل UpdateLocation لديها **مهلة ثابتة مدتها 10 ثوانٍ**
- MAP لتلقي عمليات STP بـ M3UA يتطلب اتصال
- بعد تغيير الأوضاع، يجب إعادة تشغيل التطبيق لتفعيل التغييرات
- **واجهة الويب**: انظر دليل واجهة الويب للحصول على معلومات حول استخدام الواجهة الويب
- **الوصول إلى واجهة برمجة التطبيقات**: انظر دليل واجهة برمجة التطبيقات للحصول على وثائق واجهة برمجة التطبيقات REST والوصول إلى واجهة Swagger

---

# قاعدة بيانات المشتركين

**يدير OmniHSS** **جميع بيانات المشتركين** بما في ذلك الهويات، وبيانات الاعتماد الخاصة بالمصادقة، وملفات تعريف الخدمة، ومعلومات الموقع. يسترجع OmniSS7 هذه البيانات عبر مكالمات واجهة برمجة التطبيقات RESTful.

## نموذج مشترك OmniHSS

يخزن OmniHSS معلومات شاملة عن المشتركين:

- **عدة IMSIs لكل مشترك**: دعم لتكوينات متعددة IMSI (eSIM، تبديل ملفات التجوال، الشبكات)
- **بيانات اعتماد المصادقة**: Ki، OPc، واختيار الخوارزمية (Milenage أو COMP128)
- **ملفات تعريف الخدمة**: فئة المشترك، والخدمات المسموح بها، ومعلمات QoS
- **تتبع الموقع**: تتبع مستقل لموقع VLR/MSC و (جلسة دائرية) ISGSN/GGSN الحالي (جلسة حزمة)
- **بيانات اشتراك CAMEL**: مفاتيح الخدمة، ونقاط الزناد، وعناوين gsmSCF
- **الخدمات التكميلية**: إعادة توجيه المكالمات، والحظر، والانتظار، وتكوينات CLIP/CLIR
- **الحالة الإدارية**: مفعل/معطل، قيود الخدمة، تواريخ انتهاء الصلاحية

---

# متجهات المصادقة

## توليد متجهات المصادقة

**يولد OmniHSS** **متجهات المصادقة** باستخدام خوارزميات Milenage أو COMP128 بناءً على طريقة المصادقة المكونة لكل مشترك. عندما يتلقى OmniSS7 طلبات MAP **sendAuthenticationInfo**:

1. يستخرج OmniSS7 IMSI من طلب MAP
2. يستدعي OmniSS7 واجهة برمجة تطبيقات OmniHSS لتوليد متجهات المصادقة
3. يسترجع OmniHSS بيانات اعتماد Ki وOPc الخاصة بالمشترك
4. يولد OmniHSS العدد المطلوب من المتجهات (RAND، XRES، CK، IK، AUTN)
5. يقوم OmniSS7 بترميز المتجهات في تنسيق MAP وإعادتها إلى VLR/SGSN المطلوب

## تكامل واجهة برمجة تطبيقات OmniHSS

يتواصل OmniSS7 مع OmniHSS عبر واجهة برمجة تطبيقات HTTPS REST لاسترجاع معلومات المشتركين، وتحديث بيانات الموقع، وتوليد متجهات المصادقة:

```
config :omniss7,
  hlr_api_base_url: "https://omnihss-server:8443"
```

عندما يتلقى OmniSS7 عمليات MAP من شبكة SS7، يستعلم OmniHSS لـ:

- حسب MSISDN أو IMSI **استرجاع بيانات المشترك**
- المخزنة Ki/OPc باستخدام بيانات اعتماد **توليد متجهات المصادقة**
- UpdateLocation عندما يقوم المشتركون بتنفيذ **تحديث موقع جلسة الدائرة**
- وحقوق الخدمة **التحقق من حالة المشترك**

---

# تحديثات الموقع

## معالجة تحديث الموقع

عند استلام طلبات MAP **updateLocation**، يقوم OmniSS7 بالتنسيق مع OmniHSS لتسجيل المشترك في VLR جديد:

1. **استخراج معلومات الموقع** من طلب UpdateLocation (IMSI، GT VLR الجديد، GT MSC الجديد)
2. **استعلام OmniHSS** للتحقق من وجود المشترك وتمكينه
3. **تحديث جلسة الدائرة** في OmniHSS بموقع VLR/MSC الجديد
4. **إرسال رسائل InsertSubscriberData (ISD)** لتزويد المشترك في VLR الجديد
5. **إرجاع استجابة UpdateLocation** إلى VLR (تتضمن GT HLR من `hlr_service_center_gt_address`)
6. **إرسال alertServiceCenter** إلى GTs لـ SMSc المكونة (إذا كانت `hlr_smsc_alert_gts` ممتلئة)

**ملاحظة**: تحدد المعلمة `hlr_service_center_gt_address` عنوان GT لـ HLR الذي يُعاد في ردود UpdateLocation. وهذا يسمح لـ VLR/MSC هذا بالتعرف على الرسائل وإعادة توجيهها إلى HLR.

## تكامل مركز الخدمة التنبيهي

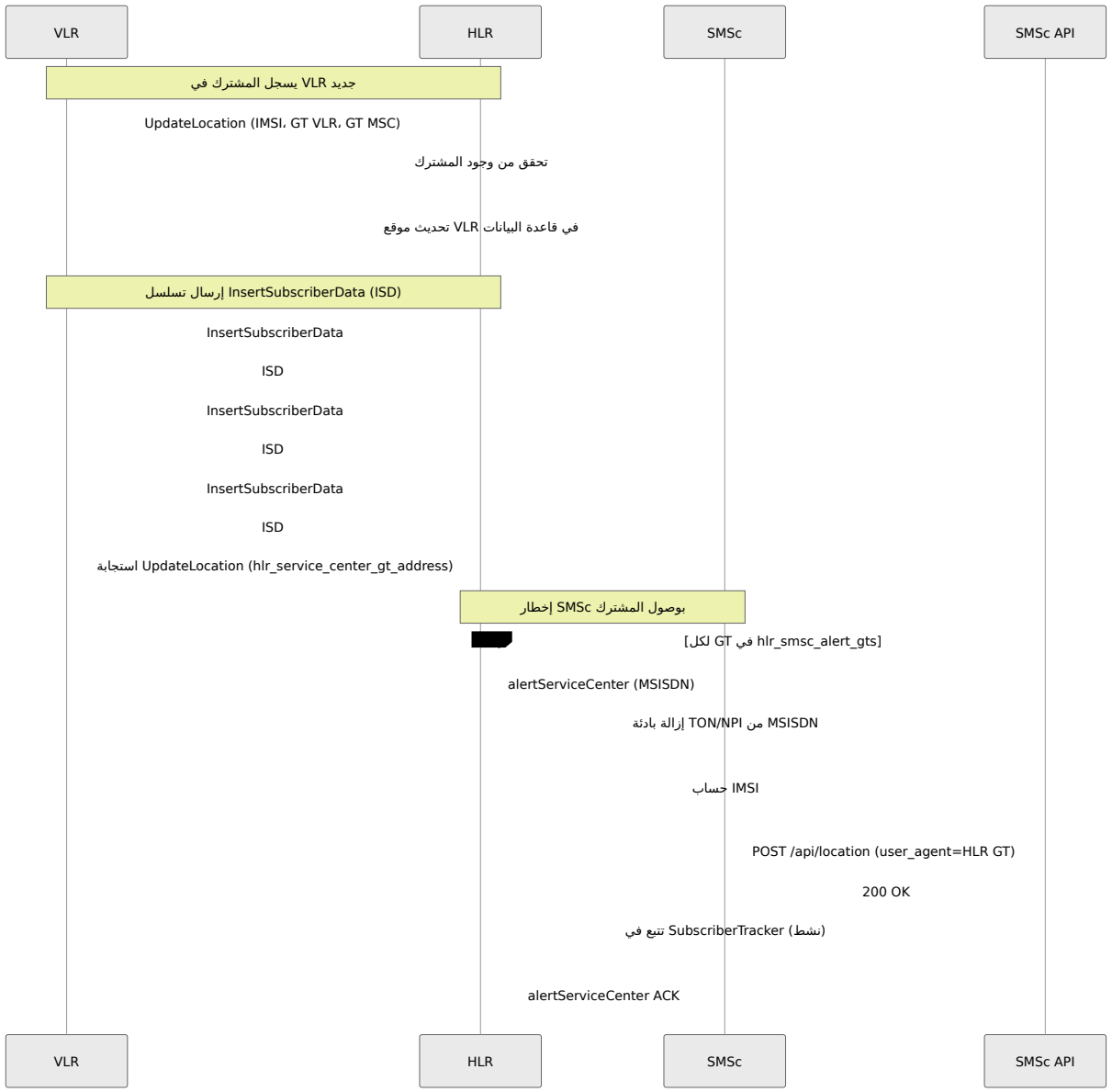
بعد نجاح UpdateLocation، يمكن لـ HLR إعلام أنظمة SMSc الآن تلقائيًا بأن المشترك أصبح قابلاً للوصول عن طريق إرسال رسائل **alertServiceCenter** (عملية MAP 64). لمزيد من المعلومات حول كيفية تعامل SMSc مع هذه التنبيهات، انظر معالجة مركز الخدمة التنبيهي في دليل SMSc.

### التكوين

قم بتكوين قائمة من عناوين GT لـ SMSc رها��لإخط:

```
config :omniss7,
  # قائمة من GTs لـ SMSc لإرسال alertServiceCenter بعد
UpdateLocation
  hlr_smsc_alert_gts: [
    "15559876543",
    "15559876544"
  ],

  # وقت انتهاء صلاحية الموقع عند استلام SMSc alertServiceCenter
(الافتراضي: 48 ساعة)
  hlr_alert_location_expiry_seconds: 172800
```

## مخطط التدفق



**السلوك**

عندما يقوم المشترك بتنفيذ UpdateLocation:

1. يرسل HLR alertServiceCenter **كل** إلى GT في قائمة `hlr_smsc_alert_gts`
2. تتضمن الرسالة MSISDN الخاص بالمشترك
3. يستخدم HLR `hlr_service_center_gt_address` كـ GT للمتصل
4. توجيه SCCP: SSN 6=المتصل (HLR)، SSN 8=المستلم (SMSc)

يتلقى SMSc التنبيه و:

- **يزيل بادئة TON/NPI** من MSISDN (→ "19123123213"، على سبيل المثال "123123213"")
- يحدد المشترك كقابل للوصول في قاعدة بيانات الموقع الخاصة به (عبر POST إلى /api/location)
- **يحدد حقل** `user_agent` إلى HLR GT (للتتبع عند استدعاء واجهة برمجة التطبيقات أي HLR أرسل التنبيه)
- يحدد وقت انتهاء صلاحية الموقع بناءً على `hlr_alert_location_expiry_seconds`
- يتتبع المشترك في SMSc Subscriber Tracker للمراقبة

**الاختبار**

استخدم صفحة **المشتركين النشطين** في واجهة الويب لإرسال رسائل alertServiceCenter يدويًا للاختبار:

1. انتقل إلى علامة التبويب "المشتركين النشطين"
2. ابحث عن قسم "اختبار مركز الخدمة التنبيهي"
3. أدخل MSISDN، GT SMSc، وGT HLR (القيم الافتراضية مُعبأة مسبقًا من التكوين)
   - GT SMSc الافتراضي هو أول إدخال في `hlr_smsc_alert_gts`
   - GT HLR الافتراضي هو `hlr_service_center_gt_address`
4. انقر على "إرسال alertServiceCenter"

هذا مفيد لاختبار معالجة تنبيهات SMSc دون الحاجة إلى تدفق UpdateLocation كامل. يستخدم النموذج التحقق من `phx-blur` لتجنب عرض الأخطاء أثناء الكتابة.

# تكوين InsertSubscriberData (ISD)

بعد نجاح UpdateLocation، يرسل HLR بيانات تزويد المشترك إلى VLR باستخدام رسائل **InsertSubscriberData** (ISD). يسمح تكوين ISD وكيفية لك بتخصيص البيانات المرسلة وكيفية ذلك.

للحصول على مرجع معلمات التكوين، انظر تكوين ISD في مرجع التكوين.

### تسلسل ISD

يمكن لـ HLR إرسال ما يصل إلى 3 رسائل ISD متتالية:

1. **ISD #1** (يتم إرساله دائمًا) - بيانات المشترك الأساسية:
    - IMSI
    - MSISDN
    - فئة المشترك
    - حالة المشترك (serviceGranted)
    - قائمة خدمات الحامل
    - قائمة خدمات الهاتف
    - وضع الوصول إلى الشبكة
2. **ISD #2** (اختياري) - بيانات الخدمات التكميلية (SS):
    - إعدادات إعادة توجيه المكالمات (غير مشروط، مشغول، لا رد، غير قابل للوصول)
    - انتظار المكالمات
    - حجز المكالمات
    - خدمة متعددة الأطراف
    - حالة وميزات الخدمة التكميلية
3. **ISD #3** (اختياري) - بيانات حظر المكالمات:
    - حظر جميع المكالمات الصادرة (BAOC)
    - حظر المكالمات الدولية الصادرة (BOIC)
    - بيانات قيود الوصول

### خيارات التكوين

```
# تكوين InsertSubscriberData
# وضع الوصول إلى الشبكة: :packetAndCircuit، :packetOnly، أو
:circuitOnly
isd_network_access_mode: :packetAndCircuit,

# إرسال ISD #2 (بيانات الخدمات التكميلية)
isd_send_ss_data: true,

# إرسال ISD #3 (بيانات حظر المكالمات)
isd_send_call_barring: true,
```

**وضع الوصول إلى الشبكة**

تتحكم معلمة `isd_network_access_mode` في نوع الوصول إلى الشبكة المسموح به للمشترك:

| القيمة | الوصف | حالة الاستخدام |
|---|---|---|
| `:packetAndCircuit` | كل من تبديل الحزمة (GPRS/LTE) والتبديل الدائري (الصوت) | الافتراضي - مشتركون كاملو الخدمة |
| `:packetOnly` | تبديل الحزمة فقط (البيانات/LTE) | بطاقات SIM للبيانات فقط، أجهزة IoT |
| `:circuitOnly` | تبديل الدائرة فقط (الصوت/SMS) | أجهزة قديمة، خطط صوتية فقط |

**التحكم في رسائل ISD**

يمكنك التحكم في الرسائل ISD المرسلة بناءً على متطلبات الشبكة الخاصة بك:

**إرسال جميع ISDs** (الافتراضي - مجموعة ميزات كاملة):

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

**إرسال بيانات المشترك الأساسية فقط** (تزويد الح�ّ الأدنى):

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

**إرسال الأساسية + الخدمات التكميلية** (بدون حظر المكالمات):

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

**مثال على تدفق ISD**

عند استلام UpdateLocation:

```
VLR → HLR: UpdateLocation (BEGIN)
HLR → VLR: InsertSubscriberData #1 (CONTINUE) - البيانات الأساسية
VLR → HLR: ISD #1 ACK (CONTINUE)
HLR → VLR: InsertSubscriberData #2 (CONTINUE) - بيانات SS [إذا
كانت مفعلة]
VLR → HLR: ISD #2 ACK (CONTINUE)
HLR → VLR: InsertSubscriberData #3 (CONTINUE) - [إذا] حظر المكالمات
[كانت مفعلة
VLR → HLR: ISD #3 ACK (CONTINUE)
HLR → VLR: استجابة UpdateLocation (END)
```

إذا كانت `isd_send_ss_data` أو `isd_send_call_barring` مضبوطة على `false`، يتم تخطي تلك الرسائل ISD، ويتم إرسال END لـ UpdateLocation في وقت أقرب.

**أفضل الممارسات**

- **التكوين الافتراضي**: استخدم `:packetAndCircuit` وقم بتمكين جميع ISDs لتحقيق أقصى قدر من التوافق
- **IoT/M2M**: استخدم `:packetOnly` وقم بتعطيل بيانات SS وحظر المكالمات للأجهزة ا��تي تدعم البيانات فقط
- **التوافق**: قد لا تدعم بعض VLRs القديمة جميع الخدمات التكميلية - قم بتعطيل `isd_send_ss_data` إذا واجهت مشاكل
- **الأداء**: يؤدي تعطيل ISDs غير المستخدمة إلى تقليل الحمل الرسائلي وتسريع تحديثات الموقع

---

# تكامل CAMEL

## تكوين CAMEL لردود SendRoutingInfo

يمكن لـ HLR، عند الرد على طلبات **SendRoutingInfo** (SRI) من GMSC (Gateway MSC)، إرشاد GMSC لاستدعاء خدمات CAMEL في الخدمة لتوجيه المكالمات الذكي والتحكم.

للحصول على مرجع معلمات التكوين، انظر تكوين CAMEL في مرجع التكوين.

### ما هو CAMEL؟

**CAMEL** (تطبيقات مخصصة لشبكات الهاتف المحمول المعززة) هو بروتوكول يمكّن الخدمات الذكية في شبكات GSM/UMTS. يسمح لمشغلي الشبكات بتنفيذ خدمات ذات قيمة مضافة مثل:

- الفوترة المدفوعة مسبقًا
- تصفية المكالمات وحظرها
- الشبكات الخاصة الافتراضية (VPN)
- خدمات الأسعار المميزة
- إعادة توجيه المكالمات مع منطق مخصص
- خدمات قائمة على الموقع

### خيارات التكوين

```
# تكوين CAMEL (لردود SendRoutingInfo)
# مفتاح الخدمة لبدء خدمة CAMEL
camel_service_key: 11_110,

# نقطة اكتشاف الزناد CAMEL
# الخيارات: :termAttemptAuthorized، :tBusy، :tNoAnswer، :tAnswer
camel_trigger_detection_point: :termAttemptAuthorized,
```

### مفتاح الخدمة

تحدد `camel_service_key` الخدمة CAMEL في gsmSCF (وظيفة التحكم في الخدمة) التي يجب استدعاؤها. هذا هو معرف رقمي يتم تكوينه في شبكتك:

| مفتاح الخدمة | حالة الاستخدام النموذجية |
|---|---|
| `11_110` | التحكم في المكالمات المدفوعة مسبقًا (افتراضي) |
| `100` | خدمة مدفوعة مسبقًا للمكالمات الصادرة |
| `200` | إعادة توجيه المكالمات مع منطق مخصص |
| `300` | الشبكة الخاصة الافتراضية (VPN) |
| مخصص | خدمات محددة من قبل المشغل |

**مثال على التكوين:**

```
# للتحكم في المكالمات المدفوعة مسبقًا
camel_service_key: 11_110,

# لخدمة VPN
camel_service_key: 300,
```

**نقطة اكتشاف الزناد**

أثناء إعداد CAMEL متى يجب استدعاء خدمة `camel_trigger_detection_point` تحدد المكالمة:

| متى يتم استدعاؤها | الوصف | نقطة الاكتشاف |
| --- | --- | --- |
| قبل توجيه المكالمة إلى المشترك | تم تفويض محاولة المكالمة (افتراضي) | `:termAttemptAuthorized` |
| عندما يكون المشترك مشغولًا | مشغول نهائي | `:tBusy` |
| عندما لا يرد المشترك | لا رد نهائي | `:tNoAnswer` |
| عندما يرد المشترك على المكالمة | إجابة نهائية | `:tAnswer` |

**أمثلة على التكوين:**

**التحكم المدفوع مسبقًا القياسي** (استدعاء قبل التوجيه):

```
camel_trigger_detection_point: :termAttemptAuthorized,
```

**معالجة مخصصة للمشغول** (استدعاء عند المشغول):

```
camel_trigger_detection_point: :tBusy,
```

**الفوترة بناءً على الإجابة** (استدعاء عند الإجابة):

```
camel_trigger_detection_point: :tAnswer,
```

**استجابة SRI مع CAMEL**

معلومات اشتراك CAMEL: SendRoutingInfo عند التكوين، تتضمن ردو

```
GMSC → HLR: SendRoutingInfo (BEGIN)
HLR → GMSC: استجابة SRI (END) مع:
  - IMSI
  - رقم VLR
  - حالة المشترك
  - معلومات توجيه CAMEL:
    * مفتاح الخدمة: 110_11
    * عنوان gsmSCF: <عنوان مكون>
    * نقطة اكتشاف الزناد: termAttemptAuthorized
    * معالجة المكالمات الافتراضية: continueCall

يتصل GMSC بـ gsmSCF لتنفيذ خدمة CAMEL في نقطة الزناد
```

**أفضل الممارسات**

- **الشبكات الإنتاجية**: استخدم مفاتيح الخدمة القياسية المتفق عليها مع مزود gsmSCF الخاص بك
- **الاختبار**: استخدم `:termAttemptAuthorized` للاختبار الأكثر شمولاً
- **الخدمات المدفوعة مسبقًا**: مفتاح الخدمة `110_11` هو معيار صناعي شائع للمكالمات المدفوعة مسبقًا
- **معالجة التراجع**: يضمن `defaultCallHandling: :continueCall` أن المكالمات تستمر إذا كان gsmSCF غير متاح

---

# معالجة المشتركين المتجولين

## الكشف عن VLR المتجول مقابل VLR المنزلي

يحتاج إلى تحديد ما إذا كان المشترك على، عندما يتلقى HLR طلب **SendRoutingInfo** (SRI). يختلف السلوك بناءً على VLR منزلي" (داخل شبكتك) أو على "VLR متجول (يزور شبكة أخرى). هذا التحديد:

للحصول على مرجع معلمات التكوين، انظر بادئات VLR المنزلية في مرجع التكوين.

- **VLR المنزلي**: إرجاع رد SRI القياسي مع معلومات توجيه CAMEL
- **VLR المتجول**: إرسال طلب توفير رقم التجوال (PRN) للحصول على MSRN، ثم إعادته في رد SRI

# التكوين

```
# بادئات VLR المنزل
# قائمة من بادئات عنوان VLR شبكة منزلية "التي تعتبر"
# للمشترك بأحد هذه البادئات، استخدم رد VLR إذا بدأ عنوان SRI
القياسي
# للحصول على PRN خلاف ذلك، يكون المشترك متجولًا ونحتاج إلى إرسال MSRN
home_vlr_prefixes: ["555123"],
```

**مثال على التكوين**:

```
# مشغل شبكة منزلية واحدة
home_vlr_prefixes: ["555123"],

# (مثل مناطق مختلفة أو فروع) مشغلو شبكة متعددة
home_vlr_prefixes: ["555123", "555124", "555125"],
```

# كيف يعمل

### 1. تدفق المشترك المنزلي (قياسي)

عندما يبدأ عنوان VLR للمشترك ببادئة منزلية مكونة:

```
GMSC → HLR: SendRoutingInfo (MSISDN: "1234567890")
HLR يستعلم عن واجهة برمجة التطبيقات الخلفية لبيانات المشترك
HLR يتحقق من عنوان VLR: "5551234567"
HLR يحدد: VLR شبكة منزلية → "555123" يبدأ بـ
HLR → GMSC: استجابة SRI مع معلومات توجيه CAMEL:
  - IMSI
  - رقم VLR: "5551234567"
  - عنوان gsmSCF (MSC): "5551234501"
  - مفتاح خدمة CAMEL: 11_110
  - نقطة اكتشاف الزناد: termAttemptAuthorized
```

### 2. تدفق المشترك المتجول (مطلوب PRN)

عندما لا يتطابق عنوان VLR للمشترك مع أي بادئة منزلية مكونة:

```
GMSC → HLR: SendRoutingInfo (MSISDN: "1234567890")
HLR يستعلم عن واجهة برمجة التطبيقات الخلفية لبيانات المشترك
HLR يتحقق من عنوان VLR: "49170123456"
HLR يحدد: VLR متجول → "555123" لا يبدأ بـ
HLR → MSC: ProvideRoamingNumber (PRN):
  - MSISDN: "1234567890"
  - IMSI: "999999876543210"
  - رقم MSC: "49170123456"
  - عنوان GMSC: "5551234501"
MSC → HLR: استجابة PRN مع MSRN: "49170999888777"
HLR → GMSC: استجابة SRI مع معلومات التوجيه:
  - IMSI
  - رقم VLR: "49170123456"
  - رقم التجوال (MSRN): "49170999888777"
```

## اختلافات هيكل الاستجابة

**استجابة المشترك المنزلي SRI**

```
%{
  imsi: "999999876543210",
  extendedRoutingInfo: {
    :camelRoutingInfo, %{
      gmscCamelSubscriptionInfo: %{
        "t-CSI": %{
          serviceKey: 11_110,
          "gsmSCF-Address": "5551234501",
          defaultCallHandling: :continueCall,
          "t-BcsmTriggerDetectionPoint": :termAttemptAuthorized
        }
      }
    }
  },
  subscriberInfo: %{
    locationInformation: %{"vlr-number": "5551234567"},
    subscriberState: {:notProvidedFromVLR, :NULL}
  }
}
```

**استجابة المشترك المتجول SRI**

```
%{
  imsi: "999999876543210",
  extendedRoutingInfo: {
    :routingInfo, %{
      roamingNumber: "49170999888777"  # MSRN من PRN
    }
  },
  subscriberInfo: %{
    locationInformation: %{"vlr-number": "49170123456"},
    subscriberState: {:notProvidedFromVLR, :NULL}
  }
}
```

# عملية توفير رقم التجوال (PRN)

### هيكل طلب PRN

يتضمن طلب PRN المرسل إلى MSC/VLR:

| الحقل | المصدر | الوصف |
|---|---|---|
| **MSISDN** | طلب SRI | رقم هاتف المشترك |
| **IMSI** | واجهة برمجة التطبيقات HLR | IMSI الخاص بالمشترك |
| **رقم MSC** | واجهة برمجة التطبيقات HLR | MSC الذي يخدم المشترك المتجول (`serving_msc`) |
| **عنوان GMSC** | طلب SRI | GMSC الذي يقوم بإجراء طلب SRI الأصلي |
| **رقم المرجع للمكالمة** | ثابت | معرف مرجع المكالمة |
| **مراحل CAMEL المدعومة** | ثابت | مراحل CAMEL المدعومة من قبل GMSC |

**معالجة استجابة PRN**

يتوقع HLR استجابة PRN تحتوي على:

- **MSRN** (رقم محطة التجوال المتنقل): رقم مؤقت مخصص بواسطة الشبكة الزائرة لتوجيه المكالمة

**معالجة الأخطاء**:

- إذا انتهت مدة PRN → في استجابة (المشترك غائب) 27 ترجع خطأ SRI
- إذا فشل PRN → في استجابة (المشترك غائب) 27 ترجع خطأ SRI
- إذا لم يمكن استخراج MSRN → في استجابة (المشترك غائب) 27 ترجع خطأ SRI

# أمثلة على التكوين

**مشغل شبكة منزلية واحدة**

```
# جميع عناوين VLR تعتبر منزلية "555123" التي تبدأ بـ
home_vlr_prefixes: ["555123"],
```

- VLR `5551234567` → منزلي (استجابة CAMEL)
- VLR `5551235001` → منزلي (استجابة CAMEL)
- VLR `49170123456` → متجول (PRN + استجابة MSRN)

**مشغل شبكة متعددة المناطق**

```
# شبكات منزلية متعددة عبر مناطق مختلفة
home_vlr_prefixes: ["555123", "555124", "555125"],
```

- VLR `5551234567` → منزلي (المنطقة 1)
- VLR `5552341234` → منزلي (المنطقة 2)
- VLR `5553411111` → منزلي (المنطقة 3)
- VLR `44201234567` → متجول (دولي)

**تكوين الاختبار**

لاختبار وظيفة PRN، قم بتعيين قائمة فارغة لمعالجة جميع VLRs كمتجولين:

```
# يتم التعامل مع جميع VLRs كمتجولين (لاختبار تدفق PRN)
home_vlr_prefixes: [],
```

## أفضل الممارسات

- **اختيار البادئات** لشبكتك (مثل رمز الدولة + رمز الشبكة): استخدم أقصر بادئة فريدة تحدد VLRs
- **بادئات متعددة** في شبكتك، بما في ذلك مناطق وفروع VLR: تضمين جميع بادئات مختلفة
- **اتفاقيات التجوال** بشكل صحيح من قبل الشبكات الشريكة PRN: التأكد من دعم المتجولة
- **الاختبار**: اختبار السيناريوهات المنزلية والمتجولة بشكل شامل قبل نشر الإنتاج
- **المراقبة** لتحديد مشاكل الاتصال مع الشركاء PRN: مراقبة معدلات انتهاء مدة المتجولين

## استكشاف الأخطاء وإصلاحها

**العرض**: يتم التعامل مع جميع المشتركين كمتجولين

- **السبب**: لم يتم تكوين `home_vlr_prefixes` أو لا تتطابق البادئات مع عناوين VLR
- **الحل**: تحقق من عناوين VLR في قاعدة البيانات الخاصة بك وقم بتحديث البادئات وفقًا لذلك

**العرض**: طلبات PRN تنتهي مدةها

- **السبب**: مشاكل الاتصال بالشبكة إلى MSC/VLR الشريكة المتجولة
- **الحل**: تحقق من توجيه M3UA/SCCP إلى عناوين MSC البعيدة

**العرض**: SRI غير صالح في استجابة MSRN

- **السبب**: تنسيق استجابة PRN من الشريك المتجول لا يتطابق مع الهيكل المتوقع
- **الحل**: مراجعة سجلات استجابة PRN وضبط `extract_msrn_from_prn/1` إذا لزم الأمر

---

# عمليات HLR

## عمليات MAP المدعومة

- `updateLocation` (Opcode 2) - تسجيل موقع VLR
- `sendAuthenticationInfo` (Opcode 56) - توليد متجهات المصادقة
- `sendRoutingInfo` (Opcode 22) - توفير MSRN للمكالمات مع دعم CAMEL
- `sendRoutingInfoForSM` (Opcode 45) - توفير GT MSC للرسائل القصيرة
- `cancelLocation` (Opcode 3) - إلغاء التسجيل من VLR القديم
- `insertSubscriberData` (Opcode 7) - دفع ملف تعريف المشترك

## تعيين حقول الاستجابة

توضح هذه القسم أين تأتي كل حقل في ردود HLR.

#### استجابة SendRoutingInfo (SRI)

**الغرض**: توفير معلومات التوجيه للمكالمات الواردة إلى مشترك.

يوفر HLR نوعين مختلفين من الردود بناءً على ما إذا كان المشترك على VLR منزلي أو متجول:

##### استجابة المشترك المنزلي (توجيه CAMEL)

تستخدم عندما يبدأ عنوان VLR للمشترك ببادئة `home_vlr_prefixes` المكونة.

**هيكل الاستجابة**:

| الحقل | المصدر | الوصف |
| --- | --- | --- |
| **IMSI** | واجهة برمجة التطبيقات OmniHSS | IMSI الخاص بالمشترك من قاعدة بيانات OmniHSS |
| **رقم VLR** | واجهة برمجة التطبيقات OmniHSS | VLR الحالي الذي يخدم المشترك (`circuit_session.assigned_vlr`) |
| **حالة المشترك** | ثابت | دائمًا `notProvidedFromVLR` |
| **extendedRoutingInfo** | - | النوع: `camelRoutingInfo` |
| **عنوان gsmSCF** | واجهة برمجة التطبيقات OmniHSS | MSC الذي يخدم المشترك (`circuit_session.assigned_msc`) |
| **مفتاح الخدمة** | runtime.exs | معرف خدمة CAMEL (`camel_service_key`) |
| **نقطة اكتشاف الزناد** | runtime.exs | متى يتم استدعاء CAMEL (`camel_trigger_detection_point`) |
| **معالجة قدرة CAMEL** | ثابت | مستوى دعم مرحلة CAMEL |
| **معالجة المكالمات الافتراضية** | ثابت | غير متاح gsmSCF الاحتياطي إذا كان |

#### استجابة المشترك المتجول (توجيه MSRN)

تستخدم عندما لا يتطابق عنوان VLR مع أي قيمة من `home_vlr_prefixes` للمشترك المكونة.

**هيكل الاستجابة**:

| الحقل | المصدر | الوصف | |
|---|---|---|---|
| **IMSI** | واجهة برمجة التطبيقات OmniHSS | IMSI الخاص بالمشترك من قاعدة بيانات OmniHSS | `"9` |
| **رقم VLR** | واجهة برمجة التطبيقات OmniHSS | VLR الحالي الذي يخدم المشترك (`circuit_session.assigned_vlr`) | `"4` |
| **حالة المشترك** | ثابت | دائمًا `notProvidedFromVLR` | `:n` |
| **extendedRoutingInfo** | - | النوع: `routingInfo` | - |
| **رقم التجوال (MSRN)** | استجابة PRN | MSRN الذي تم الحصول عليه من طلب ProvideRoamingNumber | `"4` |

**منطق قرار التوجيه**:

```
1. يستقبل OmniSS7 طلب SendRoutingInfo
2. يستعلم OmniSS7 عن بيانات المشترك من واجهة برمجة التطبيقات
OmniHSS
3. يتحقق OmniSS7 من عنوان VLR مقابل home_vlr_prefixes:

   إذا بدأ VLR ببادئة منزلية:
     → إرجاع معلومات توجيه CAMEL (تدفق المشترك المنزلي)

   إذا لم يتطابق VLR مع أي بادئة منزلية:
     → إرسال ProvideRoamingNumber (PRN) إلى MSC
     → استخراج MSRN من استجابة PRN
     → إرجاع معلومات التوجيه مع MSRN (تدفق المشترك المتجول)
```

**تدفق البيانات**:

- يستعلم OmniSS7 عن OmniHSS للحصول على معلومات المشترك
- تعيد OmniHSS IMSI، وموقع VLR/MSC الحالي، وحالة المشترك

- يستخدم OmniSS7 هذه البيانات لبناء استجابة MAP

**متطلبات التكوين**:

```
# في runtime.exs
home_vlr_prefixes: ["555123"],  # قائمة بادئات VLR المنزلية
```

**استجابات الأخطاء**:

- ترجع خطأ 27 (المشترك غائب): `null` `serving_msc` و `serving_vlr` إذا كانت
- إذا لم يتم العثور على المشترك: ترجع خطأ 1 (مشترك غير معروف)
- ترجع خطأ 27 (المشترك غائب): (حالة التجوال) PRN إذا انتهت مدة طلب
- غير صالحة (حالة التجوال): ترجع خطأ 27 (المشترك غائب) PRN إذا كانت استجابة

---

### استجابة InsertSubscriberData مع UpdateLocation

**الغرض**: تسجيل المشترك في VLR الجديد وتزويد بيانات المشترك.

#### استجابة UpdateLocation END

| الحقل | المصدر | الوصف | المثال |
|---|---|---|---|
| **رقم HLR** | runtime.exs | عنوان GT العالمي لهذا HLR (`hlr_service_center_gt_address`) | `"5551234568"` |
| **نوع رسالة TCAP** | ثابت | استجابة نهائية بعد جميع ISDs | `END` |

#### InsertSubscriberData #1 (بيانات المشترك الأساسية)

| الحقل | المصدر | الوصف | المثال |
| --- | --- | --- | --- |
| **IMSI** | الطلب | من طلب UpdateLocation | `"999999876543210"` |
| **MSISDN** | واجهة برمجة التطبيقات OmniHSS | رقم هاتف المشترك من OmniHSS | `"555123456"` |
| **الفئة** | ثابت | فئة المشترك | `"\n"` (0x0A) |
| **حالة المشترك** | ثابت | حالة الخدمة | `:serviceGranted` |
| **قائمة خدمات الحامل** | ثابت | خدمات الحامل المدعومة | `[<<31>>]` |
| **قائمة خدمات الهاتف** | ثابت | خدمات الهاتف المدعومة | `[<<17>>, "!", "\""]` |
| **وضع الوصول إلى الشبكة** | runtime.exs | الوصول إلى الحزمة/الدائرة (`isd_network_access_mode`) | `:packetAndCircuit` |

**InsertSubscriberData #2 (الخدمات التكم��لية) - اختياري**

| الحقل | المصدر | الوصف | يتحكم فيه |
|---|---|---|---|
| **SS المجهزة** | ثابت | بيانات الخدمات التكميلية | `isd_send_ss_data: true` |
| **إعادة توجيه المكالمات** | ثابت | إعدادات إعادة التوجيه (غير مشروط، مشغول، لا رد، غير قابل للوصول) | تم تفعيل التكوين |
| **انتظار المكالمات** | ثابت | حالة خدمة انتظار المكالمات | تم تفعيل التكوين |
| **خدمة متعددة الأطراف** | ثابت | دعم المكالمات الجماعية | تم تفعيل التكوين |

**ISD #2 تتضمن**:

- إعادة توجيه المكالمات غير المشروطة (رمز SS 21)
- إعادة توجيه المكالمات عند المشغول (رمز SS 41)
- إعادة توجيه المكالمات عند عدم الرد (رمز SS 42)
- إعادة توجيه المكالمات عند عدم الوصول (رمز SS 62)
- انتظار المكالمات (رمز SS 43)
- خدمة متعددة الأطراف (رمز SS 51)
- خدمات CLIP/CLIR

**InsertSubscriberData #3 (حظر المكالمات) - اختياري**

| الحقل | المصدر | الوصف | يتحكم فيه |
|---|---|---|---|
| **معلومات حظر المكالمات** | ثابت | إعدادات حظر المكالمات | `isd_send_call_barring: true` |
| **BAOC** | ثابت | حظر جميع المكالمات الصادرة (رمز SS 146) | تم تفعيل التكوين |
| **BOIC** | ثابت | حظر المكالمات الدولية الصادرة (رمز SS 147) | تم تفعيل التكوين |
| **بيانات قيود الوصول** | ثابت | قيود الوصول إلى الشبكة | تم تفعيل التكوين |

**تحكم تسلسل ISD**:

- ISD #1: يحتوي على بيانات المشترك الأساسية الضرورية - **يتم إرساله دائمًا**
- ISD #2: يتم إرساله فقط إذا كانت `isd_send_ss_data: true` في runtime.exs
- ISD #3: يتم إرساله فقط إذا كانت `isd_send_call_barring: true` في runtime.exs

---

**استجابة SendRoutingInfoForSM (SRI-for-SM)**

**الغرض**: توفير معلومات توجيه MSC/SMSC لتسليم الرسائل القصيرة. عندما يحتاج SMSc إلى تسليم رسالة قصيرة إلى مشترك، يرسل طلب SRI-for-SM إلى HLR لتحديد مكان توجيه الرسالة.

**هيكل الاستجابة**:

| الحقل | المصدر | الوصف | كيفية التوليد | |
|---|---|---|---|---|
| **IMSI** | محسوب | IMSI صناعي مشتق من MSISDN | `PLMN_PREFIX + zero_padded_MSISDN` | `"00` |
| **رقم العقدة الشبكية** | runtime.exs | عنوان GT لـ SMSC لتوجيه الرسائل القصيرة | `smsc_service_center_gt_address` | `"55` |

**معلمات التكوين** (من `runtime.exs`):

```
# عنوان مركز الخدمة GT (المعاد في ردود SRI-for-SM)
# المطلوب أين يجب إرسال SMSc هذا يخبر MT-ForwardSM
smsc_service_center_gt_address: "5551234567",  # مطلوب

# تكوين MSISDN ↔ IMSI
# بادئة PLMN: MCC (001 = شبكة اختبار) + MNC (01 = مشغل اختبار)
hlr_imsi_plmn_prefix: "001001",                # المعلمة الوحيدة المطلوبة!
```

---

## تعيين MSISDN ↔ IMSI

**معلمات التكوين**:

تتحكم هذه المعلمات في كيفية توليد OmniSS7 IMSIs الصناعية من MSISDNs لاستجابات SRI-for-SM:

- **`hlr_imsi_plmn_prefix`**: بادئة MCC+MNC لاستخدامها عند بناء IMSIs الصناعية (على سبيل المثال، "50557" لـ MCC=505، MNC=57)

- `hlr_msisdn_country_code`: رمز البلد الذي يتم إضافته عند إجراء تعيين عكسي IMSI→MSISDN (على سبيل المثال، "61" لأستراليا، "1" للولايات المتحدة/كندا)
- `hlr_msisdn_nsn_offset`: موضع الحرف الذي يبدأ فيه رقم المشترك الوطني (NSN) داخل MSISDN (رمز البلد، أو طول رمز البلد MSISDN عادةً 0 إذا لم تتضمن إذا كانت تتضمنه)
- `hlr_msisdn_nsn_length`: عدد الأرقام التي يجب استخراجها من MSISDN كـ NSN

للحصول على تفاصيل تكوين إضافية، انظر تعيين MSISDN ↔ IMSI في مرجع التكوين.

**لماذا نحتاج إلى تعيين MSISDN إلى IMSI؟**

يتطلب بروتوكول MAP لـ **SendRoutingInfoForSM** (SRI-for-SM) من HLR إرجاع **IMSI** المطلوب فقط SMSc في استجابته. ومع ذلك، يعرف (معرف المشترك الدولي المحمول) **MSISDN** (رقم الهاتف).

في شبكة تقليدية:

- يرسل SMSc SRI-for-SM مع MSISDN الوجهة (على سبيل المثال، "5551234567")
- يجب على HLR البحث عن المشترك في قاعدة بياناته للعثور على IMSI الخاص به
- ثم يعيد HLR IMSI في استجابة SRI-for-SM
- يستخدم SMSc هذا بعد ذلك IMSI عند إرسال MT-ForwardSM إلى MSC/VLR

**نهج OmniSS7 - IMSIs الصناعية**:

يستخدم IMSI، بدلاً من الاحتفاظ بقاعدة بيانات كاملة للمشتركين مع تعيينات MSISDN إلى IMSIs الصناعية مباشرة من MSISDN. يوفر هذا النهج OmniSS7 **لحساب** مخطط ترميز بسيط فائدتين رئيسيتين:

1. **الخصوصية**: لا يتم الكشف عن IMSIs الحقيقية للمشتركين المخزنة في قاعدة بيانات HLR في استجابات SRI-for-SM المرسلة عبر شبكة SS7
2. **البساطة**: لا حاجة للاستعلام عن قاعدة بيانات HLR لعمليات البحث عن IMSI أثناء عمليات SRI-for-SM - يتم حساب IMSI في الوقت الفعلي من MSISDN

**كيف يعمل**:

يتم ترميز MSISDNs من IMSI (الأرقام بعد MCC+MNC) مباشرة في جزء المشترك من

```
IMSI = PLMN_PREFIX + zero_padded_MSISDN
```

حيث:

- **PLMN_PREFIX**: MCC + MNC (على سبيل المثال، "001001" لشبكة الاختبار)
- **MSISDN**: جميع الأرقام من رقم الهاتف
- **Zero Padding**: يتم إضافة أصفار من اليسار لملء IMSI إلى 15 رقمًا بالضبط

**مثال خطوة بخطوة**:

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

# الإدخال: MSISDN من طلب SRI-for-SM (تم فك تشفير TBCD)
msisdn = "555123456"    # 9 أرقام

# الخطوة 1: حساب المساحة المتاحة لرقم المشترك
subscriber_digits = 15 - String.length("001001")  # = 9 أرقام

# الخطوة 2: إضافة أصفار إلى MSISDN لملء جزء المشترك
padded_msisdn = String.pad_leading("555123456", 9, "0")  # =
"555123456" (لا حاجة للإضافة)

# الخطوة 3: دمج بادئة PLMN + MSISDN المملوء
imsi = "001001" <> "555123456"  # = "001001555123456" (15 بالضبط
رقمًا)
```

**أمثلة كاملة**:

| MSISDN المدخل | بادئة PLMN | الأرقام المتاحة للمشترك | MSISDN المملوء | IMSI النهائي |
|---|---|---|---|---|
| `"555123456"` | `"001001"` (6) | 9 | `"555123456"` | `"001001555123456` |
| `"99"` | `"001001"` (6) | 9 | `"000000099"` | `"001001000000099` |
| `"999999999"` | `"001001"` (6) | 9 | `"999999999"` | `"001001999999999` |
| `"91123456789"` | `"001001"` (6) | 9 | `"555123456"` | `"001001555123456` |

**معالجة حالات الحافة**:

- **MSISDNs القصيرة**: يتم إضافة أصفار من اليسار (على سبيل المثال، `"99"` → `"000000099"`)
- **MSISDNs الطويلة**: يتم الاحتفاظ بالأرقام اليمنى، ويتم تقصير الأرقام اليسرى (على سبيل المثال، `"91123456789"` → `"555123456"`)
- **طول IMSI**: يكون IMSI دائمًا بالضبط 15 رقمًا

**عكس التعيين (IMSI → MSISDN)**:

يمكن لـ SMSc عكس هذا التعيين لتحويل IMSIs مرة أخرى إلى MSISDNs:

```
# الإدخال: IMSI من استجابة SRI-for-SM
imsi = "001001555123456"

# الخطوة 1: إزالة بادئة PLMN
plmn_prefix = "001001"
subscriber_portion = String.slice(imsi, 6, 9)  # = "555123456"

# الخطوة 2: إزالة الأصفار من اليسار للحصول على MSISDN الفعلي
msisdn = String.replace_leading(subscriber_portion, "0", "")  # =
"555123456"
```

**أمثلة على عكس التعيين**:

| المدخل IMSI | بادئة PLMN | جزء المشترك | إزالة الأصفار من اليسار | MSISDN النهائي |
|---|---|---|---|---|
| `"001001555123456"` | `"001001"` | `"555123456"` | `"555123456"` | `"55512345` |
| `"001001000000099"` | `"001001"` | `"000000099"` | `"99"` | `"99"` |
| `"001001999999999"` | `"001001"` | `"999999999"` | `"999999999"` | `"99999999` |

**خصائص هذا التعيين**:

- ✅ **تحديد**: نفس IMSI دائمًا ينتج نفس MSISDN
- ✅ **قابل للعكس**: يمكن تحويل IMSI إلى MSISDN
- ✅ **تكوين الحد الأدنى**: يتطلب فقط `hlr_imsi_plmn_prefix`
- ✅ **حماية الخصوصية**: لا يتم الكشف عن IMSIs الحقيقية
- ✅ **لا حاجة لبحث قاعدة البيانات**: حساب سريع، لا حاجة لمكالمات API
- ✅ **دائمًا 15 رقمًا**: IMSI دائمًا بالضبط 15 رقمًا

**معالجة إدخال MSISDN**:

عندما يتلقى HLR طلب SRI-for-SM، يخضع MSISDN لفك تشفير TBCD:

1. **فك تشفير TBCD**: تحويل TBCD الثنائي إلى سلسلة (قد تتضمن بادئة TON/NPI مثل "91")

2. **استخراج الأرقام**: الاحتفاظ بالأرقام فقط، وإزالة أي أحرف غير رقمية
3. **تطبيع**: إذا كانت أطول من المساحة المتاحة، يتم أخذ الأرقام اليمنى؛ إذا كانت أقصر، تتم إضافة أصفار من اليسار
4. **الترميز**: دمج بادئة PLMN + MSISDN المعياري

**اعتبارات الأمان**:

تُستخدم IMSIs الصناعية التي يتم إرجاعها في استجابات SRI-for-SM لأغراض التوجيه فقط. إنها ليست IMSIs الحقيقية المخزنة في قاعدة بيانات المشتركين في HLR. يوفر ذلك طبقة إضافية من حماية الخصوصية، حيث يتم الكشف عن IMSIs الحقيقية فقط عند الضرورة القصوى (على سبيل المثال، أثناء UpdateLocation أو SendAuthenticationInfo التي تتطلب متجهات المصادقة الحقيقية).

**تدفق الاستجابة**:

```
1. SMSc → HLR: طلب SRI-for-SM
   - MSISDN (TBCD): "91123456789" (يتضمن TON/NPI)

2. معالجة HLR:
   - فك تشفير TBCD: "91123456789"
   - استخراج الأرقام: "91123456789" (11 رقمًا)
   - ملاءمة لـ 9 أرقام: "555123456" (أقصى 9)
   - إضافة PLMN: "001001" + "555123456" = "001001555123456"
   - الحصول على GT SMSC من التكوين: "5551234567"

3. HLR → SMSc: استجابة SRI-for-SM
   - IMSI: "001001555123456" (صناعي، دائمًا 15 رقمًا)
   - 5551234567" :رقم العقدة الشبكية" (حيث يجب إرسال MT-
ForwardSM)

4. SMSc ترسل MT-ForwardSM مع "5551234567" إلى IMSI
"001001555123456"
```

**التكوين**:

تُستخدم المعلمات التالية في `runtime.exs`:

```
# بادئة PLMN: MCC (001 = شبكة اختبار) + MNC (01 = مشغل اختبار)
hlr_imsi_plmn_prefix: "001001",

# استخراج NSN (إذا كانت MSISDNs تتضمن رمز البلد)
hlr_msisdn_country_code: "1",      # تستخدم للتعيين العكسي (IMSI→MSISDN)
hlr_msisdn_nsn_offset: 1,          # تخطي 1-رقم رمز البلد
hlr_msisdn_nsn_length: 10          # استخراج 10 أرقام NSN
```

**تكوين استخراج NSN**:

إذا كانت MSISDNs تتضمن رمز البلد (على سبيل المثال، `"68988000088"` بدلاً من `"88000088"` NSN:)، يجب عليك تكوين استخراج

- **`hlr_msisdn_nsn_offset`**: الموضع الذي يبدأ فيه NSN (عادةً طول رمز البلد)
- **`hlr_msisdn_nsn_length`**: عدد الأرقام في NSN

**أمثلة**:

| المثال | رمز البلد | MSISDN المثال | nsn_offset | nsn_length | NSN المستخرج |
|---|---|---|---|---|---|
| رقم-1 CC | `"9"` | `"95551234567"` | `1` | `10` | `"555123456` |
| رقم-2 CC | `"99"` | `"99412345678"` | `2` | `9` | `"412345678` |
| رقم-3 CC | `"999"` | `"99988000088"` | `3` | `8` | `"88000088"` |

**كيف يعمل**:

1. **MSISDN → IMSI**: استخراج NSN من MSISDN، إضافة أصفار من اليسار، دمج مع بادئة PLMN

```
MSISDN: "99988000088"
NSN: String.slice("99988000088", 3, 8) = "88000088"
MSISDN (9 أرقام) "088000088" :المملوء
IMSI: "547050" + "088000088" = "547050088000088"
```

2. **IMSI → MSISDN**: إزالة بادئة PLMN، إضافة رمز البلد، إزالة الأصفار من اليسار

```
IMSI: "547050088000088"
088000088" :جزء المشترك"
88000088" :إزالة الأصفار"
MSISDN: "+999" + "88000088" = "+99988000088"
```

# متطلبات واجهة برمجة التطبيقات: لا شيء - تستخدم SRI-for-SM القيم المحسوبة والتكوين فقط. لا حاجة

# لمكالمات واجهة برمجة التطبيقات الخلفية.

## ملخص مصدر الحقول

| نوع المصدر | الوصف | الأمثلة |
|---|---|---|
| **واجهة برمجة التطبيقات OmniHSS** | بيانات ديناميكية من قاعدة بيانات المشتركين OmniHSS | IMSI، MSISDN، VLR/MSC الخادم من circuit_session |
| **runtime.exs** | معلمات تكوين OmniSS7 | `smsc_service_center_gt_address`، `camel_service_key`، `isd_network_access_mode` |
| **ثابت** | قيم ثابتة في مولد الاستجابة | SS حالة المشترك، خدمات الحامل، رموز |
| **طلب** | الحقول المستخرجة من طلب MAP الوارد | IMSI من UpdateLocation، MSISDN من SRI |
| **محسوب** | القيم المستمدة باستخدام المنطق | IMSI الصناعي في SRI-for-SM (hlr_imsi_prefix + NSN) |

## تبعيات التكوين

**مطلوب في runtime.exs**:

- `hlr_service_center_gt_address` - مستخدم في ردود UpdateLocation
- `smsc_service_center_gt_address` - مستخدم في ردود SRI-for-SM (حيث يجب توجيه MT-ForwardSM)

**اختياري في runtime.exs** (مع القيم الافتراضية):

- `camel_service_key` - الافتراضي: `110_11`
- `camel_trigger_detection_point` - الافتراضي: `:termAttemptAuthorized`
- `isd_network_access_mode` - الافتراضي: `:packetAndCircuit`
- `isd_send_ss_data` - الافتراضي: `true`
- `isd_send_call_barring` - الافتراضي: `true`
- `hlr_imsi_plmn_prefix` - الافتراضي: `001001"` (بادئة PLMN لتعيين MSISDN↔IMSI)

**مطلوب من OmniHSS**:

يجب أن توفر OmniHSS ت��نقاط نهاية واجهة برمجة التطبيق REST لـ:

- البحث عن المشترك حسب MSISDN وIMSI
- تحديث موقع الجلسة الدائرية (تعيين VLR/MSC)
- توليد متجهات المصادقة
- استعلامات حالة المشترك وملف تعريف الخدمة

---

# الوثائق ذات الصلة

**وثائق OmniSS7:**

- ← العودة إلى الوثائق الرئيسية
- دليل الميزات الشائعة
- دليل عميل MAP
- المرجع الفني
- مرجع التكوين

**وثائق OmniHSS:** لإدارة المشتركين، وتزويدهم، وتكوين المصادقة، والعمليات الإدارية، يرجى الرجوع إلى **وثائق منتج OmniHSS**. يحتوي OmniHSS على كل منطق قاعدة بيانات المشتركين، وخوارزميات المصادقة، وقواعد تزويد الخدمة، وقدرات إدارة Multi-IMSI.

---

**OmniSS7** بواسطة Omnitouch Network Services

# MAP دليل تكوين عميل

← العودة إلى الوثائق الرئيسية

يوفر هذا الدليل تكوينًا تفصيليًا لاستخدام OmniSS7 كـ **عميل** **MAP** لإرسال طلبات بروتوكول MAP إلى عناصر الشبكة.

## جدول المحتويات

1. ما هو وضع عميل MAP؟
2. تمكين وضع عميل MAP
3. عمليات MAP المتاحة
4. إرسال الطلبات عبر API
5. المقاييس والمراقبة
6. استكشاف الأخطاء وإصلاحها

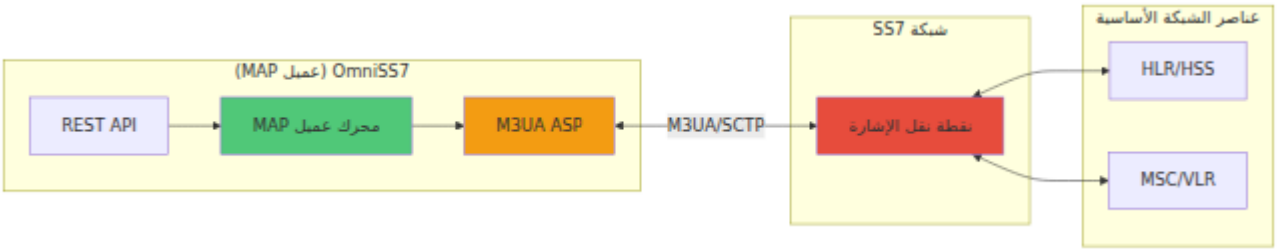
---

# ما هو وضع عميل MAP؟

**وضع عميل MAP** يسمح لـ OmniSS7 بالاتصال كـ **عملية خادم تطبيق (ASP)** إلى نظير M3UA (STP أو SGP) وإرسال/استقبال رسائل **MAP (جزء التطبيق المحمول)** لخدمات مثل:

- **استعلامات HLR**: SRI (إرسال معلومات التوجيه)، SRI-for-SM، معلومات المصادقة
- **تحديثات الموقع**: تحديث الموقع، إلغاء الموقع
- **إدارة المشتركين**: توفير رقم التجوال (PRN)، إدخال بيانات المشترك

## بنية الشبكة




---

# تمكين وضع عميل MAP

قم بتحرير `config/runtime.exs` وتكوين إعدادات عميل MAP. لمزيد من المعلومات حول التكوين الكامل، راجع معلمات اتصال في مرجع التكوين M3UA.

## التكوين الأساسي

```
config :omniss7,
  # تمكين وضع عميل MAP
  map_client_enabled: true,

  # اتصال M3UA لعميل MAP (يتصل كـ ASP إلى STP/SGP البعيد)
  map_client_m3ua: %{
    mode: "ASP",                      # وضع M3UA: "ASP" (عميل) أو
"SGP" (خادم)
    callback: {MapClient, :handle_payload, []},  # رد الاتصال
للرسائل الواردة
    process_name: :map_client_asp,    # اسم العملية المسجلة
    local_ip: {10, 0, 0, 100},       # عنوان IP المحلي
    local_port: 2905,                 # منفذ SCTP المحلي
    remote_ip: {10, 0, 0, 1},        # عنوان IP البعيد لـ STP/SGP
    remote_port: 2905,                # منفذ STP/SGP البعيد
    routing_context: 1                # سياق التوجيه M3UA
  }
```

## مثال على تكوين الإنتاج

```
config :omniss7,
  # تمكين عميل MAP للإنتاج
  map_client_enabled: true,

  # اتصال M3UA للإنتاج
  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},        # عنوان IP لـ STP للإنتاج
    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"
  }
```

---

# عمليات MAP المتاحة

## 1. إرسال معلومات التوجيه لـ SM (SRI-for-SM)

يستعلم عن HLR لتحديد MSC الخادم لتسليم SMS. لمزيد من المعلومات التفصيلية حول كيفية معالجة HLR لطلبات SRI-for-SM، راجع دليل SRI-for-SM في HLR.

**نقطة نهاية API:** `POST /api/sri-for-sm`

**الطلب:**

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

**الاستجابة:**

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

**مثال cURL:**

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

---

## 2. إرسال معلومات التوجيه (SRI)

يستعلم عن HLR الصوت مكالم��ت توجيه معلومات لـ.

**نقطة نهاية API:** `POST /api/sri`

**الطلب:**

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

**الاستجابة:**

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

---

## 3. توفير رقم التجوال (PRN)

يطلب رقم تجوال مؤقت (MSRN) من MSC الخادم.

**نقطة نهاية API:** `POST /api/prn`

**الطلب:**

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

---

## 4. إرسال معلومات المصادقة

يطلب متجهات المصادقة من HLR المشترك لمصادقة.

**نقطة نهاية API:** `POST /api/send-auth-info`

**الطلب:**

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

**الاستجابة:**

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

---

## 5. تحديث الموقع

يسجل الموقع الحالي لمشترك مع HLR. لمزيد من المعلومات التفصيلية حول معالجة UpdateLocation وتسلسلات InsertSubscriberData، راجع تحديثات الموقع في دليل HLR.

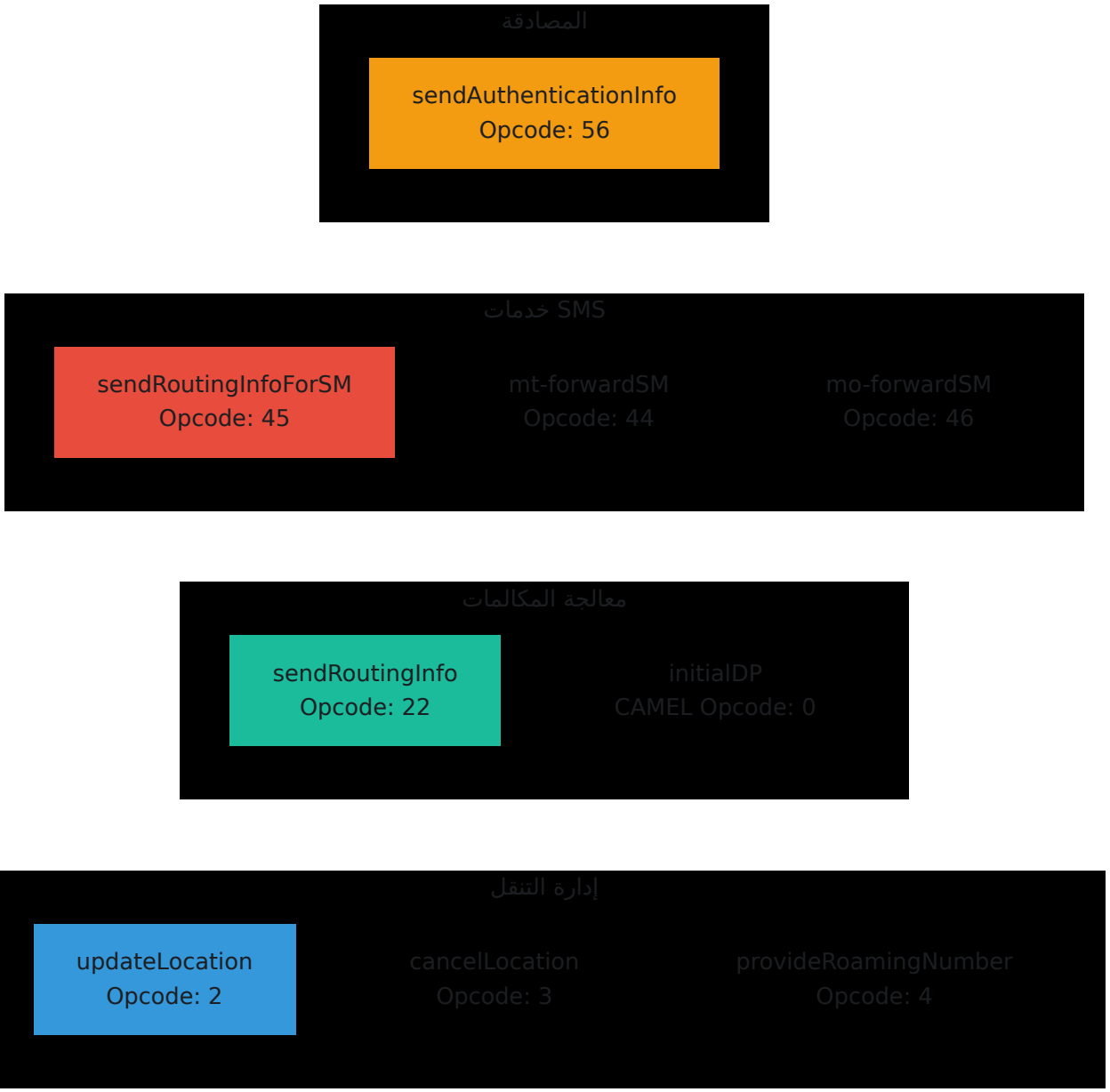
**نقطة نهاية API:** `POST /api/updateLocation`

**الطلب:**

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

---

## ملخص عمليات MAP




---

# إرسال الطلب��ت عبر API

## استخدام واجهة Swagger

توفر واجهة Swagger واجهة تفاعلية لإرسال طلبات SS7.

**الوصول إلى واجهة Swagger:**

1. انتقل إلى `http://your-server/swagger`
2. تصفح نقاط نهاية API المتاحة
3. انقر على أي نقطة نهاية لتوسيع تفاصيلها

**إرسال طلب:**

1. انقر على نقطة النهاية التي تريد استخدامها (مثل `/api/sri-for-sm`)
2. انقر على زر "جربها"
3. املأ المعلمات المطلوبة في جسم الطلب
4. انقر على "تنفيذ"
5. عرض الاستجابة أدناه

## رموز استجابة API

- **200** - نجاح، تم إرجاع النتيجة في جسم الاستجابة
- **400** - طلب غير صالح، معلمات غير صحيحة
- **504** - مهلة بوابة، لا استجابة من شبكة SS7 خلال 10 ثوانٍ

---

# مقاييس عميل MAP

## المقاييس المتاحة

**مقاييس الطلب:**

- `map_requests_total` - إجمالي عدد طلبات MAP المرسلة

    - التسميات: `operation` (القيم: `sri`, `sri_for_sm`, `prn`, `authentication_info`, إلخ)
- `map_request_errors_total` - إجمالي عدد أخطاء طلبات MAP
    - التسميات: `operation`
- `map_request_duration_milliseconds` - هيستوغرام لمدد طلبات MAP
    - التسميات: `operation`
- `map_pending_requests` - العدد الحالي لطلبات MAP المعلقة (مقياس)

## أمثلة على استعلامات Prometheus

```
# إجمالي طلبات SRI-for-SM في الساعة الماضية
increase(map_requests_total{operation="sri_for_sm"}[1h])

# متوسط وقت الاستجابة لطلبات SRI
rate(map_request_duration_milliseconds_sum{operation="sri"}[5m]) /
rate(map_request_duration_milliseconds_count{operation="sri"}[5m])

# معدل الأخطاء لجميع عمليات MAP
sum(rate(map_request_errors_total[5m])) by (operation)

# الطلبات المعلقة الحالية
map_pending_requests
```

---

# استكشاف الأخطاء وإصلاحها عميل MAP

## المشكلة: انتهاء مهلة الطلبات

**الأعراض:**

- API 504 تعيد Gateway Timeout
- لا استجابة من HLR/MSC

**التحققات:**

1. تحقق من أن اتصال M3UA نشط:

```
# في وحدة الـتحكم IEx
:sys.get_state(:map_client_asp)
```

2. تحقق من الاتصال الشبكي بـ STP
3. تحقق من سياق التوجيه وعنوان SCCP
4. تحقق من السجلا�ت بحثًا عن أخطاء SCCP

---

## المشكلة: أخطاء SCCP

**الأعراض:**

- API تعيد استجابات خطأ SCCP
- السجلات تظهر رسائل "خدمة بيانات وحدة SCCP"

**رموز خطأ SCCP الشائعة:**

- STP **لا ترجمة**: العنوان العالمي غير موجود في جدول توجيه
- غير متاح (HLR SSN 6) **فشل النظام الفرعي**: النظام الفرعي الوجهة
- **فشل الشبكة**: ازدحام الشبكة أو فشلها

**الحلول:**

- اتصل بمسؤول STP للتحقق من تكوين التوجيه
- تحقق مما إذا كان العنوان العالمي الوجهة قابلًا للوصول
- تحقق مما إذا كان النظام الفرعي الوجهة يعمل

---

# الوثائق ذات الصلة

- ← العودة إلى الوثائق الرئيسية

- واجهة الويب - دليل الميزات الشائعة، API، المراقبة
- تكوين التوجيه - دليل STP
- تسليم SMS - دليل مركز SMS
- مواصفات البروتوكول - المرجع الفني

---

**OmniSS7** بواسطة خدمات شبكة Omnitouch

# دليل تكوين مركز الرسائل القصيرة (SMSc)

← العودة إلى الوثائق الرئيسية

يوفر هذا الدليل تكوينًا مفصلًا لاستخدام OmniSS7 كواجهة أمامية لمركز الرسائل القصيرة (SMSc) مع OmniMessage كمنصة تخزين الرسائل والتسليم الخلفية.

## تكامل OmniMessage

**يعمل وضع SMSc في OmniSS7 كواجهة أمامية لإشارات SS7** تتفاعل مع **OmniMessage**، وهي منصة رسائل SMS على مستوى الناقل. تفصل هذه البنية بين الاهتمامات:

- **OmniSS7 (واجهة SMSc الأمامية)**: تتعامل مع جميع إشارات بروتوكول SS7/MAP، توجيه SCCP، والتواصل الشبكي
- **OmniMessage (الجهة الخلفية للرسائل القصيرة)**: تدير تخزين الرسائل، الانتظار، منطق إعادة المحاولة، تتبع التسليم، وقرارات التوجيه

## لماذا OmniMessage؟

يوفر OmniMessage قدرات رسائل SMS مع ميزات تشمل على مستوى الناقل:

- **إدارة قائمة الانتظار للرسائل**: تخزين دائم مع منطق إعادة المحاولة القابل للتكوين وقائمة انتظار ذات أولوية
- **تتبع التسليم**: حالة التسليم في الوقت الحقيقي، تقارير التسليم (DLR)، وتتبع أسباب الفشل
- **دعم متعدد SMSc**: يمكن لعدة مثيلات أمامية الاتصال بجهة خلفية واحدة من OmniMessage لتحقيق التوازن في الحمل والموثوقية
- **ذكاء التوجيه**: قواعد توجيه متقدمة بناءً على الوجهة، المرسل، محتوى الرسالة، ووقت اليوم
- **تحديد معدل**: تحكم لكل مسار لمنع ازدحام الشبكة (المعاملات في الثانية) TPS
- **تصميم يعتمد على API**: واجهة برمجة تطبيقات HTTP REST للتكامل مع أنظمة الفوترة، بوابات العملاء، والتطبيقات الخارجية

- **التحليلات والتقارير**: إحصائيات حجم الرسائل، معدلات نجاح التسليم، ومقاييس الأداء

يتم تخزين جميع بيانات الرسائل، حالة التسليم، وتكوينات التوجيه وإدارتها في OmniMessage. تستعلم OmniSS7 عن OmniMessage عبر استدعاءات API عبر HTTPS لاسترداد الرسائل المعلقة، تحديث حالة التسليم، والتسجيل كواجهة أمامية نشطة.

هو **واجهة أمامية للإشارات فقط**. يتم التعامل مع جميع OmniSS7 في SMSc **مهم**: وضع منطق توجيه الرسائل، إدارة قوائم الانتظار، خوارزميات إعادة المحاولة، تتبع التسليم، وقواعد الأعمال بواسطة OmniMessage. يغطي هذا الدليل تكوين بروتوكول SS7/MAP في OmniSS7. لمعلومات حول توجيه الرسائل، تكوين القوائم، تقارير التسليم، تحديد المعدل، والتحليلات، **يرجى الرجوع إلى وثائق OmniMessage**.

# جدول المحتويات


1. تكامل OmniMessage
2. ما هو وضع مركز الرسائل القصيرة؟
3. تمكين وضع SMSc
4. تكوين واجهة برمجة تطبيقات HTTP
5. تدفقات رسائل SMS
6. منع الحلقات
7. تتبع المشتركين في SMSc
8. تكوين التفريغ التلقائي
9. المقاييس والمراقبة
10. استكشاف الأخطاء وإصلاحها


---

# ما هو وضع مركز الرسائل القصيرة؟

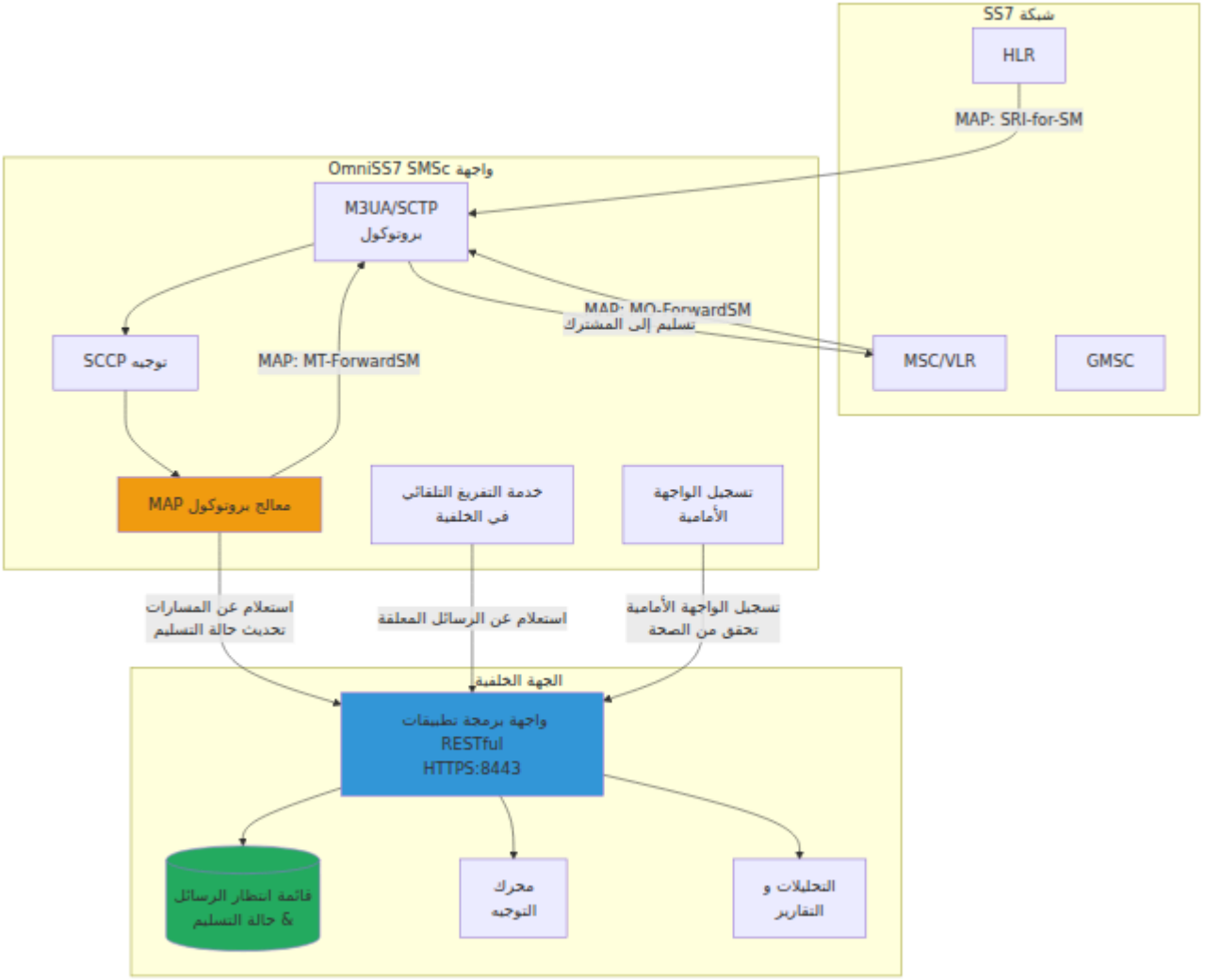
**ملاحظة**: يغطي هذا القسم تكوين إشارات SS7 في OmniSS7 فقط. لمزيد من المعلومات حول قواعد توجيه الرسائل، إدارة القوائم، تتبع التسليم، وتكوين منطق الأعمال، انظر **وثائق منتج OmniMessage**.

يمكّن **وضع مركز الرسائل القصيرة** OmniSS7 من العمل كـ SMSc لـ:

- **تسليم MT-SMS**: تسليم الرسائل القصيرة إلى المشتركين

- استقبال وتوجيه الرسائل القصيرة المنشأة من الهاتف المحمول: **معالجة MO-SMS**
- **قائمة انتظار الرسائل**: قائمة انتظار مدعومة بقاعدة بيانات مع منطق إعادة المحاولة
- **التفريغ التلقائي**: تسليم الرسائل القصيرة تلقائيًا من القائمة
- **تقارير التسليم**: تتبع حالة تسليم الرسائل

## بنية مركز الرسائل القصيرة




---

# تمكين وضع SMSc

تحتاج إلى تمكين وضع SMSc، العمل في أوضاع مختلفة. لاستخدامه كـ OmniSS7 يمكن لـ في التكوين.

## التبديل إلى وضع SMSc

يحتوي ملف `config/runtime.exs` في OmniSS7. على ثلاثة أوضاع تشغيل مُعدة مسبقًا لتمكين وضع SMSc:

1. **افتح** `config/runtime.exs`
2. **ابحث** عن أقسام التكوين الثلاثة (الأسطر 53-204):
   - وضع التكوين 1: STP (الأسطر 53-95)
   - وضع التكوين 2: HLR (الأسطر 97-142)
   - وضع التكوين 3: SMSc (الأسطر 144-204)
3. **قم بالتعليق على** أي تكوين نشط آخر (أضف `#` إلى كل سطر)
4. **قم بإلغاء تعليق** تكوين SMSc (قم بإزالة `#` من الأسطر 144-204)
5. **خصص** معلمات التكوين حسب الحاجة
6. **أعد تشغيل** التطبيق: `iex -S mix`

## تكوين وضع SMSc

يبدو تكوين SMSc الكامل كما يلي:

```
config :omniss7,
  # تمكين ميزات الوضع - علامات STP + SMSc
  # ملاحظة: map_client_enabled صحيح لأن SMSc يحتاج إلى قدرات التوجيه
  map_client_enabled: true,
  hlr_mode_enabled: false,
  smsc_mode_enabled: true,

  # تكوين واجهة برمجة تطبيقات OmniMessage الخلفية
  smsc_api_base_url: "https://10.179.3.219:8443",
  # تعريف SMSc للتسجيل مع الجهة الخلفية
  smsc_name: "ipsmgw",
  # عنوان GT لمركز الخدمة لعمليات SMS
  smsc_service_center_gt_address: "5551234567",

  # تكوين التفريغ التلقائي (معالجة قائمة انتظار الرسائل القصيرة في
(الخلفية)
  auto_flush_enabled: true,
  auto_flush_interval: 10_000,
  auto_flush_dest_smsc: "ipsmgw",
  auto_flush_tps: 10,

  # تكوين اتصال M3UA
  # الاتصال كـ ASP لإرسال/استقبال عمليات MAP SMS
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :stp_client_asp,
    # نقطة النهاية المحلية (نظام SMSc)
    local_ip: {10, 179, 4, 12},
    local_port: 2905,
    # نقطة النهاية البعيدة STP
    remote_ip: {10, 179, 4, 10},
    remote_port: 2905,
    routing_context: 1
  }

config :control_panel,
  use_additional__pages: [
    {SS7.Web.EventsLive, "/events", "أحداث SS7"},
    {SS7.Web.TestClientLive, "/client", "عميل SS7"},
    {SS7.Web.M3UAStatusLive, "/m3ua", "M3UA"},
    {SS7.Web.RoutingLive, "/routing", "التوجيه"},
    {SS7.Web.RoutingTestLive, "/routing_test", "اختبار التوجيه"},
```

```
    {SS7.Web.SmscLinksLive, "/smsc_links", "روابط SMSc"}
  ],
  page_order: ["/events", "/client", "/m3ua", "/routing",
"/routing_test", "/smsc_links", "/application", "/configuration"]
```

## معلمات التكوين للتخصيص

للحصول على مرجع كامل لجميع معلمات التكوين، انظر مرجع التكوين.

| المعلمة | النوع | الافتراضي | وصف |
|---|---|---|---|
| `smsc_api_base_url` | سلسلة | *مطلوب* | **نطة نهاية** جهة برمجة تطبيقات OmniMes |
| `smsc_name` | سلسلة | `"{hostname}_SMSc"` | SM معرف الخاص بك للتسجيل |
| `smsc_service_center_gt_address` | سلسلة | *مطلوب* | GT عنوان الخدمة |
| `auto_flush_enabled` | منطقي | `true` | كين معالجة مة التلقائية |
| `auto_flush_interval` | عدد صحيح | `10_000` | ترة معالجة ئمة بالمللي ثانية |
| `auto_flush_dest_smsc` | سلسلة | *مطلوب* | SMSc اسم جهة للتفريغ التلقائي |
| `auto_flush_tps` | عدد صحيح | `10` | دل معالجة الرسائل (المعاملات/ الثانية) |
| `local_ip` | مجموعة | *مطلوب* | م IP عنوان اص SMSc بك |
| `local_port` | عدد صحيح | `2905` | ذ المحلي لـ SCTP |

| المعلمة | النوع | الافتراضي | وصف |
|---|---|---|---|
| `remote_ip` | مجموعة | *مطلوب* | ن IP عنوان ل بـ STP بـ SS7 |
| `remote_port` | عدد صحيح | `2905` | فذ البعيد لـ SCTP |
| `routing_context` | عدد صحيح | `1` | عرف سياق التوجيه لـ M |

## SMSc ماذا يحدث عند تمكين وضع

عند تعيين `smsc_mode_enabled: true` و `map_client_enabled: true`، ستظهر واجهة الويب:

- **أحداث SS7** - تسجيل الأحداث
- **عميل SS7** - اختبار عمليات MAP
- **M3UA** - حالة الاتصال
- **التوجيه** - إدارة جدول التوجيه (تم تمكين STP)
- **اختبار التوجيه** - اختبار التوجيه (تم تمكين STP)
- **روابط SMSc** - حالة واجهة برمجة تطبيقات SMSc + إدارة SMS قائمة انتظار محدد لـ *SMSc* ←
- **الموارد** - مراقبة النظام
- **التكوين** - عارض التكوين

**روابط** سيتم إخفاء علامة التبويب **HLR**.

## ملاحظات مهمة

- يتطلب وضع SMSc `map_client_enabled: true` لتمكين قدرات التوجيه
- **الجهة الخلفية OmniMessage**: يجب أن تكون واجهة برمجة تطبيقات الخلفية متاحة على OmniMessage المكون `smsc_api_base_url`

- كل OmniMessage **5** **تسجيل الواجهة الأمامية**: يقوم النظام بالتسجيل تلقائيًا مع `SMS.FrontendRegistry` **دقائق** عبر وحدة
- **مهلة طلب API**: جميع طلبات واجهة برمجة تطبيقات OmniMessage لها **مهلة ثابتة مدتها 5 ثوانٍ**
- **مهلة طلب MAP**: جميع طلبات MAP (SRI-for-SM، MT-ForwardSM، إلخ) لها **مهلة ثابتة مدتها 10 ثوانٍ**
- تلقائيًا في الخلفية SMS يقوم التفريغ التلقائي بمعالجة قائمة انتظار
- MAP SMS لإرسال/استقبال عمليات STP مع M3UA يتطلب الاتصال بـ
- بعد تغيير الأوضاع، يجب إعادة تشغيل التطبيق لتفعيل التغييرات
- **واجهة الويب**: انظر دليل واجهة الويب لمعلومات حول استخدام واجهة الويب
- **الوصول إلى API**: انظر دليل API لوثائق واجهة برمجة تطبيقات REST والوصول إلى واجهة Swagger UI

---

# تكوين واجهة برمجة تطبيقات HTTP

## إعداد OmniMessage الخلفية

يتواصل OmniSS7 مع OmniMessage عبر واجهة برمجة تطبيقات HTTPS REST لإدارة تسليم الرسائل، تتبع حالة المشتركين، والتسجيل كواجهة أمامية نشطة:

```
config :omniss7,
  # عنوان URL الأساسي لواجهة برمجة تطبيقات OmniMessage
  smsc_api_base_url: "https://10.5.198.200:8443",
  # معرف اسم SMSc للتسجيل (يستخدم hostname_SMSc إذا كان فارغًا)
  smsc_name: "omni-smsc01",
  # عنوان GT لمركز الخدمة لعمليات SMS
  smsc_service_center_gt_address: "5551234567"
```

**معلمات التكوين:**

| المعلمة | النوع | مطلوب | الافتراضي |
| --- | --- | --- | --- |
| `smsc_api_base_url` | سلسلة | نعم | `"https://localhost` |
| `smsc_name` | سلسلة | لا | `""` (يستخدم `"{hostname}_SMSc"`) |
| `smsc_service_center_gt_address` | سلسلة | لا | `"5551234567"` |

## تسجيل الواجهة الأمامية

عند بدء التشغيل و**إعادة التسجيل كل 5 دقائق** عبر وحدة `SMS.FrontendRegistry`. يتيح ذلك لـ OmniMessage يقوم النظام بالتسجيل تلقائيًا مع OmniMessage:

- تتبع الواجهات الأمامية النشطة لتحقيق التوازن في الحمل
- مراقبة وقت التشغيل وحالة الصحة
- جمع معلومات التكوين
- إدارة توجيه SMS الموزع عبر واجهات أمامية متعددة

**تفاصيل التنفيذ:**

- **فترة التسجيل**: 5 دقائق (ثابتة)
- **العملية**: تبدأ تلقائيًا عند تعيين `smsc_mode_enabled: true`

**حمولة التسجيل:**

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

**ملاحظة**: يتم أخذ اسم الواجهة الأمامية من معلمة التكوين `smsc_name`. إذا لم يتم تعيينها، فإنها تعود إلى `"{hostname}_SMSc"`.

## اتصال واجهة برمجة تطبيقات OmniMessage

عندما يتلقى OmniSS7 عمليات MAP من شبكة SS7 أو يعالج قائمة الرسائل، يتواصل مع OmniMessage لـ:

- **التسجيل كواجهة أمامية نشطة** والإبلاغ عن حالة الصحة
- **إرسال الرسائل المنشأة من الهاتف المحمول (MO)** المستلمة من المشتركين
- **استرداد الرسائل الموجهة إلى الهاتف المحمول (MT)** من قائمة الانتظار للتسليم
- **تحديث حالة التسليم** مع تقارير النجاح/الفشل
- **استعلام معلومات التوجيه** لتوجيه الرسائل

| نقطة النهاية | الطريقة | الغرض | جسم الطلب |
| --- | --- | --- | --- |
| `/api/frontends` | POST | تسجيل مثيل الواجهة الأمامية | `{"frontend_name": "...", "frontend_type": "SMSc", "hostname": "...", "uptime_seconds": ...}` |
| `/api/messages_raw` | POST | إدراج رسالة SMS جديدة | `{"source_msisdn": "...", "source_smsc": "...", "message_body": "..."}` |
| `/api/messages` | GET | الحصول على قائمة انتظار الرسائل | رأس: `smsc: <smsc_name>` |
| `/api/messages/{id}` | PATCH | وضع علامة على الرسالة كتم تسليمها | `{"deliver_time": "...", "dest_smsc": "..."}` |
| `/api/messages/{id}` | PUT | تحديث حالة الرسالة | `{"dest_smsc": null}` |
| `/api/locations` | POST | إدراج/تحديث موقع المشترك | `{"msisdn": "...", "imsi": "...", "location": "...", "ims_capable": true, "csfb": false, "expires": "...", "user_agent": "...", "ran_location": "...", "imei": "...", "registered": "..."}` |

| نقطة النهاية | الطريقة | الغرض | جسم الطلب |
|---|---|---|---|
| `/api/events` | POST | إضافة تتبع الحدث | `{"message_id": ..., "name": "...", "description": "..."}` |
| `/api/status` | GET | تحقق من الصحة | - |

## تنسيق استجابة واجهة برمجة التطبيقات

تستخدم جميع استجابات واجهة برمجة التطبيقات تنسيق JSON التالية مع الاتفاقيات:

- **استجابات النجاح**: HTTP 200-201 مع جسم JSON يحتوي على بيانات النتيجة
- **استجابات الخطأ**: HTTP 4xx/5xx مع تفاصيل الخطأ في جسم الاستجابة
- **الطوابع الزمنية**: تنسيق ISO 8601 (مثل، `T12:34:56Z"21-10-2025"`)
- **معرفات الرسائل**: معرفات صحيحة أو سلسلة

## وحدات عميل واجهة برمجة التطبيقات

يتكون نظام SMS من ثلاث وحدات رئيسية:

### 1. SMSc.APIClient

وحدة عميل API الرئيسية التي توفر جميع اتصالات واجهة برمجة التطبيقات مع HTTP OmniMessage:

- `frontend_register/4` - تسجيل الواجهة الأمامية مع OmniMessage
- `insert_message/3` - إدراج رسالة SMS خام (مع نسخة متوافقة مع Python 3 من معلمات)
- `insert_location/9` - إدراج/تحديث بيانات موقع المشترك
- `get_message_queue/2` - استرداد الرسائل المعلقة من قائمة الانتظار
- `mark_dest_smsc/3` - وضع علامة على الرسالة كتم تسليمها أو فشلت
- `add_event/3` - إضافة تتبع الحدث للرسائل
- `flush_queue/2` - معالجة الرسائل المعلقة (SRI-for-SM + MT-forwardSM)
- `auto_flush/2` - حلقة معالجة قائمة ا��انتظار المستمرة

### 2. SMS.FrontendRegistry

يتعامل مع تسجيل الواجهة الأمامية بشكل دوري مع الجهة الخلفية:

- يسجل تلقائيًا عند بدء التشغيل
- يعيد التسجيل كل 5 دقائق
- يستخدم `smsc_name` (يعود إلى اسم المضيف) من التكوين
- يجمع معلومات التكوين ووقت التشغيل للنظام

### 3. SMS.Utils

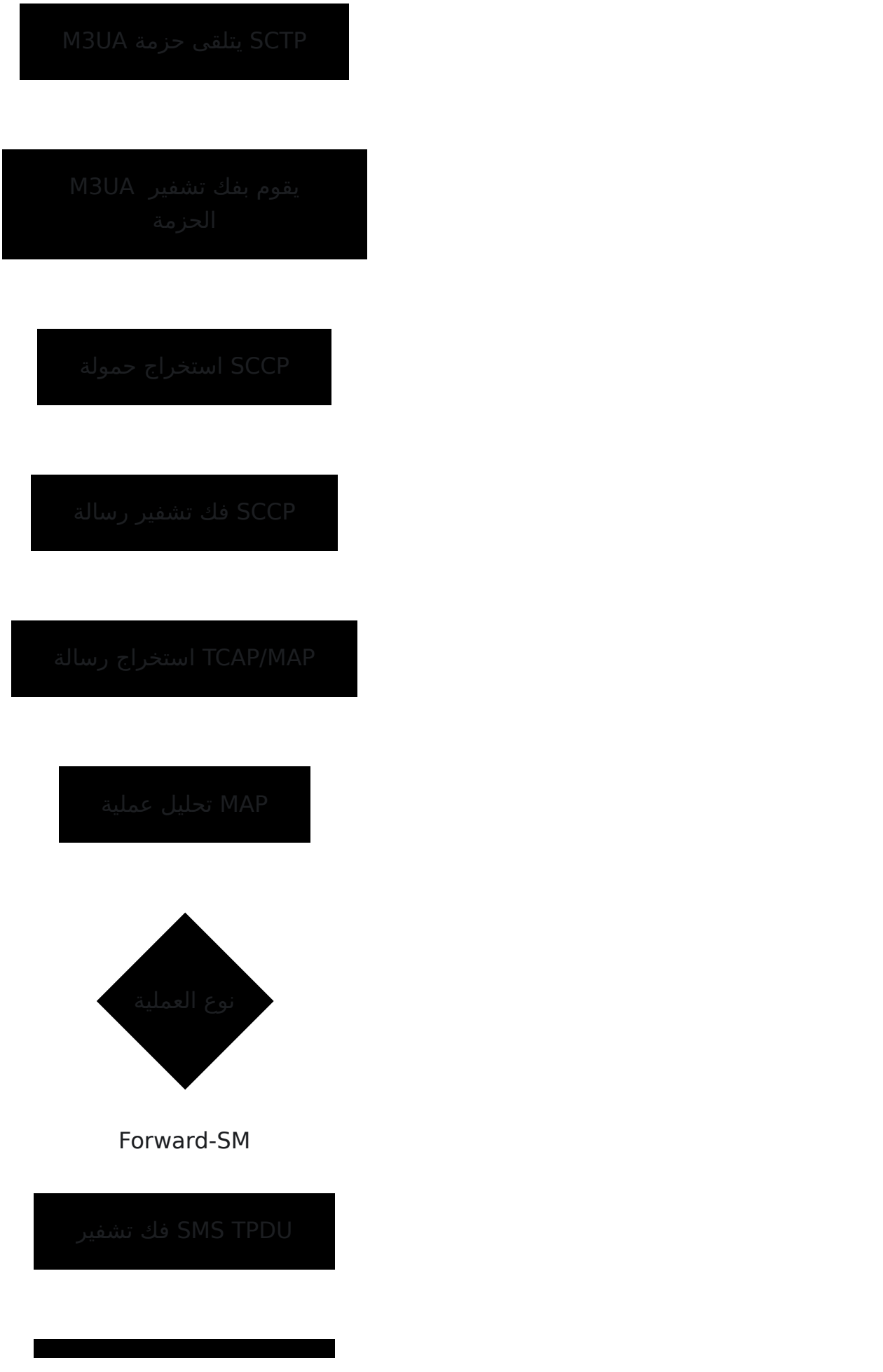
وظائف مساعدة لعمليات SMS:

- `generate_tp_scts/0` - توليد الطابع الزمني للرسائل القصيرة في تنسيق TPDU

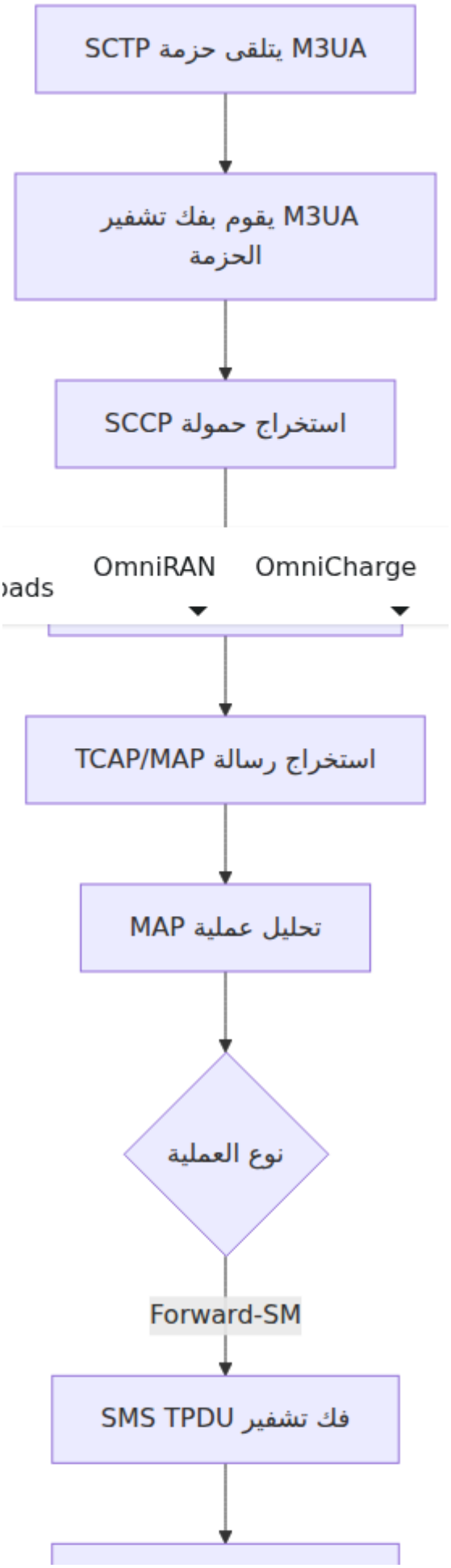
---

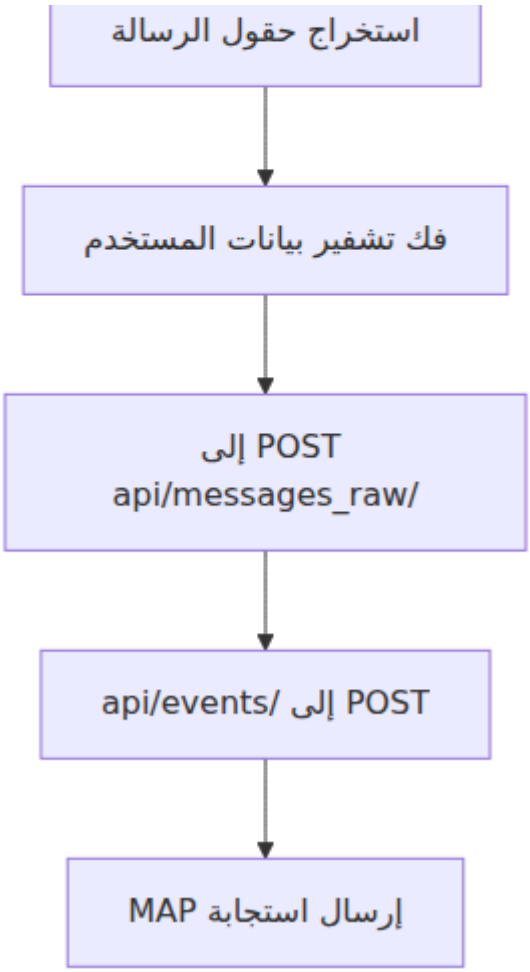
# تدفقات رسائل SMS

## تدفق SMS (منشأ من الهاتف المحمول) الوارد


SCTP يتلقى حزمة M3UA
M3UA يقوم بفك تشفير الحزمة
SCCP استخراج حمولة
SCCP فك تشفير رسالة
TCAP/MAP استخراج رسالة
MAP تحليل عملية
نوع العملية
Forward-SM
SMS TPDU فك تشفير

## الصادر (موجه إلى الهاتف المحمول) SMS تدفق

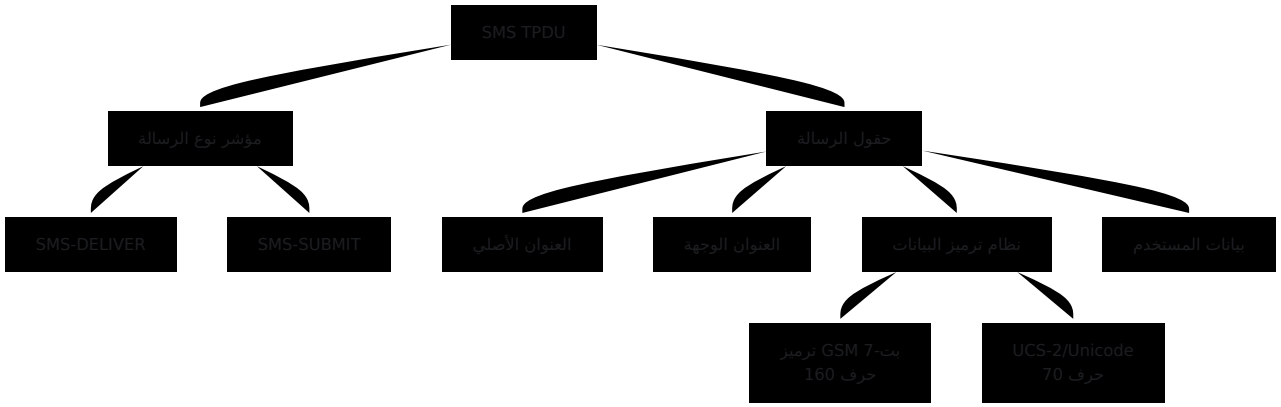

M3UA يتلقى حزمة SCTP
M3UA يقوم بفك تشفير الحزمة
استخراج حمولة SCCP
OmniRAN
OmniCharge
ads
استخراج رسالة TCAP/MAP
تحليل عملية MAP
نوع العملية
Forward-SM
فك تشفير SMS TPDU




**شرح الخطوات الرئيسية**:

- **طلب SRI-for-SM**: يستعلم SMSc عن HLR باستخدام MSISDN الوجهة لتحديد مكان توجيه رسالة SMS. يستجيب HLR بـ:
  - IMSI اصطناعي (محسوب من MSISDN للخصوصية) - انظر تعيين MSISDN ↔ IMSI
  - عنوان GT حيث يجب إرسا (رقم عقدة الشبكة) لمركز الخدمة لـ MT-ForwardSM
  - لمزيد من التفاصيل حول كيفية عمل ذلك، انظر دليل SRI-for-SM في HLR
- **طلب MT-forwardSM**: يرسل SMSc بمجرد الحصول على معلومات التوجيه، رسالة SMS إلى MSC/VLR الفعلية الذي يخدم المشترك

## هيكل SMS TPDU




---

# معالجة مركز الخدمة للإشعارات

يمكن لـ SMSc تلقي رسائل **alertServiceCenter** من HLR لتتبع حالة قابلية الوصول للمشتركين.

انظر تكامل مركز الخدمة، alertServiceCenter لرسائل HLR لمعلومات حول كيفية إرسال للإشعارات في دليل HLR.

## ما هو alertServiceCenter؟

عندما يقوم مشترك بإجراء UpdateLocation في HLR (جديد VLR/MSC، أي التسجيل مع)، بأن المشترك أصبح الآن قابلًا للوصول عن طريق إرسال رسالة SMSc إبلاغ أنظمة HLR يمكن لـ **alertServiceCenter** (رمز MAP 64).

## التكوين

يتم تكوين وقت انتهاء الموقع في HLR:

```
config :omniss7,
  # وقت انتهاء الموقع عند استلام SMSc alertServiceCenter (الافتراضي:
ساعة 48)
  hlr_alert_location_expiry_seconds: 172800
```

## السلوك

عندما يستقبل SMSc رسالة alertServiceCenter:

1. **فك تشفير MSISDN**: استخراج MSISDN الخاص بالمشترك من الرسالة (تنسيق TBCD)
2. **إزالة بادئة TON/NPI**: إزالة البادئات الشائعة مثل "19"، "11"، "91" (مثل، "123123213" → "19123123213")
3. **حساب IMSI**: توليد IMSI اصطناعي باستخدام نفس التعيين مثل SRI-for-SM
4. **POST إلى /api/location**: تحديث قاعدة بيانات الموقع بـ:
   - `msisdn`: رقم هاتف المشترك (منظف)
   - `imsi`: IMSI الاصطناعي
   - `location`: اسم SMSc (مثل، "ipsmgw")
   - `expires`: الوقت الحالي + `hlr_alert_location_expiry_seconds`
   - `csfb`: true (المشترك قابل للوصول عبر الدائرة المبدلة)
   - `ims_capable`: false (هذا تسجيل CS 2G/3G، وليس IMS/VoLTE)
   - `user_agent`: GT لـ HLR (للتتبع) الذي أرسل الإشعار
   - `ran_location`: "SS7"
5. **تتبع في SMSc Subscriber Tracker**: تسجيل المشترك مع GT لـ HLR، الحالة=نشط، عدادات الرسائل عند 0
6. **إرسال ACK**: الرد على HLR مع اعتراف alertServiceCenter

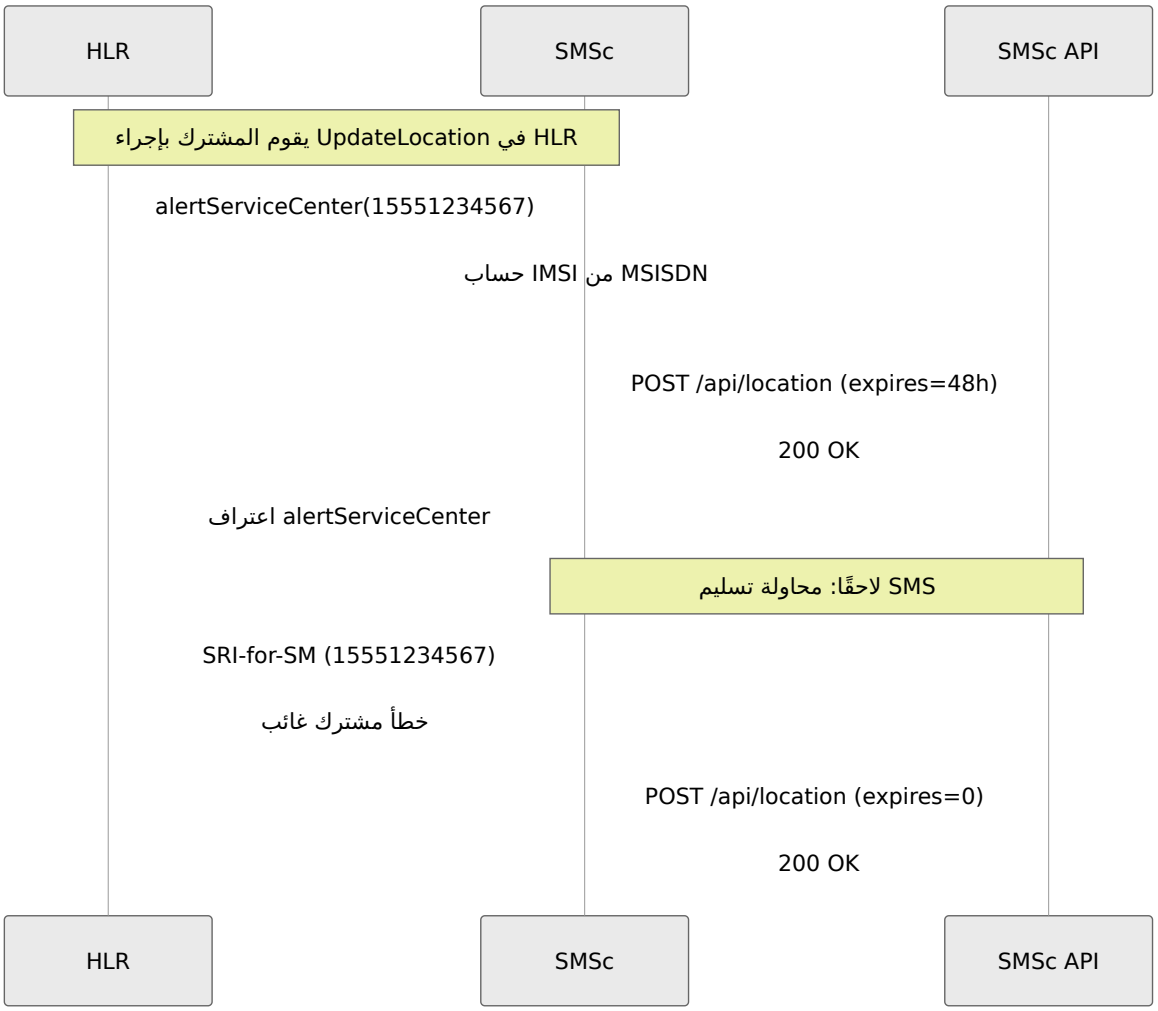
## معالجة المشترك الغائب

عندما يحاول SMSc تسليم رسالة ويتلقى خطأ "مشترك غائب" أثناء SRI-for-SM (لمزيد من المعلومات حول SRI-for-SM، انظر HLR في دليل SRI-for-SM):

1. **كشف الغياب**: تعيد SRI-for-SM خطأ `absentSubscriberDiagnosticSM`
2. **انتهاء الموقع**: POST إلى /api/location مع `expires=0` لوضع علامة على المشترك كغير قابل للوصول
3. **وكيل المستخدم**: تعيينه إلى "SS7_AbsentSubscriber" لتحديد المصدر
4. **تحديث المتتبع**: وضع علامة على المشترك كـ `failed` في SMSc Subscriber Tracker

هذا يضمن أن تعكس قاعدة بيان��ت الموقع والمتتبع بدقة حالة قابلية الوصول للمشتركين.

# مخطط التدفق




# نقطة النهاية 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"
}
```


**ملاحظة**: يحتوي حقل `user_agent` على GT لـ HLR الذي أرسل alertServiceCenter، مما يسمح لـ SMSc بتتبع HLR الذي يوفر تحديثات الموقع.

`expires` بالنسبة للمشتركين الغائبين، يتم تعيين (انتهاء فوري) إلى الوقت الحالي.

---

# منع الحلقات

**منع الحلقات التلقائي** يطبق SMSc لتجنب حلقات توجيه الرسائل اللانهائية عندما تنشأ الرسائل من شبكات SS7.

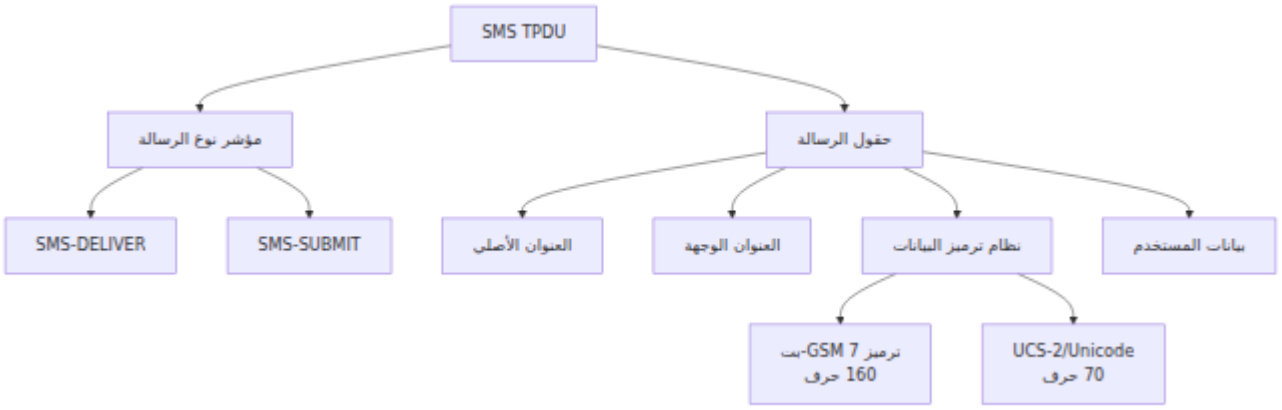
## لماذا يعتبر منع الحلقات مهمًا

عندما يستقبل SMSc رسائل SMS منشأة من الهاتف المحمول (MO) من شبكة SS7، فإنه يُدخلها في قائمة الانتظار مع حقل `source_smsc` الذي يحدد مصدرها (مثل، `"SS7_GT_15551234567"`). بدون منع الحلقات، يمكن أن تكون هذه الرسائل:

1. تحتوي على `source_smsc` "SS7" مُدرجة مع → مستلمة من شبكة SS7
2. مستردة من قائمة الانتظار → معالجة للتسليم
3. إنشاء حلقة → إرسالها مرة أخرى إلى شبكة SS7

## كيف يعمل

يكتشف SMSc تلقائيًا ويمنع الحلقات أثناء معالجة الرسائل:




## التنفيذ

من حقل SMSc عند معالجة الرسائل من قائمة الانتظار، يتحقق `source_smsc`:

- **إذا كانت `source_smsc` تحتوي على "SS7"**:
  - يتم تخطي الرسالة
  - يتم إضافة حدث: `"منع الحلقة"` مع وصف يشرح سبب التخطي
  - يتم وضع علامة على الرسالة كفاشلة عبر طلب PUT
  - يتم تسجيلها بمستوى تحذير
- **بخلاف ذلك**:
  - تتم معالجة الرسالة بشكل طبيعي
  - تستمر عمليات MT-ForwardSM و SRI-for-SM

## قيم SMSc المصدر

يمكن أن تحتوي الرسائل على قيم `source_smsc` متنوعة:

| الإجراء | قيمة المثال | المصدر |
| --- | --- | --- |
| **تم تخطيها** - منع الحلقة | `"SS7_GT_15551234567"` | شبكة SS7 (MO-FSM) |
| معالجة بشكل طبيعي | `"ipsmgw"` أو `"api_gateway"` | واجهة برمجة التطبيقات الخارجية/SMPP |
| عالجة بشكل طبيعي | `"smsc-node-01"` | أخرى SMSc |

## تتبع الأحداث

عندما يتم تخطي رسالة بسبب منع الحلقة، يتم تسجيل حدث:

```
{
  "message_id": 12345,
  "name": "منع الحلقة",
  "description": "تم تخطي الرسالة - source_smsc
'SS7_GT_15551234567' يحتوي على 'SS7'، مما يمنع حلقة الرسالة"
}
```

هذا الحدث مرئي في:

- **واجهة الويب**: صفحة أحداث SS7 (`/events`)
- **قاعدة البيانات**: جدول `events` عبر API
- **السجلات**: إدخالات سجل بمستوى تحذير

## التكوين

يتم **تمكين منع الحلقات دائمًا** ولا يمكن تعطيله. هذه ميزة أمان حرجة لمنع تعطيل الشبكة من حلقات الرسائل.

## سيناريو المثال

**السيناريو**: يقوم مشترك الهاتف المحمول بإرسال SMS عبر شبكة SS7

```
1. الهاتف المحمول → MSC/VLR → SMSc (عبر MO-ForwardSM)
2. يستقبل SMSc MO-FSM من GT 15551234567
3. يُدرج SMSc في قائمة الانتظار: source_smsc = "SS7_GT_15551234567"
4. يسترد التفريغ ⁇ التلقائي الرسالة من قائمة الانتظار
5. يكتشف SMSc "SS7" في source_smsc → تخطي
6. يتم تسجيل الحدث: "منع الحلقة"
7. يتم وضع علامة على الرسالة كفاشلة
8. لا يتم إرسال SRI-for-SM أو MT-ForwardSM (تم منع الحلقة)
```

مما قد ينشئ ،SS7 بدون منع الحلقات، ستقوم الخطوة 8 بإرسال الرسالة مرة أخرى إلى شبكة حلقة لانهائية.

---

# SMSc تتبع المشتركين في

يتضمن SMSc **متتبع المشتركين** GenServer الذي يحافظ على حالة حقيقية للمشتركين ومحاولات تسليم الرسائل alertServiceCenter استنادًا إلى رسائل.

## الغرض

يوفر المتتبع:

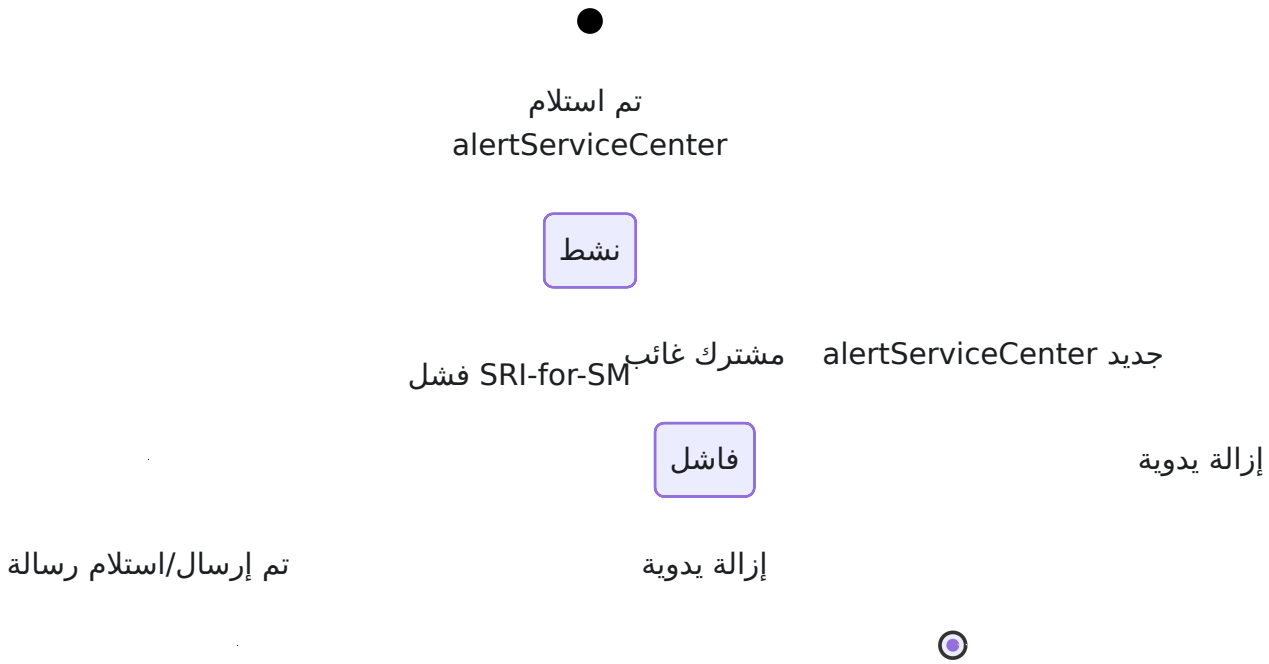
- **مراقبة قابلية الوصول**: أي المشتركين قابلون للوصول حاليًا
- **تتبع HLR**: أي HLR أرسل alertServiceCenter لكل مشترك
- **عدادات الرسائل**: عدد الرسائل المرسلة/المستلمة لكل مشترك
- **تتبع الفشل**: وضع علامة على المشتركين كفاشلين عندما تفشل محاولات التسليم
- **رؤية واجهة الويب**: لوحة معلومات حقيقية تعرض جميع المشتركين المتعقبين

## المعلومات المتعقبة

بالنسبة لكل مشترك، يخزن المتتبع:

| الحقل | الوصف | المثال |
|---|---|---|
| `msisdn` | رقم هاتف المشترك (المفتاح) | "15551234567" |
| `imsi` | IMSI الخاص بالمشترك | "001010123456789" |
| `hlr_gt` | GT لـ HLR الذي أرسل alertServiceCenter | "15551111111" |
| `messages_sent` | عدد رسائل MT-FSM المرسلة | 5 |
| `messages_received` | عدد رسائل MO-FSM المستلمة | 2 |
| `status` | `:active` أو `:failed` | `:active` |
| `updated_at` | الطابع الزمني Unix لآخر تحديث | 1730246400 |

## انتقالات الحالة




## السلوك

**عند استلام alertServiceCenter:**

- إنشاء أو تحديث إدخال المشترك
- تعيين `status = :active`
- تسجيل GT لـ HLR
- إعادة تعي��ن أو الحفاظ على عدادات الرسائل

**عند نجاح SRI-for-SM:**

- زيادة عداد `messages_sent`
- تحديث الطابع الزمني `updated_at`

**عند فشل SRI-for-SM:**

- تعيين `status = :failed`
- الاحتفاظ به في المتتبع للمراقبة

**عند إزالة المشترك:**

- الحذف من جدول ETS
- لم يعد يظهر في واجهة الويب

## SMSc واجهة الويب - صفحة مشتركي

**المسار**: `/smsc_subscribers` :**التحديث التلقائي**: كل 2 ثانية

**ملاحظة**: تتوفر هذه الصفحة فقط عند التشغيل في وضع SMSc. بعد إلغاء تعليق تكوين SMSc في `config/runtime.exs`، يجب إعادة تشغيل التطبيق لتصبح المسار متاحًا.

توفر صفحة **مشتركي** **SMSc** مراقبة حقيقية لجميع المشتركين المتعقبين:

**الميزات**

1. **جدول المشتركين**
    - MSISDN، IMSI، HLR GT
    - عدادات الرسائل المرسلة/المستلمة
    - شارة الحالة (نشطة/فاشلة) مع ترميز اللون
    - الطابع الزمني لآخر تحديث ومدة
    - زر إزالة للمشتركين الفرديين

2. **إحصائيات ملخصة**

   - إجما��ي المشتركين المتعقبين
   - عدد المشتركين النشطين
   - عدد المشتركين الفاشلين
   - عدد HLRs الفريدة

3. **الإجراءات**

   - مسح الكل: إزالة جميع المشتركين المتعقبين
   - إزالة: إزالة مشترك فردي

**عرض المثال**

```
┌──────────────────────────────────────────────────────────────────────────┐
│ مشتركو SMSc 3 :الإجمالي                                المتعقبون        │
├──────────────────────────────────────────────────────────────────────────┤
│ MSISDN          IMSI              HLR GT          Msgs  الحالة          │
│                                                    S/R                   │
├──────────────────────────────────────────────────────────────────────────┤
│ 15551234567     001010123456789   15551111111     5/2   ● نشط          │
│ 15559876543     001010987654321   15551111111     0/0   ● نشط          │
│ 15551112222     001010111222233   15552222222     3/1   ○ فاشل         │
└──────────────────────────────────────────────────────────────────────────┘

الفريدة: 2 HLRs  |  الملخص: الإجمالي: 3  |  النشط: 2  |  الفاشل: 1
```

# وظائف API

يقدم المتتبع هذه الوظائف للوصول البرمجي:

```
# يتم استدعاؤه عند استلام alertServiceCenter
SMSc.SubscriberTracker.alert_received(msisdn, imsi, hlr_gt)

# زيادة عدادات الرسائل
SMSc.SubscriberTracker.message_sent(msisdn)
SMSc.SubscriberTracker.message_received(msisdn)

# وضع علامة على الفشل (فشل SRI-for-SM)
SMSc.SubscriberTracker.mark_failed(msisdn)

# الإزالة من التتبع
SMSc.SubscriberTracker.remove_subscriber(msisdn)

# وظائف الاستعلام
SMSc.SubscriberTracker.get_active_subscribers()
SMSc.SubscriberTracker.get_subscriber(msisdn)
SMSc.SubscriberTracker.count_subscribers()
SMSc.SubscriberTracker.clear_all()
```

## التكامل

يتم دمج المتتبع تلقائيًا مع:

- **معالج alertServiceCenter**: يستدعي `alert_received/3` عند تحديث الموقع بنجاح
- **معالج SRI-for-SM**: يزيد `messages_sent` عند التوجيه الناجح
- **معالج المشترك الغائب**: يستدعي `mark_failed/1` عندما يكون المشترك غائبًا
- **أخطاء المشترك غير المعروف**: يستدعي `mark_failed/1` عندما تفشل SRI-for-SM

---

# تفريغ قائمة انتظار SMS التلقائي

تقوم خدمة **التفريغ التلقائي** بمعالجة الرسائل القصيرة المعلقة تلقائيًا.

للحصول على مرجع معلمات التكوين، انظر تكوين التفريغ التلقائي في مرجع التكوين.

## التكوين

```
config :omniss7,
  auto_flush_enabled: true,           # تمكين/تعطيل التفريغ
التلقائي
  auto_flush_interval: 10_000,        # فترة الاستعلام بالمللي ثانية
  auto_flush_dest_smsc: nil,          # تصفية: nil = الكل
  auto_flush_tps: 10                  # الحد الأقصى للمعاملات في
الثانية
```

## كيف يعمل

1. **الاستعلام**: كل `auto_flush_interval` مللي ثانية، يستعلم عن API للرسائل المعلقة
2. **التصفية**: تصفية اختيارية حسب `auto_flush_dest_smsc`
3. **تحديد المعدل**: معالجة ما يصل إلى `auto_flush_tps` من الرسائل في كل دورة
4. **التسليم**: لكل رسالة:
   - إرسال **SRI-for-SM** إلى HLR (إرسال معلومات التوجيه للرسالة القصيرة) للحصول على معلومات التوجيه
     - يعيد HLR IMSI اصطناعي محسوب من MSISDN
     - يعيد HLR عنوان GT لمركز الخدمة حيث يجب إرسال MT-ForwardSM
     - للحصول على الوثائق HLR في دليل SRI-for-SM انظر تفاصيل الكاملة
   - عند النجاح، إرسال **MT-forwardSM** إلى MSC/VLR
   - تحديث حالة الرسالة عبر API (تم التسليم/فشل)
   - إضافة تتبع الحدث عبر API

> **الغوص الفني العميق**: للحصول على شرح كامل حول كيفية عمل SRI-for-SM، بما في ذلك تعيين MSISDN إلى IMSI، وتوليد GT لمركز الخدمة، تكوين عنوان IMSI الاصطناعي HLR في دليل تكوين SRI-for-SM الذي يحافظ على الخصوصية، انظر قسم.

---

# مقاييس SMSc

## المقاييس المتاحة

**مقاييس قائمة انتظار SMS:**

- `smsc_queue_depth` - العدد الحالي للرسائل المعلقة
- `smsc_messages_delivered_total` - إجمالي الرسائل التي تم تسليمها بنجاح
- `smsc_messages_failed_total` - إجمالي الرسائل التي فشلت في التسليم
- `smsc_delivery_duration_milliseconds` - هيستوجرام لأوقات التسليم

**استعلامات المثال:**

```
# عمق القائمة الحالي
smsc_queue_depth

# معدل نجاح التسليم (آخر 5 دقائق)
rate(smsc_messages_delivered_total[5m]) /
(rate(smsc_messages_delivered_total[5m]) +
rate(smsc_messages_failed_total[5m]))

# متوسط وقت التسليم
rate(smsc_delivery_duration_milliseconds_sum[5m]) /
rate(smsc_delivery_duration_milliseconds_count[5m])
```

---

# استكشاف الأخطاء وإصلاحها في SMSc

## المشكلة: عدم تسليم الرسائل

**التحققات:**

1. تحقق من تمكين التفريغ التلقائي
2. تحقق من اتصال قاعدة البيانات
3. راقب السجلات بحثًا عن الأخطاء
4. تحقق من أن اتصال M3UA نشط

5. تحقق من حدود TPS

## لمشكلة: ارتفاع عمق القائمة��

**الأسباب المحتملة:**

- حد TPS منخفض جدًا
- مشكلات مهلة HLR
- مشكلات في الاتصال بالشبكة
- أرقام وجهة غير صالحة

**الحلول:**

- زيادة `auto_flush_tps`
- تحقق من توفر HLR
- مراجعة سجلات الرسائل الفاشلة

---

# واجهة برمجة تطبيقات MT-forwardSM

## إرسال SMS عبر API

**نقطة النهاية API:** `POST /api/MT-forwardSM`

**الطلب:**

```
{
  "imsi": "234509876543210",
  "destination_serviceCentre": "447999555111",
  "originating_serviceCenter": "447999123456",
  "smsPDU":
"040B917477218345F60000157030185714 0C0BD4F29C0E9281C4E1F11A"
}
```

**الاستجابة:**

```
{
  "result": "success",
  "message_id": "12345"
}
```

---

# الوثائق ذات الصلة

**وثائق OmniSS7:**

- ← العودة إلى الوثائق الرئيسية
- دليل تكوين HLR - إعداد وضع HLR والعمليات
  - **تفاصيل فنية SRI-for-SM** - الوثائق الكاملة حول تعيين MSISDN إلى IMSI وتكوين مركز الخدمة
- واجهة الويب - دليل الميزات الشائعة، API، المراقبة
- دليل عميل MAP - عمليات MAP
- المرجع الفني - مواصفات البروتوكول

**وثائق OmniMessage:** للحصول على تكوين توجيه الرسائل، إدارة القوائم، تتبع التسليم، تحديد المعدل، والتحليلات، يرجى الرجوع إلى **وثائق منتج OmniMessage**. يحتوي OmniMessage على جميع منطق توجيه الرسائل، خوارزميات إعادة المحاولة للقوائم، معالجة تقارير التسليم، ومحرك قواعد الأعمال.

---

**OmniSS7** بواسطة Omnitouch Network Services

# دليل تكوين M3UA STP

← العودة إلى الوثائق الرئيسية

يوفر هذا الدليل تكوينًا تفصيليًا لاستخدام OmniSS7 كنقطة نقل إشارات **(STP)**.

## جدول المحتويات


1. ما هي نقطة نقل الإشارات (STP)؟
2. أدوار شبكة STP
3. تمكين وضع STP
4. تكوين الأقران
5. دعم بروتوكول M2PA
    - M3UA مقابل M2PA
    - تكوين أقران M2PA
    - إدارة M2PA عبر واجهة الويب
    - مقاييس M2PA
6. توجيه رمز النقطة
7. توجيه العنوان العالمي
8. ميزات إدارة المسار
    - تعطيل المسارات
    - DROP Routes - منع حلقات التوجيه
9. التوجيه المتقدم
10. اختبار التكوين
11. المقاييس والمراقبة
12. مراقبة أقران M3UA


---

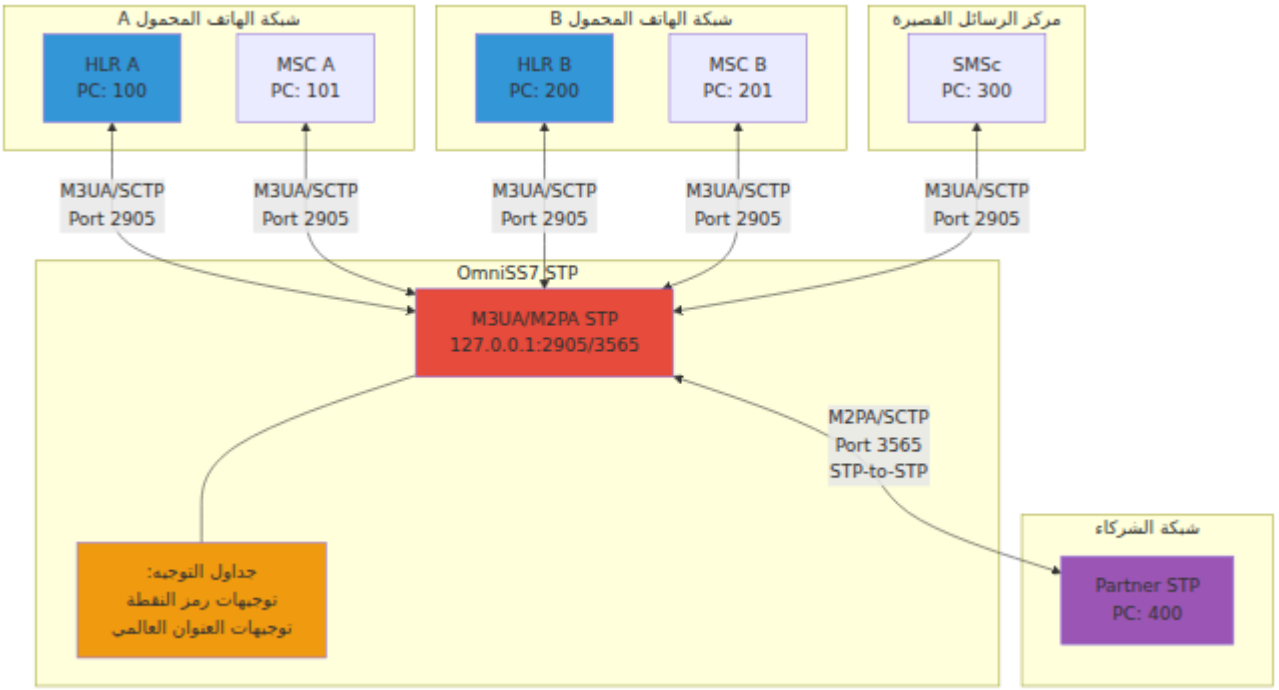
## ما هي نقطة نقل الإشارات (STP)؟

نقطة نقل الإشارات **(STP)** هي عنصر حيوي في الشبكة في شبكات SS7 والإشارات المعتمدة على IP التي تقوم بتوجيه رسائل الإشارات بين عقد الشبكة.

## وظائف STP

- **توجيه الرسائل** SS7 بناءً على رمز النقطة الوجهة (PC) أو العنوان العالمي (GT): توجيه حركة مرور إشارات
- **ترجمة البروتوكولات** SS7 بشبكات M3UA/SCTP المعتمدة على IP: ربط الشبكات التقليدية
- **توزيع الحمل**: توزيع حركة المرور عبر وجهات متعددة باستخدام التوجيه القائم على الأولوية
- **بوابة الشبكة**: ربط الشبكات المختلفة للإشارات ومزودي الخدمة
- **إخفاء الطوبولوجيا**: يمكنه إعادة كتابة العناوين لإخفاء الطوبولوجيا الداخلية للشبكة

## مخطط شبكة STP




---

# أدوار شبكة STP الموضحة

## ASP (عملية خادم التطبيق)

- **الدور**: عميل يتصل بـ SGP/STP عن بُعد
- **الاتجاه**: اتصال خارجي
- **حالة الاستخدام**: يتصل STP الخاص بك بـ STP لشبكة الشركاء

## SGP (عملية بوابة الإشارات)

- **الدور**: خادم يقبل الاتصالات من ASPs
- **الاتجاه**: اتصال داخلي
- **حالة الاستخدام**: تتصل الشبكات الشريكة بـ STP الخاص بك

## AS (خادم التطبيق)

- **التعريف**: مجموعة منطقية من واحد أو أكثر من ASPs

- **الغرض**: يوفر التكرار ومشاركة الحمل
- **حالة الاستخدام**: عدة ASPs تخدم نفس الوجهة

---

# تمكين وضع M3UA STP

يمكن أن يعمل OmniSS7 في أوضاع مختلفة. لاستخدامه كنقطة نقل إشارات، تحتاج إلى تمكين وضع STP في التكوين.

## التبديل إلى وضع STP

يحتوي ملف `config/runtime.exs` الخاص بـ OmniSS7 على ثلاثة أوضاع تشغيل مُعدة مسبقًا. لتمكين وضع STP:

1. **فتح** `config/runtime.exs`
2. **البحث عن** الأقسام الثلاثة للتكوين (الأسطر 53-174):
   - وضع التكوين 1: STP (الأسطر 53-85)
   - وضع التكوين 2: HLR (الأسطر 87-123)
   - وضع التكوين 3: SMSc (الأسطر 125-174)
3. **تعليق** على التكوين النشط حاليًا (أضف `#` إلى كل سطر)
4. **إلغاء تعليق** تكوين STP (قم بإزالة `#` من الأسطر 53-85)
5. **تخصيص** معلمات التكوين حسب الحاجة
6. **إعادة تشغيل** التطبيق: `iex -S mix`

## تكوين وضع STP

يبدو تكوين STP الكامل كما يلي:

```
config :omniss7,
  # تمكين ميزات الوضع - علامات STP فقط
  map_client_enabled: true,
  hlr_mode_enabled: false,
  smsc_mode_enabled: false,

  # تكوين اتصال M3UA
  # الاتصال كـ ASP (عملية خادم التطبيق) إلى STP/SGW عن بُعد
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :stp_client_asp,
    # نقطة النهاية المحلية (هذا النظام)
    local_ip: {10, 179, 4, 10},
    local_port: 2905,
    # نقطة النهاية STP/SGW عن بُعد
    remote_ip: {10, 179, 4, 11},
    remote_port: 2905,
    routing_context: 1
  }
```

## معلمات التكوين للتخصيص

للحصول على مرجع كامل لجميع معلمات التكوين، انظر مرجع التكوين.

| المعلمة | النوع | الافتراضي | الوصف | المثال |
|---|---|---|---|---|
| `map_client_enabled` | Boolean | `true` | تمكين عميل MAP وقدرات التوجيه | `true` |
| `local_ip` | Tuple or List | *مطلوب* | عنوان IP الخاص بنظامك. فردي: `{10, 0, 0, 1}` أو قائمة للتعدد المنزلي: `[{10, 0, 0, 1}, {10, 0, 0, 2}]` | `{10, 179, 4, 10}` |
| `local_port` | Integer | `2905` | منفذ SCTP المحلي | `2905` |
| `remote_ip` | Tuple or List | *مطلوب* | عنوان IP الخاص بـ STP/SGW عن بُعد. فردي أو قائمة للتعدد المنزلي | `{10, 179, 4, 11}` |
| `remote_port` | Integer | `2905` | منفذ SCTP عن بُعد | `2905` |
| `routing_context` | Integer | `1` | معرف سياق توجيه M3UA | `1` |
| `enable_gt_routing` | Boolean | `false` | تمكين توجيه العنوان العالمي (بالإضافة إلى توجيه PC) | `true` |

> **نصيحة**: استخدم التعدد المنزلي SCTP من خلال توفير قائمة من عناوين IP لـ `local_ip` و/أو `remote_ip` لتمكين الفشل التلقائي. انظر دليل التعدد المنزلي SCTP.

## ماذا يحدث عند تمكين وضع STP

عند `map_client_enabled: true`، ستظهر واجهة الويب:

- **SS7 أحداث** - تسجيل الأحداث

- 📞 **عميل SS7** - اختبار تشغيل MAP
- 🔗 **M3UA** - حالة الاتصال
- 🔀 *STP* **التوجيه** - إدارة جدول التوجيه ← محدد لـ
- 🧪 *STP* **اختبار التوجيه** - اختبار التوجيه ← محدد لـ
- 📊 **الموارد** - مراقبة النظام
- ⚙️ **التكوين** - عارض التكوين

**روابط** ستكون علامات التبويب **روابط HLR** و **SMSc** مخفية.

## ملاحظات هامة

- يجب السماح لبروتوكول SCTP (بروتوكول IP 132) عبر جدران الحماية
- المنفذ الافتراضي لـ M3UA هو 2905 (معيار الصناعة)
- تأكد من توفر موارد كافية في النظام للتعامل مع حركة مرور التوجيه
- **استمرارية التوجيه**: يتم تخزين جميع المسارات المكونة عبر واجهة الويب أو API وتبقى **بعد إعادة التشغيل** في **قاعدة بيانات Mnesia**
- **دمج التكوين**: يتم تحميل المسارات من `runtime.exs` عند بدء التشغيل ودمجها مع مسارات Mnesia
- بعد تغيير الأوضاع، يجب إعادة تشغيل التطبيق لتفعيل التغييرات
- **واجهة الويب**: انظر دليل واجهة الويب لإدارة المسارات عبر واجهة الويب
- **الوصول إلى API**: انظر دليل API لوثائق REST API والوصول إلى واجهة Swagger

---

# وضع STP المستقل

يمكنك تشغيل `map_client_enabled: true`، بالإضافة إلى قدرات توجيه STP المتاحة عند **خادم STP M3UA مستقل** يستمع للاتصالات الواردة.

## تمكين STP المستقل

أضف هذا التكوين إلى `config/runtime.exs`:

```
config :omniss7,
  m3ua_stp: %{
    enabled: true,
    local_ip: {127, 0, 0, 1},    # عنوان IP للاستماع
    local_port: 2905,             # المنفذ للاستماع
    point_code: 100               # رمز النقطة الخاص بـ STP
  }
```

## معلمات تكوين STP

| المعلمة | النوع | لافتراضي�? | الوصف | المثال |
|---|---|---|---|---|
| `enabled` | Boolean | `false` | تمكين خادم STP المستقل | `true` |
| `local_ip` | Tuple | `{127, 0, 0, 1}` | عنوان IP للاستماع للاتصالات | `{0, 0, 0, 0}` |
| `local_port` | Integer | `2905` | المنفذ للاستماع | `2905` |
| `point_code` | Integer | *مطلوب* | رمز النقطة الخاص بـ STP لهذا SS7 | `100` |

## متى تستخدم STP المستقل

- **التوجيه النقي** M3UA: عندما تحتاج فقط إلى توجيه MAP دون وظيفة عميل
- **STP مركزي**: لإنشاء جهاز توجيه إشارات مركزي لعناصر الشبكة المتعددة
- **معمارية المحور**: ربط عدة SMSCs و MSCs و HLRs من خلال STP مركزي

**ملاحظة**: يمكنك تمكين كل من `m3ua_stp` و `map_client_m3ua` في نفس الوقت إذا كنت بحاجة إلى كل من الاتصالات الخارجية ووظيفة STP الواردة.

---

# استمرارية جدول التوجيه (Mnesia)

يتم تخزين جميع جداول التوجيه (الأقران، مسارات رمز النقطة، ومسارات العنوان العالمي) في **قاعدة بيانات Mnesia** من أجل الاستمرارية.

## كيف يعمل التوجيه

1. **مسارات runtime.exs**: يتم تحميل المسارات المعرفة في `config/runtime.exs` تحت `m3ua_peers` و `m3ua_routes` و `m3ua_gt_routes` عند بدء تشغيل التطبيق
2. **مسارات واجهة الويب**: يتم تخزين المسارات المضافة عبر صفحة توجيه واجهة الويب في Mnesia
3. **دمج المسارات**: عند إعادة التشغيل، يتم دمج مسارات runtime.exs مع المسارات الموجودة في Mnesia (بدون تكرارات)
4. **الاستمرارية**: جميع المسارات المكونة عبر واجهة الويب **تبقى بعد إعادة تشغيل التطبيق**

## نوع تخزين Mnesia

تحكم في كيفية تخزين جداول التوجيه. لمزيد من التفاصيل حول تكوين قاعدة البيانات، انظر معلمات قاعدة البيانات في مرجع التكوين.

```
config :omniss7,
  mnesia_storage_type: :disc_copies  # أو :ram_copies للاختبار
```

| نوع التخزين | الوصف | الاستمرارية | حالة الاستخدام |
|---|---|---|---|
| `:disc_copies` | تخزين مدعوم بالقرص (افتراضي) | **تبقى بعد إعادة التشغيل** | بيئات الإنتاج |
| `:ram_copies` | في الذاكرة فقط | تفقد عند إعادة التشغيل | الاختبار، التطوير |

**افتراضي**: `:disc_copies`

## موقع قاعدة بيانات Mnesia

تقوم Mnesia بتخزين جداول التوجيه في دليل Mnesia الخاص بالتطبيق:

- **الموقع**: `Mnesia.{node_name}/` (على سبيل المثال، `Mnesia.nonode@nohost/`)
- **الجداول**: `m3ua_peer`، `m3ua_route`، `m3ua_gt_route`

## إدارة المسارات

لديك ثلاث خيارات لإدارة المسارات:

1. **runtime.exs** - تكوين ثابت يتم تحميله عند بدء التشغيل
2. **واجهة الويب** - إدارة تفاعلية للمسار (انظر دليل واجهة الويب)
3. **REST API** - إدارة المسار برمجيًا (انظر دليل API)

**أفضل الممارسات**: استخدم `runtime.exs` للتكوين الأساسي وواجهة الويب لتغييرات المسار الديناميكية أثناء التشغيل.

---

# تكوين أقران M3UA

يمثل الأقران نقاط نهاية اتصال M3UA (STPs الأخرى، HLRs، MSCs، SMSCs). أضف الأقران إلى `config/runtime.exs`.

## مثال على تكوين نظير

```
config :omniss7,
  m3ua_peers: [
    # اتصال خارجي بـ STP الشريك (الدور: :client)
    %{
      peer_id: 1,                          # معرف فريد
      name: "Partner_STP_West",            # اسم وصفي
      role: :client,                       # :client للاتصال
الخارجي، :server للاتصال الداخلي
      local_ip: {10, 0, 0, 1},            # IP المحلي للربط
      local_port: 0,                       # 0 = تعيين منفذ
ديناميكي
      remote_ip: {10, 0, 0, 10},          # IP النظير البعيد
      remote_port: 2905,                   # منفذ النظير البعيد
      routing_context: 1,                  # سياق توجيه M3UA
      point_code: 100,                     # رمز النقطة لهذا
النظير
      network_indicator: :international    # :international أو
:national
    },

    # اتصال بـ HLR المحلي (الدور: :client)
    %{
      peer_id: 2,
      name: "Local_HLR",
      role: :client,
      local_ip: {10, 0, 0, 1},
      local_port: 0,
      remote_ip: {10, 0, 0, 20},
      remote_port: 2905,
      routing_context: 2,
      point_code: 200,
      network_indicator: :international
    },

    # اتصال داخلي من MSC البعيد (الدور: :server)
    # الاتصال الوارد STP ينتظر، بالنسبة للدور :server
    %{
      peer_id: 3,
      name: "Remote_MSC",
      role: :server,                       # قبول الاتصال الوارد
      remote_ip: {10, 0, 0, 30},          # IP المصدر المتوقع
```

```
      remote_port: 2905,                    # منفذ المصدر المتوقع
(0 = قبول من أي منفذ)
      routing_context: 3,
      point_code: 300,
      network_indicator: :international
    },

    # اتصال داخلي مع منفذ مصدر ديناميكي (بدون تصفية المنفذ)
    %{
      peer_id: 4,
      name: "Dynamic_Client",
      role: :server,
      remote_ip: {10, 0, 0, 40},           # IP المصدر المتوقع
      remote_port: 0,                       # 0 = قبول الاتصالات من
أي منفذ مصدر
      routing_context: 4,
      point_code: 400,
      network_indicator: :international
    }
  ]
```

## معلمات تكوين النظير

| المعلمة | النوع | مطلوب | الو��ف |
|---|---|---|---|
| `peer_id` | Integer | نعم | معرف رقمي فريد للنظير |
| `name` | String | نعم | اسم قابل للقراءة البشرية للسجلات والمراقبة |
| `role` | Atom | نعم | `:server` أو (خارجي) `:client` (داخلي) |
| `local_ip` | Tuple or List | نعم (عميل) | عنوان IP المحلي للربط. فردي: `{10, 0, 0, 1}` أو متعدد للتعدد المنزلي SCTP: `[{10, 0, 0, 1}, {10, 0, 0, 2}]` |
| `local_port` | Integer | نعم (عميل) | المنفذ المحلي (0 للديناميكي) |
| `remote_ip` | Tuple or List | نعم | عنوان IP للنظير البعيد. فردي: `{10, 0, 0, 10}` أو متعدد: `[{10, 0, 0, 10}, {10, 0, 0, 11}]` |
| `remote_port` | Integer | نعم | منفذ النظير البعيد (0 للداخلي = قبول من أي منفذ مصدر) |
| `routing_context` | Integer | نعم | معرف سياق توجيه M3UA |
| `point_code` | Integer | نعم | رمز النقطة SS7 لهذا النظير |
| `network_indicator` | Atom | لا | `:national` أو `:international` |

> **التعدد المنزلي SCTP**: لكل من IP من أجل استمرارية الشبكة، يمكنك تكوين عدة عناوين `local_ip` و `remote_ip`. هذا يمكّن الفشل الت��قائي إذا فشل أحد مسارات الشبكة. انظر دليل التعدد المنزلي SCTP للحصول على أمثلة تكوين مفصلة وأفضل الممارسات.

## تصفية منفذ المصدر للاتصالات الواردة

في تصفية منفذ `remote_port` تتحكم معلمة، (الدور: `:server`) بالنسبة لـ **الاتصالات الواردة** المصدر:

- قبول الاتصالات فقط من (**منفذ محدد**، على سبيل المثال `remote_port: 2905`): ذلك المنفذ المصدر بالضبط
  - يوفر أمانًا إضافيًا من خلال التحقق من منفذ المصدر
  - استخدم عندما يستخدم النظير البعيد منفذ مصدر ثابت
- قبول الاتصالات من أي منفذ مصدر (**أي منفذ** `remote_port: 0`):
  - مفيد عندما يستخدم النظير البعيد منافذ مصدر ديناميكية/عابرة
  - يتحقق فقط من عنوان IP المصدر
  - أكثر مرونة ولكن أقل أمانًا قليلاً

**مثال:**

```
# قبول فقط من 10.5.198.200:2905 (منفذ محدد)
%{
  peer_id: 1,
  name: "Strict_Peer",
  role: :server,
  remote_ip: {10, 5, 198, 200},
  remote_port: 2905,
  # ... تكوين آخر
}

# قبول من 10.5.198.200 مع أي منفذ مصدر
%{
  peer_id: 2,
  name: "Flexible_Peer",
  role: :server,
  remote_ip: {10, 5, 198, 200},
  remote_port: 0,  # قبول من أي منفذ مصدر
  # ... تكوين آخر
}
```

---

# دعم بروتوكول M2PA

يدعم OmniSS7 كل من بروتوكولات **M2PA** و **M3UA** لنقل إشارات SS7.

## ما هو M2PA؟

**M2PA** هو بروتوكول موحد من قبل IETF (RFC 4165) لنقل طبقة التكيف بين مستخدم (MTP2) رسائل MTP3 الخاصة بـ SS7 عبر الشبكات المعتمدة على IP باستخدام SCTP.

## الاختلافات الرئيسية: M2PA مقابل M3UA

| الميزة | M3UA | M2PA |
|---|---|---|
| **العمارة** | عميل/خادم (ASP/SGW) | نظير إلى نظير |
| **حالة الاستخدام** | بوابة بين IP و SS7 | روابط مباشرة نقطة إلى نقطة |
| **إدارة حالة الرابط** | مستوى التطبيق (ASPUP/ASPAC) | نمط MTP2 (محاذاة، إثبات، جاهز) |
| **أرقام التسلسل** | لا يوجد تسلسل داخلي | 24 بت BSN/FSN للتسليم المرتب |
| **نشر نموذجي** | بوابة SS7 إلى IP، STP | روابط إشار��ت مباشرة بين العقد |
| **RFC** | RFC 4666 | RFC 4165 |

## إرشادات اختيار البروتوكول

**التوصية: استخدم M3UA بشكل افتراضي. استخدم M2PA فقط عند الحاجة بشكل محدد.**

## متى تستخدم M3UA (موصى به)

هو البروتوكول الموصى به لمعظم النشر M3UA:

- **نشر STP**: تنفيذات نقطة نقل الإشارات القياسية
- **وظائف البوابة**: ربط الشبكات SS7 بالإشارات المعتمدة على IP
- **اتصالات عناصر الشبكة**: ربط SMSCs و MSCs و HLRs وعناصر الشبكة الأخرى بـ STP الخاص بك
- **بوابة الإشارات (SGW)**: بوابة مركزية تقبل الاتصالات من عدة خوادم تطبيقات
- **الطوبولوجيات المرنة**: هياكل عميل/خادم مع تحكم مركزي
- **شبكات متعددة البائعين**: معيار الصناعة المدعوم على نطاق واسع (RFC 4666)

**استخدم M3UA لربط عناصر الشبكة (HLR، MSC، SMSC، VLR، إلخ) بـ STP الخاص بك.**

## متى تستخدم M2PA (فقط في حالات خاصة)

يجب استخدام M2PA فقط في سيناريوهات محددة:

- **روابط STP إلى STP**: ا��صالات مباشرة نقطة إلى نقطة بين نقاط نقل الإشارات في شبكة متعددة STP
- **استبدال TDM القديم**: استبدال الروابط التقليدية SS7 TDM عندما يتطلب النظام البعيد بشكل محدد M2PA
- **توافق MTP2 مطلوب**: عند الاتصال بأنظمة قديمة تتطلب إدارة حالة الرابط بأسلوب MTP2
- **متطلبات الشريك**: عندما يتطلب شريك أو اتصال بشكل محدد بروتوكول M2PA

**مهم**: لا تستخدم M2PA لربط عناصر الشبكة (HLR، MSC، SMSC) بـ STP الخاص بك - استخدم M3UA بدلاً من ذلك. تم تصميم M2PA للاتصالات بين STP حيث تعمل كلا الجانبين كعقد توجيه.

## تكوين أقران M2PA

يتم تكوين أقران M2PA بنفس طريقة تكوين أقران M3UA، مع معلمة إضافية `protocol`.

**تكوين نظير M2PA**

أضف أقران M2PA إلى تكوين `m3ua_peers` في `config/runtime.exs` (نعم، يتشاركون نفس قسم التكوين على الرغم من كونهم بروتوكولات مختلفة):

**المعلمات الرئيسية لـ M2PA:**

| المعلمة | القيمة | الوصف |
| --- | --- | --- |
| `protocol` | `:m2pa` | يحدد بروتوكول M2PA (يفترض `:m3ua` إذا تم حذفه) |
| `role` | `:client` أو `:server` | اتجاه الاتصال |
| `local_port` | Integer | منفذ SCTP المحلي (المنفذ القياسي لـ M2PA هو **3565**) |
| `remote_port` | Integer | منفذ SCTP عن بُعد (المنفذ القياسي لـ M2PA هو **3565**) |
| `point_code` | Integer | رمز النقطة الخاص بك |
| `adjacent_point_code` | Integer | رمز النقطة الخاص بالنظير (خاص بـ M2PA) |

**ملاحظة**: يستخدم M2PA كمعيار صناعي **المنفذ 3565** (مختلف عن المنفذ 2905 الخاص بـ M3UA).

**حالات رابط M2PA**

تتقدم روابط M2PA عبر عدة حالات أثناء التهيئة:

1. **أسفل** - لا يوجد اتصال تم إنشاؤه
2. **محاذاة** - مرحلة التزامن الأولية (~1 ثانية)
3. **إثبات** - التحقق من جودة الرابط (~2 ثانية)
4. **جاهز** - الرابط نشط وجاهز لحركة المرور

تضمن تقدم حالة الرابط موثوقية الإشارات قبل تبادل حركة المرور.

# إدارة أقران M2PA عبر واجهة الويب

توفر M2PA صفحة **التوجيه** في واجهة الويب دعمًا كاملًا لإدارة أقران:

1. **انتقل** إلى صفحة التوجيه

2. علامة التبويب "الأقران" **حدد**
3. على "إضافة نظير جديد" **انقر**
4. من قائمة البروتوكولات "M2PA (RFC 4165)" **اختر**
5. تكوين النظير **املأ**:
   - اسم النظير (معرف وصفي)
   - البروتوكول: M2PA
   - الدور: عميل أو خادم
   - رمز النقطة (PC الخاص بك)
   - عناوين IP المحلية/البعيدة
   - المنافذ المحلية/البعيدة (M2PA عادة 3565 لـ)
   - مؤشر الشبكة (دولي أو محلي)
6. على "حفظ النظير" **انقر**

تظهر جدول الأقران نوع البروتوكول مع ترميز الألوان:

- أقران - **أزرق** M3UA
- أقران - **أخضر** M2PA

## سلوك توجيه M2PA

تتكامل أقران M2PA بـ OmniSS7 بسلاسة مع نظام التوجيه الخاص:

- **مسارات رمز النقطة**: تعمل بشكل مماثل لـ M3UA و M2PA
- **مسارات العنوان العالمي**: مدعومة بالكامل على روابط M2PA
- **أولوية المسار**: يمكن خلط أقران M3UA و M2PA في نفس جداول التوجيه
- **تبادل الرسائل**: يمكن أن تصل الرسائل عبر M2PA وتوجه إلى M3UA، والعكس صحيح

## مقاييس M2PA

يوفر M2PA مقاييس شاملة لـ Prometheus لمراقبة صحة الرابط وحركة المرور:

**مقاييس الحركة:**

- `m2pa_messages_sent_total` - إجمالي رسائل MTP3 المرسلة لكل رابط
- `m2pa_messages_received_total` - إجمالي رسائل MTP3 المستلمة لكل رابط

- `m2pa_bytes_sent_total` - إجمالي البايتات المرسلة عبر M2PA
- `m2pa_bytes_received_total` - إجمالي البايتات المستلمة عبر M2PA

تُوسم جميع مقاييس الحركة بـ: `link_name`، `point_code`، `adjacent_pc`

**مقاييس حالة الرابط:**

- `m2pa_link_state_changes_total` - انتقالات حالة الرابط (أسفل → محاذاة → إثبات → جاهز)
  - التسميات: `link_name`، `from_state`، `to_state`

**مقاييس الأخطاء:**

- `m2pa_errors_total` - إجمالي الأخطاء حسب النوع
  - `decode_error` - فشل فك تشفير رسالة M2PA
  - `encode_error` - فشل تشفير رسالة M2PA
  - `sctp_send_error` - فشل الإرسال عبر SCTP
  - التسميات: `link_name`، `error_type`

**الوصول إلى المقاييس:**

- نقطة نهاية Prometheus: `http://your-server:8080/metrics`
- يتم تسجيل المقاييس تلقائيًا عند بدء تشغيل التطبيق

# أفضل الممارسات لـ M2PA

1. **اختيار المنفذ**: استخدم المنفذ 3565 لـ M2PA (معيار الصناعة)
2. **مراقبة الرابط**: راقب تغييرات حالة الرابط عبر المقاييس
3. **قواعد جدار الحماية**: تأكد من السماح بـ SCTP (بروتوكول IP 132)
4. **رموز النقطة**: تأكد من تكوين رموز النقطة المجاورة بشكل صحيح على كلا الجانبين
5. **مؤشر الشبكة**: يجب أن يتطابق بين الأقران (دولي أو محلي)
6. **الاختبار**: استخدم صفحة اختبار التوجيه للتحقق من الاتصال بعد التكوين

---

# تكوين توجيه رمز النقطة

**رمز النقطة الوجهة** يوجه توجيه رمز النقطة الرسائل بناءً على **(DPC)** في رأس MTP3.

## فهم رموز النقطة في كومة بروتوكول SS7

فهم هذا التمييز مهم .SS7 توجد رموز النقطة في طبقات مختلفة من كومة بروتوكول:

**طبقات كومة البروتوكول:**

```
┌─────────────────────────────────────────────┐
│  طبقة التطبيق (SCCP/TCAP/MAP)               │
├─────────────────────────────────────────────┤
│  طبقة MTP3                                  │ ← توجيه بيانات المستخدم
│  - تسمية التوجيه: DPC، OPC، SLS             │ ← تستخدم لتوجيه STP
│  - بايت معلومات الخدمة (SIO)                │
├─────────────────────────────────────────────┤
│  M3UA أو M2PA (طبقة التكيف)                 │ ← بروتوكول النقل
│  - بيانات البروتوكول (تحتوي على MTP3)       │
│  - إدارة الشبكة (DUNA/DAVA)                 │ ← حالة الشبكة
├─────────────────────────────────────────────┤
│  SCTP (النقل)                               │
└─────────────────────────────────────────────┘
```

**نوعان من رموز النقطة:**

1. **رموز النقطة في طبقة MTP3** (تستخدم للتوجيه):
   - تقع في تسمية التوجيه MTP3 (DPC، OPC)
   - موجودة في معلمة بيانات البروتوكول M3UA (علامة 528)
   - موجودة في رسائل بيانات المستخدم M2PA
   - **يستخدم STP هذه القيم DPC لقرارات التوجيه**
   - تحدد المكان الذي يتم تسليم الرسالة في النهاية
2. **رموز النقطة في طبقة M3UA** (تستخدم لإدارة الشبكة):
   - موجودة في رسائل إدارة M3UA (DUNA، DAVA، SCON، DUPU)
   - تشير إلى رموز النقطة المتأثرة بحالة الشبكة

- تخبر الأقران بالوجهات المتاحة/غير المتاحة
- لا تستخدم لتوجيه بيانات المستخدم

**كيف يعمل توجيه STP:**

- **M3UA DATA بالنسبة لرسائل**: يستخرج STP رسالة MTP3 من معلمة بيانات البروتوكول MTP3 (علامة 528)، التي تحتوي على تسمية توجيه MTP3 (DPC، OPC، SLS). يتم استخدام DPC من طبقة MTP3 للبحث عن المسارات.
- **M2PA بـ��نسبة لرسائل بيانات المستخدم**: يستخرج STP رسالة MTP3 من حقل بيانات المستخدم M2PA، ثم يقرأ DPC من تسمية توجيه MTP3.
- **M3UA رسائل إدارة**: تحتوي رسائل إدارة الشبكة (DUNA، DAVA، SCON) على رموز النقطة المتأثرة في طبقة M3UA لإشارات حالة الشبكة بين الأقران.

# المسارات الأساسية لرمز النقطة

أضف المسارات إلى `config/runtime.exs`:

```
config :omniss7,
  m3ua_routes: [
    # 1 توجيه كل حركة المرور لرمز النقطة 100 إلى النظير (STP الشريك)
    %{
      dest_pc: 100,                    # رمز النقطة الوجهة
      peer_id: 1,                      # النظير الذي سيتم التوجيه من خلاله
      priority: 1,                     # الأولوية (أقل = أولوية أعلى)
      network_indicator: :international
      # mask: 14                       # 14 اختياري: الافتراضي هو (مطابقة دقيقة)
    },

    # 2 توجيه كل حركة المرور لرمز النقطة 200 إلى النظير (HLR المحلي)
    %{
      dest_pc: 200,
      peer_id: 2,
      priority: 1,
      network_indicator: :international
    },

    # مثال على توزيع الحمل: رمز النقطة 300 مع مسارات أساسية واحتياطية
    %{
      dest_pc: 300,
      peer_id: 3,                      # المسار الأساسي
      priority: 1,
      network_indicator: :international
    },
    %{
      dest_pc: 300,
      peer_id: 4,                      # المسار الاحتياطي (رقم أولوية أعلى)
      priority: 2,
      network_indicator: :international
    }
  ]
```

فقط عند `mask` اختياري ويفترض افتراضيًا `14` (مطابقة دقيقة). حدد **ملاحظة**: حقل `mask` الحاجة إلى توجيه قائم على النطاق (انظر قسم أقنعة رمز النقطة أدناه).

## منطق التوجيه

1. يستقبل STP رسالة M3UA DATA المستخدم M2PA أو رسالة بيانات
2. يستخرج STP **رسالة MTP3** من معلمة البيانات (M3UA) أو حقل بيانات المستخدم (M2PA)
3. يقرأ STP **رمز النقطة الوجهة (DPC)** من تسمية توجيه MTP3
4. يبحث في جدول التوجيه عن DPC (مع مراعاة الأقنعة) المطابق
5. إذا كانت هناك مسارات متعددة، يختار المسار بأكثر **قناع محدد** (أعلى قيمة قناع)، ثم **قل رقم أولوية**
6. يلف رسالة MTP3 في بيانات مستخدم M3UA DATA أو M2PA للنظير الوجهة
7. يوجه الرسالة إلى النظير المقابل
8. إذا كان النظير المختار غير متاح، يحاول المسار التالي بأعلى أولوية

## أقنعة رمز النقطة

رموز النقطة هي قيم 14 بت (نطاق 0-16383). بشكل افتراضي، تتطابق المسارات مع رمز نقطة واحد بدقة (قناع `/14`). ومع ذلك، يمكنك استخدام **أقنعة رمز النقطة** لإنشاء مسارات تتطابق مع **نطاقات** رموز النقطة.

**فهم الأقنعة**

يحدد القناع عدد **أعلى بتات ذات دلالة** يجب أن تتطابق بين رمز النقطة الوجهة للمسار و DPC للرسالة الواردة. يمكن أن تكون البتات المتبقية أي قيمة، مما ينشئ نطاقًا من رموز النقطة المتطابقة.

**جدول مرجع القناع:**

| القناع | رموز النقطة المتطابقة | حالة الاستخدام |
|---|---|---|
| `/14` | 1 (مطابقة دقيقة) رمز نقطة | وجهة واحدة (افتراضي) |
| `/13` | 2 رموز نقطة | نطاق صغير |
| `/12` | 4 رموز نقطة | نطاق صغير |
| `/11` | 8 رموز نقطة | نطاق صغير |
| `/10` | 16 رمز نقطة | نطاق متوسط |
| `/9` | 32 رمز نقطة | نطاق متوسط |
| `/8` | 64 رمز نقطة | نطاق متوسط |
| `/7` | 128 رمز نقطة | نطاق متوسط كبير |
| `/6` | 256 رمز نقطة | نطاق كبير |
| `/5` | 512 رمز نقطة | نطاق كبير |
| `/4` | 1,024 رمز نقطة | نطاق كبير جدًا |
| `/3` | 2,048 رمز نقطة | نطاق كبير جدًا |
| `/2` | 4,096 رمز نقطة | نطاق كبير جدًا |
| `/1` | 8,192 رمز نقطة | نصف جميع الرموز |
| `/0` | 16,384 رمز نقطة | جميع الرموز (مسار افتراضي/احتياطي) |

**أمثلة على أقنعة رمز النقطة**

**ملاحظة**: حقل `mask` `14` هو **اختياري** في جميع الأمثلة. إذا تم حذفه، فإنه يفترض افتراضيًا (مطابقة دقيقة).

**مثال 1: رمز نقطة واحد (سلوك افتراضي)**

```
# بدون حقل mask (موصى به للرمز الفردي)
%{
  dest_pc: 1000,
  peer_id: 1,
  priority: 1,
  network_indicator: :international
}
# فقط: يتطابق - 14 القناع الافتراضي هو PC 1000

# قناع صريح (نفس النتيجة)
%{
  dest_pc: 1000,
  peer_id: 1,
  priority: 1,
  mask: 14,                                  # مطابقة دقيقة صريحة
  network_indicator: :international
}
# فقط: يتطابق PC 1000
```

**مثال 2: نطاق صغير**

```
%{
  dest_pc: 1000,
  peer_id: 2,
  priority: 1,
  mask: 12,                                  # يتطابق مع 4 رموز نقطة
  network_indicator: :international
}
# يتطابق: PC 1000، 1001، 1002، 1003
```

**مثال 3: نطاق متوسط**

```
%{
  dest_pc: 1000,
  peer_id: 3,
  priority: 1,
  mask: 8,                                  # يتطابق مع 64 رمز نقطة
  network_indicator: :international
}
# يتطابق: PC 1000-1063 (64 رمز نقطة متتالية)
```

**مثال 4: مسار افتراضي/احتياطي**

```
%{
  dest_pc: 0,
  peer_id: 4,
  priority: 10,                             # أولوية منخفضة (رقم
(مرتفع
  mask: 0,                                  # يتطابق مع جميع الرموز
  network_indicator: :international
}
# يتطابق: جميع رموز النقطة (0-16383)
# استخدم كمسار افتراضي/احتياطي مع أولوية منخفضة
```

**دمج المسارات المحددة والمسارات المقنعة**

يمكنك دمج المسارات المحددة مع المسارات المقنعة لتوجيه مرن:

```
config :omniss7,
  m3ua_routes: [
    # مسار محدد لرمز النقطة 100 (يأخذ الأولوية)
    %{
      dest_pc: 100,
      peer_id: 1,
      priority: 1,
      network_indicator: :international
      # القناع الافتراضي هو 14 (مطابقة دقيقة)
    },

    # مسار نطاق لرموز النقطة 1000-1063
    %{
      dest_pc: 1000,
      peer_id: 2,
      priority: 1,
      mask: 8,                          # يتطابق مع 64 رمز نقطة
      network_indicator: :international
    },

    # مسار افتراضي/احتياطي لجميع الرموز الأخرى
    %{
      dest_pc: 0,
      peer_id: 3,
      priority: 10,                     # أولوية منخفضة
      mask: 0,                          # يتطابق مع جميع الرموز
      network_indicator: :international
    }
  ]
```

**قرار التوجيه لـ DPC 1000:**

1. `14/` يتطابق مع المسار القناع (PC 1000 بدقة) - (الأكثر تحديدًا) **محدد**
2. `8/` يتطابق أيضًا مع المسار القناع (PC 1000-1063 نطاق) - (أقل تحديدًا) يتم تجاهله
3. `0/` يتطابق أيضًا مع المسار القناع (جميع الرموز) - يتم تجاهله (الأقل تحديدًا)

**قرار التوجيه لـ DPC 1015:**

1. `14/` لا يتطابق مع المسار القناع (PC 1000 فقط)
2. `8/` يتطابق مع المسار القناع (PC 1000-1063 نطاق) - (الأكثر تحديدًا) **محدد**
3. `0/` يتطابق أيضًا مع المسار القناع (جميع الرموز) - يتم تجاهله (الأقل تحديدًا)

**قرار التوجيه لـ DPC 5000:**

1. لا يتطابق مع المسار القناع `14/`
2. لا يتطابق مع المسار القناع `8/`
3. يتطابق مع المسار القناع `0/` (جميع الرموز) - **محدد** (الوحيد المتطابق، مسار احتياطي)

**أفضل الممارسات**

1. **تجنب `mask` للمقاصد الفردية**: للمطابقات الدقيقة لرموز النقطة الفردية، تجنب حقل `mask` (يفترض افتراضيًا `14/`) تمامًا
2. **استخدم `14/` بشكل صريح فقط عند الحاجة**: حدد `mask: 14` فقط عندما تحتاج إلى توضيح ذلك في الوثائق أو عند الخلط مع المسارات النطاقية
3. **استخدم الأقنعة النطاقية لكتل الشبكة**: وجه مقاطع الشبكة بالكامل إلى أقران محددين باستخدام الأقنعة `0/` إلى `13/`
4. **استخدم `0/` كمسار احتياطي**: أنشئ مسار افتراضي مع أولوية منخفضة لالتقاط حركة المرور غير المتطابقة
5. **الأكثر تحديدًا يفوز**: يختار محرك التوجيه دائمًا المسار الأكثر تحديدًا (أعلى قيمة قناع) أولاً
6. **الأولوية كمعيار فاصلة**: إذا كانت هناك مسارات متعددة بنفس القناع، فإن أقل رقم أولوية يفوز

---

# تكوين توجيه العنوان العالمي (GT)

يتيح توجيه العنوان العالمي IMSI **التوجيه القائم على المحتوى** باستخدام أرقام الهواتف أو قيم بدلاً من رموز النقطة. للحصول على ترجمة متقدمة لعنوان العنوان العالمي بناءً على الطرف المتصل/المتصل، انظر دليل NAT للعناوين العالمية.

## المتطلبات المسبقة

- تمكين توجيه GT: `enable_gt_routing: true` ف`config/runtime.exs`

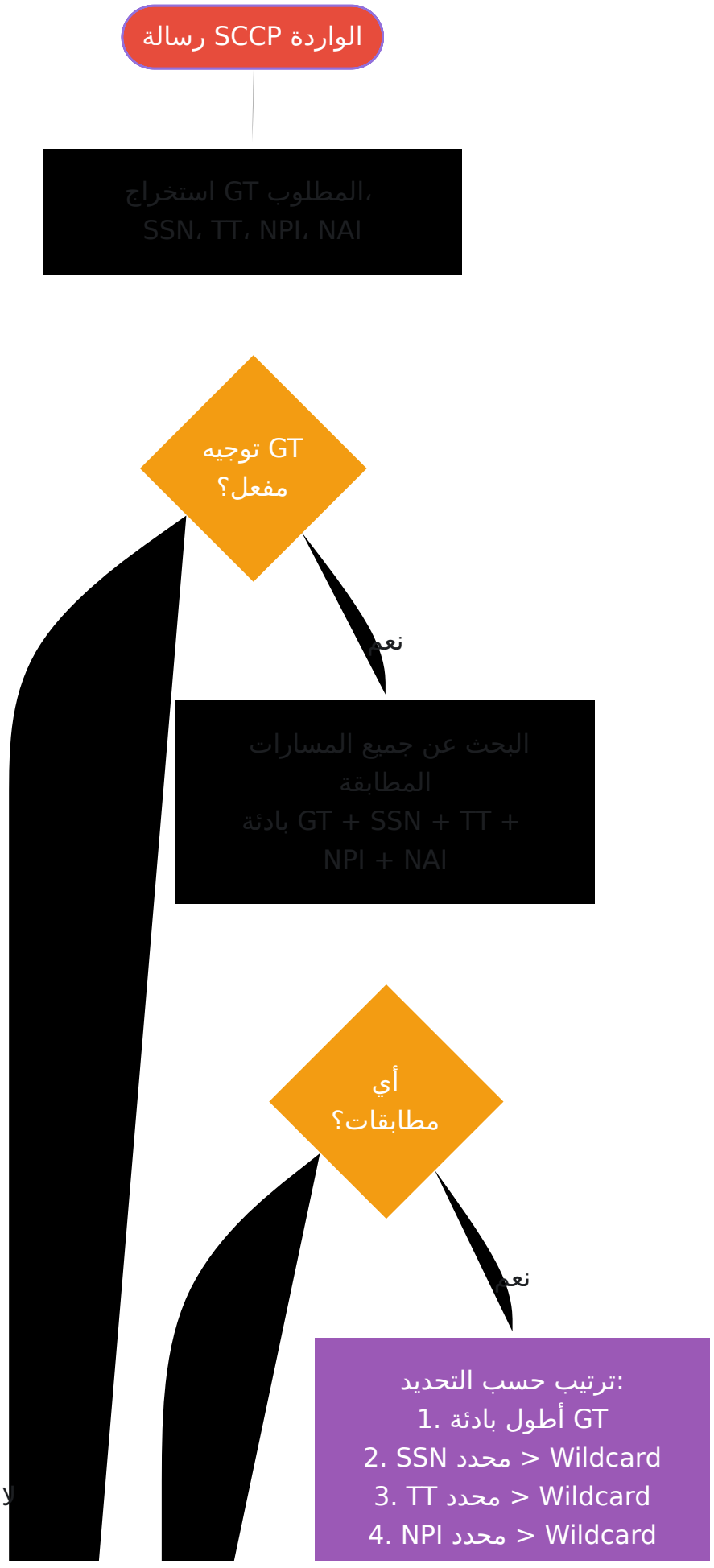
## تكوين مسار GT

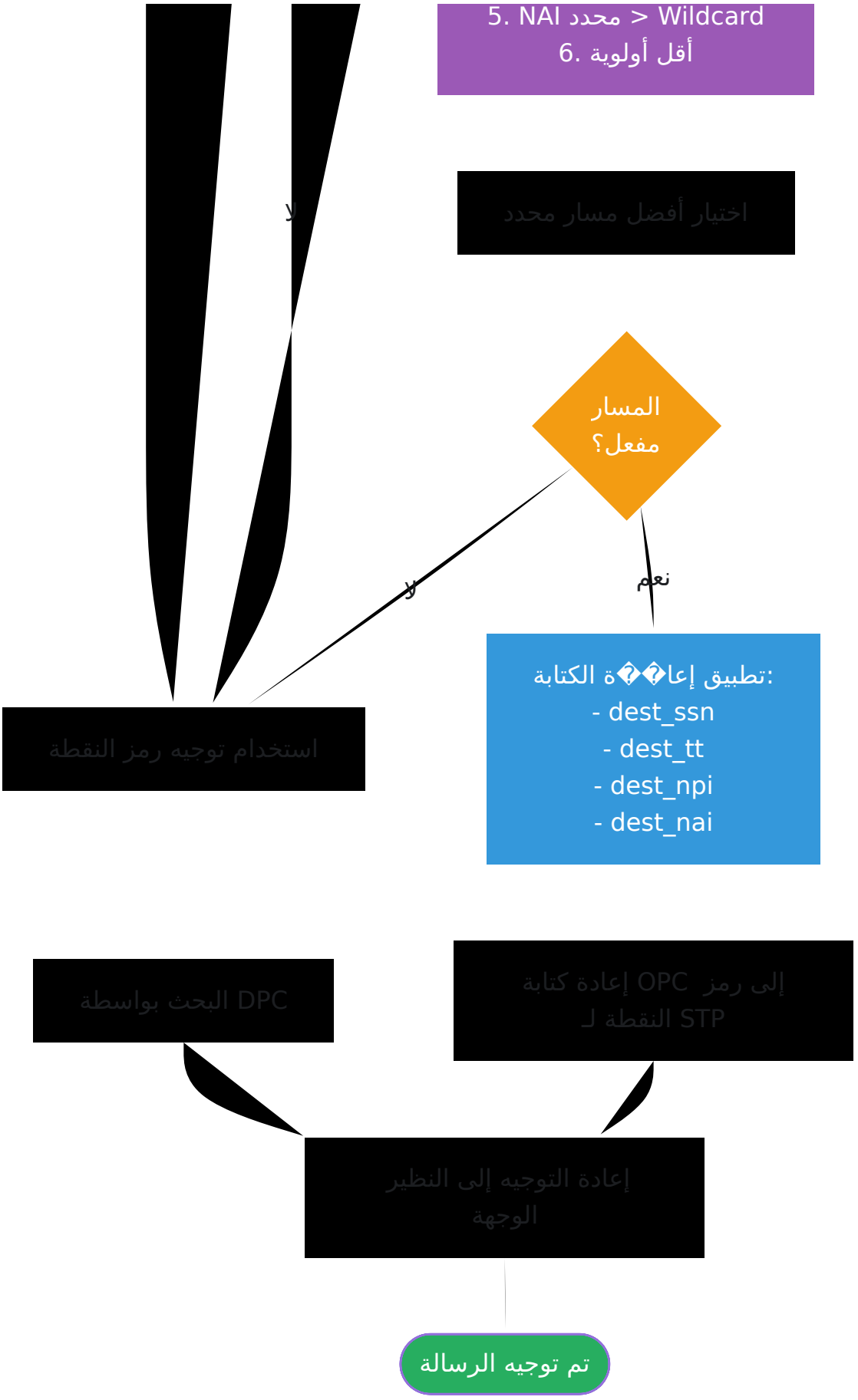
```
config :omniss7,
  # تمكين توجيه GT
  enable_gt_routing: true,

  m3ua_gt_routes: [
    # توجيه جميع الأرقام البريطانية (البادئة 44) إلى النظير 1
    %{
      gt_prefix: "44",                # بادئة العنوان العالمي للمطابقة
      peer_id: 1,                     # النظير الوجهة
      priority: 1,                    # الأولوية (أقل = أعلى)
      description: "أرقام بريطانية"   # وصف للتسجيل
    },

    # توجيه الأرقام الأمريكية (البادئة 1) إلى النظير 2
    %{
      gt_prefix: "1",
      peer_id: 2,
      priority: 1,
      description: "أرقام أمريكية"
    },

    # مسار أكثر تحديدًا: أرقام الهواتف المحمولة البريطانية التي تبدأ بـ 447
    %{
      gt_prefix: "447",               # أطول بادئة تتطابق تفوز
      peer_id: 3,
      priority: 1,
      description: "أرقام الهواتف المحمولة البريطانية"
    },

    # توجيه محدد SSN (اختياري)
    %{
      gt_prefix: "555",
      source_ssn: 8,                  # تطابق فقط إذا كانت SSN المصدر = 8 (SMSC)
      peer_id: 4,
      dest_ssn: 6,                    # إعادة كتابة SSN الوجهة إلى 6 (HLR)
      priority: 1,
      description: "حركة SMS 61 للبادئة"
```

```
    }
  ]
```

## GT منطق توجيه

هذه العملية GT يتبع خوارزمية توجيه:


رسالة SCCP الواردة
استخراج GT المطلوب،
SSN، TT، NPI، NAI
توجيه GT
مفعل؟
نعم
البحث عن جميع المسارات
المطابقة
بادئة GT + SSN + TT +
NPI + NAI
أي
مطابقات؟
نعم
ترتيب حسب التحديد:
1. أطول بادئة GT
2. SSN محدد > Wildcard
3. TT محدد > Wildcard
4. NPI محدد > Wildcard




**خطوات التوجيه:**

1. **مطابقة أطول بادئة**: يجد STP جميع مسارات GT حيث تتطابق البادئة مع بداية العنوان العالمي

- مثال: GT "447712345678" تتطابق مع كل من "44" و "447"، لكن "447" تفوز (أطول مطابقة)

2. **مطابقة SSN** (اختياري):
   - الطرف SSN يتطابق المسار فقط عندما تكون `source_ssn` إذا تم تحديد تساوي تلك القيمة SCCP المتصل في
   - SSN يتطابق المسار مع أي `nil` تساوي `source_ssn` إذا كانت (wildcard)

3. **مطابقة TT/NPI/NAI** (اختياري):
   - يجب أن `source_nai`، `source_npi` أو `source_tt` إذا تم تحديد تتطابق المسارات مع تلك المؤشرات
   - القيم `nil` تعمل كـ wildcards (تتطابق مع أي قيمة)

4. **اختيار بناءً على التحديد**:
   - تفوز المسارات التي تحتوي على معايير مطابقة أكثر تحديدًا على wildcards
   - ترتيب الأولوية: طول بادئة GT → SSN → TT → NPI → NAI → رقم الأولوية

5. **إعادة كتابة المؤشرات** (اختياري):
   - يقوم `dest_nai`، `dest_npi` أو `dest_tt` أو `dest_ssn` إذا تم تحديد بإعادة كتابة تلك المؤشرات STP
   - مفيد لتطبيع البروتوكول والاتصال بالشبكة

6. **العودة إلى رمز النقطة**:
   - إلى توجيه رمز النقطة باستخدام STP، يعود GT، إذا لم تتطابق أي مسار DPC

## توجيه GT المتقدم: نوع الترجمة، NPI، و NAI

التوجيه والتحويل بناءً على مؤشرات العنوان STP يدعم، SSN و GT بالإضافة إلى مطابقة بادئة العالمي SCCP:

- **نوع الترجمة (TT)**: يحدد خطة الترقيم ونوع العنوان
- **مؤشر خطة الترقيم (NPI)**: يحدد خطة الترقيم (مثل ISDN، البيانات، Telex)
- **مؤشر طبيعة العنوان (NAI)**: يحدد تنسيق العنوان (مثل دولي، محلي، رقم مشترك)

**المطابقة (المؤشرات المصدر)**

يمكن أن تتطابق المسارات على مؤشرات الرسالة الواردة:

- `source_tt`: مطابقة الرسائل مع نوع الترجمة المحدد
- `source_npi`: مطابقة الرسائل مع مؤشر خطة الترقيم المحدد
- `source_nai`: مطابقة الرسائل مع مؤشر طبيعة العنوان المحدد
- القيمة `nil` = wildcard (تتطابق مع أي قيمة)

**التحويل (المؤشرات الوجهة)**

يمكن أن تعيد المسارات كتابة المؤشرات عند إعادة التوجيه:

- `dest_tt`: تحويل نوع الترجمة إلى قيمة جديدة
- `dest_npi`: تحويل مؤشر خطة الترقيم إلى قيمة جديدة
- `dest_nai`: تحويل مؤشر طبيعة العنوان إلى قيمة جديدة
- القيمة `nil` = (بدون تحويل) الحفاظ على القيمة الأصلية

**اختيار ب��اءً على التحديد**

عند تطابق عدة مسارات، يتم اختيار المسار الأكثر تحديدًا باستخدام ترتيب الأولوية هذا:

1. أطول مطابقة بادئة GT
2. SSN محدد على SSN wildcard
3. TT محدد على TT wildcard
4. NPI محدد على NPI wildcard

5. NAI محدد على NAI wildcard
6. أقل رقم أولوية

**أمثلة التكوين**

```
config :omniss7,
  enable_gt_routing: true,

  m3ua_gt_routes: [
    # المثال 1: مطابقة وتحويل نوع الترجمة
    %{
      gt_prefix: "44",
      peer_id: 1,
      source_tt: 0,      # مطابقة TT=0 (غير معروف)
      dest_tt: 3,        # تحويل إلى TT=3 (محلي)
      priority: 1,
      description: "أرقام بريطانية: TT 0→3 التحويل"
    },

    # المثال 2: مطابقة NPI محدد وتحويل NAI
    %{
      gt_prefix: "1",
      peer_id: 2,
      source_npi: 1,     # مطابقة NPI=1 (ISDN/الهاتف)
      source_nai: 4,     # مطابقة NAI=4 (دولي)
      dest_nai: 3,       # تحويل إلى NAI=3 (محلي)
      priority: 1,
      description: "أرقام أمريكية: دولي→محلي NAI"
    },

    # المثال 3: توجيه مشترك ومؤشر SSN
    %{
      gt_prefix: "33",
      source_ssn: 8,     # مطابقة حركة SMSC
      source_tt: 0,      # مطابقة TT=0
      dest_ssn: 6,       # إعادة كتابة SSN إلى HLR
      dest_tt: 2,        # تحويل إلى TT=2
      dest_npi: 1,       # تعيين NPI=1 (ISDN)
      dest_nai: 4,       # تعيين NAI=4 (دولي)
      peer_id: 3,
      priority: 1,
      description: "فرنسي: تطبيع كامل SMS"
    },

    # المثال 4: TT wildcard، NPI محدد
    %{
      gt_prefix: "49",
      source_tt: nil,    # مطابقة أي TT (wildcard)
```

```
      source_npi: 6,     # مطابقة NPI=6 (بيانات)
      dest_npi: 1,       # تحويل إلى NPI=1 (ISDN)
      peer_id: 4,
      priority: 1,
      description: "تطبيع شبكة البيانات الألمانية"
    }
  ]
```

**القيم الشائعة لـ TT/NPI/NAI**

**نوع الترجمة (TT):**

- `0` = غير معروف
- `1` = دولي
- `2` = محلي
- `3` = محدد للشبكة

**مؤشر خطة الترقيم (NPI):**

- `0` = غير معروف
- `1` = الهاتف/ISDN (E.164)
- `3` = بيانات (X.121)
- `4` = Telex (F.69)
- `6` = الهاتف المحمول (E.212)

**مؤشر طبيعة العنوان (NAI):**

- `0` = غير معروف
- `1` = رقم مشترك
- `2` = محجوز للاست��دام المحلي
- `3` = رقم مهم محلي
- `4` = رقم دولي

**مثال قرار التوجيه**

لرسالة واردة مع:

- GT: "447712345678"

- SSN: 8
- TT: 0
- NPI: 1
- NAI: 4

مع هذه المسارات المكونة:

```
# المسار A: TT wildcard
%{gt_prefix: "447", peer_id: 1, priority: 1}

# المسار B: TT محدد
%{gt_prefix: "447", source_tt: 0, peer_id: 2, priority: 1}

# المسار C: TT محدد + NPI
%{gt_prefix: "447", source_tt: 0, source_npi: 1, peer_id: 3,
priority: 1}
```

**النتيجة**: يتم اختيار المسار C (يتطابق مع GT + TT + NPI الأكثر تحديدًا)

يتم إعادة توجيه الرسالة مع المؤشرات المحولة وفقًا لقيم `dest_nai` و `dest_npi` و `dest_tt` الخاصة بالمسار C.

## أمثلة توجيه GT

| السبب | المسار المطابق | NAI | NPI | TT | SSN المصدر | GT المطلوب |
|---|---|---|---|---|---|---|
| أطول مطابقة بادئة | "447" → النظير 3 | - | - | - | 6 | 447712345678 |
| مطابقة بادئة، لا يوجد مسار أكثر تحديدًا | "44" → النظير 1 | - | - | - | 6 | 441234567890 |
| مطابقة بادئة للأرقام الأمريكية | "1" → النظير 2 | - | - | - | 6 | 12125551234 |
| مطابقة GT + SSN، تعيد كتابة SSN إلى 6 | "555" (SSN 8) → النظير 4 | - | - | - | 8 | 555881234567 |
| مطابقة GT، لا إعادة كتابة SSN | "555" (SSN wildcard) → النظير X | - | - | - | 6 | 555881234567 |
| مطابقة GT + TT، تحول TT إلى 3 | "44" (TT=0) → النظير 1 | 4 | 1 | 0 | 6 | 441234567890 |
| الأكثر تحديدًا: مطابقة GT + TT + NPI + NAI | "1" (TT=0، NPI=1، NAI=4) | 4 | 1 | 0 | 8 | 12125551234 |

**حالات الاستخدام العملية لتوجيه TT/NPI/NAI**

1. **تطبيع الاتصال بين الشبكات**

- قد تستخدم الشبكات المختلفة مؤشرات مختلفة
- تحويل المؤشرات عند نقطة الاتصال لضمان التوافق
- مثال: تستخدم الشبكة الشريكة TT=0، تستخدم شبكتك TT=1 للدولي

2. **تحويل البروتوكول**

- تحويل بين خطط الترقيم عند التوجيه بين أنواع الشبكات المختلفة
- مثال: التوجيه من شبكة الهاتف المحمول (NPI=6) إلى PSTN (NPI=1)

3. **توحيد تنسيق العنوان**

- تطبيع جميع حركة المرور الواردة لاستخدام قيم NAI متسقة
- مثال: تحويل جميع التنسيقات الدولية (NAI=3) إلى تنسيق محلي (NAI=4) للتوجيه المحلي

4. **توجيه محدد حسب الناقل**

- توجيه بناءً على نوع الترجمة إلى مزودي الخدمة المختلفين
- مثال: TT=0 يتجه إلى الناقل A، TT=2 يتجه إلى الناقل B

5. **دمج الأنظمة القديمة**

- قد تستخدم الأنظمة الحديثة قيم مؤشرات مختلفة عن الأنظمة القديمة
- تحويل عند STP للحفاظ على التوافق مع الأنظمة القديمة

---

# ميزات إدارة المسار

## تعطيل المسارات

يمكن تعطيل المسارات مؤقتًا دون حذفها. هذا مفيد للاختبار أو الصيانة أو إدارة الحركة.

**علامة مفعل**

تدعم كل من مسارات رمز النقطة ومسارات العنوان العالمي علامة `enabled` اختيارية:

```
config :omniss7,
  m3ua_routes: [
    # مسار نشط
    %{
      dest_pc: 100,
      peer_id: 1,
      priority: 1,
      network_indicator: :international,
      enabled: true  # (افتراضي إذا تم حذفه) المسار نشط
    },

    # (لا يتم تقييمه أثناء التوجيه) مسار معطل
    %{
      dest_pc: 200,
      peer_id: 2,
      priority: 1,
      network_indicator: :international,
      enabled: false  # المسار معطل
    }
  ],

  m3ua_gt_routes: [
    # مسار GT معطل
    %{
      gt_prefix: "44",
      peer_id: 1,
      priority: 1,
      description: "أرقام بريطانية - معطلة مؤقتًا",
      enabled: false
    }
  ]
```

**السلوك الافتراضي:**

- إذا لم يتم تحديد `enabled`، فإن المسارات تفترض افتراضيًا `enabled: true`
- يتم تخطي المسارات المعطلة تمامًا أثناء البحث عن المسار
- استخدم واجهة الويب لتبديل المسارات تشغيل/إيقاف دون تعديل التكوين

**حالات الاستخدام:**

- اختبار تدفقات الحركة

- نوافذ صيانة مؤقتة
- اختبار A/B لمسارات توجيه مختلفة
- طرح تدريجي لمسارات جديدة

---

# DROP Routes - منع حلقات التوجيه

تقوم مسارات DROP (مع `peer_id: 0`) بالتخلص من الحركة مرورًا بدلاً من إعادة توجيهها. يمنع ذلك حلقات التوجيه ويمكّن من تصفية الحركة المتقدمة.

### تكوين مسارات DROP

```
config :omniss7,
  m3ua_routes: [
    # مسار DROP لـنقطة الـمحدد لـرمز ??
    %{
      dest_pc: 999,
      peer_id: 0,        # peer_id=0 يعني DROP
      priority: 1,
      network_indicator: :international
    }
  ],

  m3ua_gt_routes: [
    # مسار DROP لـبادئة GT
    %{
      gt_prefix: "999",
      peer_id: 0,        # peer_id=0 يعني DROP
      priority: 99,
      description: "حظر نطاق الاختبار"
    }
  ]
```

### كيف تعمل مسارات DROP

عندما تتطابق رسالة مع مسار DROP:

1. يحدد محرك التوجيه `peer_id: 0`
2. يتم **التخلص من الرسالة بهدوء** (لا يتم إعادة توجيهها)

3. **سجل** يتم إنشاء **INFO**: `"لرمز النقطة 999" DROP تمت المطابقة مع مسار` أو `"تمت المطابقة مع مسار DROP لGT 999"`
4. تعود عملية البحث عن التوجيه بـ `{:error, :dropped}`

**مهم:** يتم تسجيل الحركة المفقودة على مستوى INFO للمراقبة واستكشاف الأخطاء وإصلاحها.

**حالة الاستخدام الشائعة: قائمة بيضاء للبادة**

هي **قائمة بيضاء للبادة** - السماح فقط بأرقام DROP واحدة من أقوى استخدامات مسارات محددة ضمن نطاق كبير بينما يتم حظر جميع الأرقام الأخرى.

**النمط:**

1. إنشاء مسار DROP (مثل 99) **رقم أولوية مرتفع** للبادة بالكامل مع
2. إنشاء مسارات السماح المحددة لأرقام فردية مع **أرقام أولوية منخفضة** (مثل 1)
3. نظرًا لأن الأرقام ذات الأولوية المنخفضة يتم تقييمها أولاً، فإن المسارات المسموح بها تتطابق قبل مسار DROP
4. أي رقم غير مسموح به يتم القبض عليه بواسطة مسار DROP

**مثال سيناريو:**

لديك بادئة GT `1234` التي تمثل نطاق 10,000 رقم (1234000000 - 1234999999)، لكنك تريد فقط توجيه 3 أرقام محددة: `1234567890`، `1234555000`، و `1234111222`.

```
config :omniss7,
  m3ua_gt_routes: [
    # مسار DROP (يتم تقييمه أخيرًا) مع رقم أولوية مرتفع
    %{
      gt_prefix: "1234",
      peer_id: 0,              # DROP
      priority: 99,            # يتم = أولوية منخفضة = رقم مرتفع
تقييمه أخيرًا
      description: "حظر جميع 1234* باستثناء الأرقام المسموح بها"
    },

    # مسارات السماح المحددة مع أرقام أولوية منخفضة (يتم تقييمها
أولاً)
    %{
      gt_prefix: "1234567890",
      peer_id: 1,              # 1 توجيه إلى النظير
      priority: 1,             # يتم = أولوية عالية = رقم منخفض
تقييمه أولاً
      description: "1 رقم مسموح به"
    },
    %{
      gt_prefix: "1234555000",
      peer_id: 1,
      priority: 1,
      description: "2 رقم مسموح به"
    },
    %{
      gt_prefix: "1234111222",
      peer_id: 1,
      priority: 1,
      description: "3 رقم مسموح به"
    }
  ]
```

**سلوك التوجيه:**

| الإجراء | المسار المحدد | مسارات المطابقة | الوارد GT |
|---|---|---|---|
| **تم توجيهه إلى النظير 1** | "1234567890" (الأكثر تحديدًا، أعلى أولوية) | ✅ "1234567890" (الأولوية 1)<br>❌ "1234" DROP (الأولوية 99) | 1234567890 |
| **تم توجيهه إلى النظير 1** | "1234555000" (الأكثر تحديدًا، أعلى أولوية) | ✅ "1234555000" (الأولوية 1)<br>❌ "1234" DROP (الأولوية 99) | 1234555000 |
| **تم توجيهه إلى النظير 1** | "1234111222" (الأكثر تحديدًا، أعلى أولوية) | ✅ "1234111222" (الأولوية 1)<br>❌ "1234" DROP (الأولوية 99) | 1234111222 |
| **تم التخلص منه + تسجيله** | "1234" DROP (المطابقة الوحيدة) | ❌ "1234" DROP (الأو��وية 99) | 1234999999 |
| **تم التخلص منه + تسجيله** | "1234" DROP (المطابقة الوحيدة) | ❌ "1234" DROP (الأولوية 99) | 1234000000 |

**النتيجة:**

- ✅ يتم توجيه 3 أرقام محددة فقط إلى النظير 1
- ❌ يتم التخلص من جميع الأرقام الأخرى `1234*` بهدوء
- ✅ يتم تسجيل جميع الحركة المفقودة للمراقبة

**السجلات الناتجة:**

```
[INFO] تم المطابقة مع مسار DROP لGT 1234999999
[INFO] تم المطابقة مع مسار DROP لDPC 1050
```

**مسارات DROP لرموز النقطة**

يعمل نفس نمط قائمة البيضاء مع توجيه رمز النقطة:

```
config :omniss7,
  m3ua_routes: [
    # DROP (1063-1000 :رمز نقطة 64) 8/ نطاق كامل
    %{
      dest_pc: 1000,
      peer_id: 0,
      priority: 99,
      mask: 8,
      network_indicator: :international
    },

    # السماح برموز النقطة المحددة
    %{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}
  ]
```

**النتيجة:** يتم توجيه الرموز 1010 و 1020 و 1030 فقط. يتم التخلص من جميع الرموز الأخرى في نطاق 1000-1063.

**مراقبة مسارات DROP**

**تحقق من السجلات:**

```
# راقب حركة المرور المفقودة
tail -f logs/app.log | grep "تم المطابقة مع مسار DROP"

# الناتج المتوقع:
[INFO] تم المطابقة مع مسار DROP لGT 1234999999
[INFO] تم المطابقة مع مسار DROP لDPC 1050
```

**عبر واجهة الويب:**

- انتقل إلى علامة التبويب **سجلات النظام**
- قم بتصفية حسب مستوى `INFO`

- ابحث عن "تم المطابقة مع مسار DROP"

**أفضل الممارسات:**

1. ⚠ لا تحظر حركة المرور DROP راقب السجلات بانتظام للتأكد من أن مسارات المشروعة
2.  استخدم حقول `description` الوصفية لتوثيق سبب حظر المسارات
3.  استخدم أرقام أولوية مرتفعة (90-99) لمسارات DROP لضمان أنها مسارات شاملة
4.  اختبر سلوك مسارات DROP قبل نشرها في الإنتاج
5.  قم بإعداد تنبيهات لزيادة غير متوقعة في الحركة المفقودة

---

# التوجيه المتقدم: توجيه SSN وإعادة الكتابة

## أرقام النظام الفرعي (SSN)

تحدد أرقام النظام الفرعي طبقة التطبيق:

- **SSN 6**: HLR (سجل الموقع الرئيسي)
- **SSN 7**: VLR (سجل الموقع الزائر)
- **SSN 8**: MSC (مركز التحويل المحمول) / SMSC (مركز الرسائل القصيرة)
- **SSN 9**: GMLC (مركز الموقع المحمول البوابة)

## مثال على توجيه SSN

توجيه حركة SMS إلى HLRs مختلفة بناءً على بادئة الرقم:

```
m3ua_gt_routes: [
  # توجيه SMS إلى للأرقام البريطانية إعادة كتابة HLR، البريطاني SSN
8 من (SMSC) 6 إلى (HLR)
  %{
    gt_prefix: "44",
    source_ssn: 8,                  # مطابقة SSN 8 الواردة
(SMSC)
    peer_id: 1,
    dest_ssn: 6,                    # إعادة كتابة إلى SSN 6
(HLR)
    priority: 1,
    description: "SMS إلى البريطاني HLR"
  },

  # دون إعادة الكتابة (SSN 6) توجيه حركة الصوت للأرقام البريطانية
  %{
    gt_prefix: "44",
    source_ssn: 6,                  # مطابقة SSN 6 الواردة (HLR)
    peer_id: 1,
    dest_ssn: nil,                  # لا إعادة كتابة SSN
    priority: 1,
    description: "حركة الصوت البريطانية"
  }
]
```

---

# اختبار تكوين توجيه STP

بعد تكوين الأقران والمسارات، تحقق من تكوينك:

## 1. تحقق من حالة النظير

**عبر واجهة الويب:**

- انتقل إلى http://localhost
- تحقق من صفحة حالة M3UA
- تحقق من أن الأقران تظهر **الحالة: نشطة**

**عبر وحدة IEx:**

```
# الحصول على جميع حالات الأقران
M3UA.STP.get_peers_status()

# الناتج المتوقع:
# [
#   %{peer_id: 1, name: "Partner_STP_West", status: :active,
point_code: 100, ...},
#   %{peer_id: 2, name: "Local_HLR", status: :active, point_code:
200, ...}
# ]
```

## 2. اختبار توجيه رمز النقطة

```
# اختبار إلى DPC 100 إرسال رسالة M3UA
test_payload = <<1, 2, 3, 4>>  # حمولة وهمية
M3UA.STP.route_by_pc(100, test_payload, 0)

# تحقق من السجلات لقرار التوجيه
# الناتج المتوقع: OPC=... -> DPC=100 1 عبر النظير "توجيه الرسالة"
```

## 3. اختبار توجيه العنوان العالمي

```
# ابحث يدويًا عن مسار GT
M3UARouting.lookup_peer_by_gt("
```

# دليل واجهة المستخدم على الويب

← العودة إلى الوثائق الرئيسية

**واجهة المستخدم على الويب** يوفر هذا الدليل وثائق شاملة لاستخدام (واجهة Phoenix LiveView) لـ OmniSS7.

## جدول المحتويات

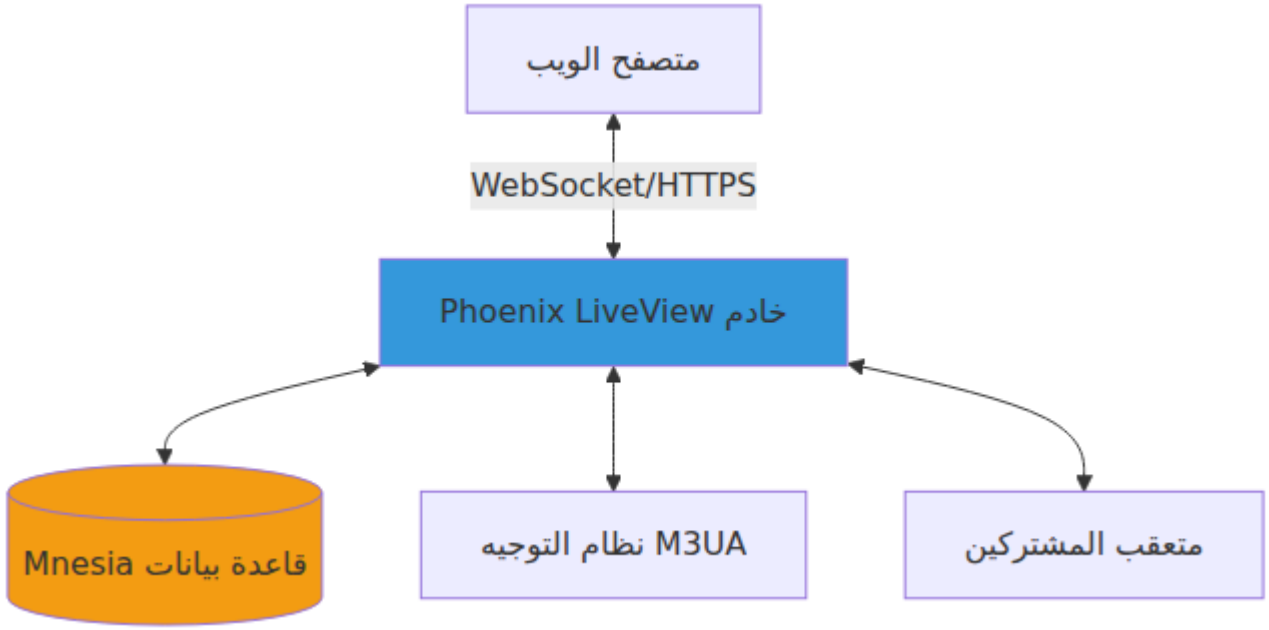
1. نظرة عامة
2. الوصول إلى واجهة المستخدم على الويب
3. صفحة إدارة التوجيه
4. صفحة المشتركين النشطين
5. العمليات الشائعة
6. سلوك التحديث التلقائي

---

## نظرة عامة

**Phoenix LiveView** يوفر قدرات واجهة المستخدم على الويب لـ OmniSS7 هي تطبيق المراقبة والإدارة في الوقت الحقيقي. تعتمد الصفحات المتاحة على وضع التشغيل النشط (STP، HLR، أو SMSc).

## بنية واجهة المستخدم على الويب




## تكوين الخادم

- **البروتوكول**: HTTPS
- **المنفذ**: 443 (مُعد في `config/runtime.exs`)
- **عنوان IP الافتراضي**: 0.0.0.0 (يستمع على جميع الواجهات)
- **الشهادات**: تقع في `priv/cert/`

**رابط الوصول**: `https://[server-ip]:443`

---

# الوصول إلى واجهة المستخدم على الويب

## المتطلبات المسبقة

1. **شهادات SSL**: تأكد من وجود شهادات SSL صالحة في `priv/cert/`:
   - `omnitouch.crt` - ملف الشهادة
   - `omnitouch.pem` - ملف المفتاح الخاص
2. **تشغيل التطبيق**: ابدأ التطبيق باستخدام `iex -S mix`

3. **جدار الحماية**: تأكد من فتح المنفذ 443 لحركة مرور HTTPS

## الصفحات المتاحة حسب الوضع

| الصفحة | وضع STP | وضع HLR | وضع SMSc | الوصف |
|---|---|---|---|---|
| **أحداث SS7** | ✅ | ✅ | ✅ | تسجيل الأحداث والتقاط رسائل SCCP |
| **عميل SS7** | ✅ | ✅ | ✅ | اختبار العمليات اليدوية لـ MAP |
| **M3UA** | ✅ | ✅ | ✅ | حالة اتصال M3UA |
| **التوجيه** | ✅ | ✅ | ✅ | إدارة جدول توجيه M3UA |
| **اختبار التوجيه** | ✅ | ✅ | ✅ | اختبار التوجيه والتحقق منه |
| **روابط HLR** | ✅ | ✅ | ✅ | حالة واجهة برمجة التطبيقات وإدارة المشتركين HLR |
| **المشتركين النشطين** | ✅ | ✅ | ✅ | تتبع موقع المشتركين في الوقت الحقيقي (HLR) |
| **روابط SMSc** | ✅ | ✅ | ✅ | حالة واجهة برمجة التطبيقات وإدارة الطوابير SMSc |
| **مشتركي SMSc** | ✅ | ✅ | ✅ | تتبع المشتركين في الوقت الحقيقي (SMSc) |
| **التطبيق** | ✅ | ✅ | ✅ | موارد النظام والمراقبة |
| **التكوين** | ✅ | ✅ | ✅ | عارض التكوين |

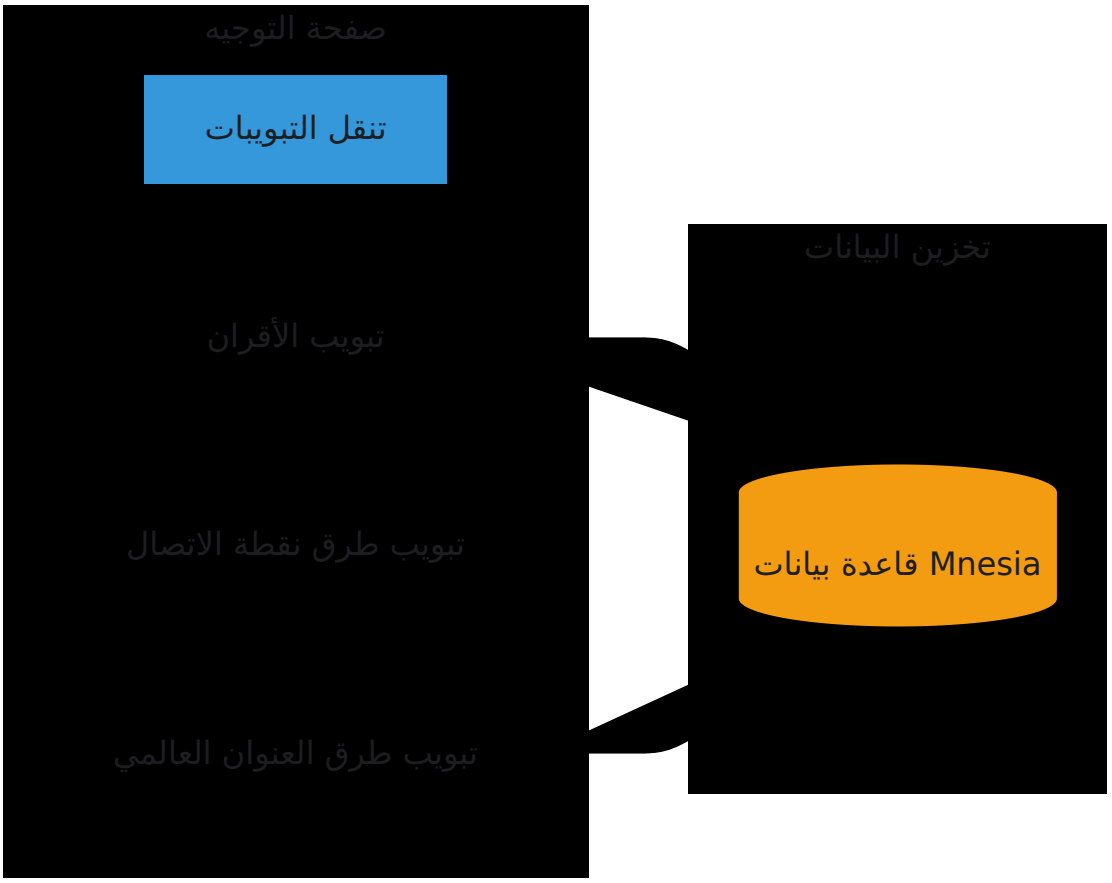
---

# صفحة إدارة التوجيه

**الصفحة**: `/routing`
**الأوضاع**: STP، SMSc
**التحديث التلقائي**: كل 5 ثوانٍ

توفر صفحة إدارة التوجيه واجهة تبويب لإدارة جداول توجيه M3UA.

## تخطيط الصفحة




## تبويب الأقران

إدارة اتصالات M3UA للأقران (STPs، HLRs، MSCs، SMSCs).

### أعمدة جدول الأقران

| العمود | الوصف | المثال |
|---|---|---|
| **ID** | معرف فريد للقرين | `1` |
| **الاسم** | اسم قرين قابل للقراءة البشرية | `"STP_West"` |
| **الدور** | دور الاتصال | `client`، `server`، `stp` |
| **نقطة الاتصال** | نقطة الاتصال SS7 للقرين | `100` |
| **عن بعد** | عن بعد IP:Port | `10.0.0.10:2905` |
| **الحالة** | حالة الاتصال | `active`، `aspup`، `down` |
| **الإجراءات** | أزرار تعديل/حذف | - |

**إضافة قرين**

1. **انقر** على تبويب الأقران
2. **املأ** حقول النموذج:
   - **معرف القرين**: يتم إنشاؤه تلقائيًا إذا ترك فارغًا
   - **اسم القرين**: اسم وصفي (مطلوب)
   - **الدور**: اختر `client`، `server`، أو `stp`
   - **نقطة الاتصال**: نقطة الاتصال SS7 (مطلوب)
   - **IP المحلي**: عنوان IP لنظامك
   - **المنفذ المحلي**: 0 لتعيين منفذ ديناميكي
   - **IP عن بعد**: عنوان IP للقرين
   - **المنفذ عن بعد**: منفذ القرين (عادةً 2905)
   - **سياق التوجيه**: معرف سياق توجيه M3UA
   - **مؤشر الشبكة**: `international` أو `national`
3. **انقر** على "إضافة قرين"

**الاستمرارية**: يتم حفظ القرين على الفور في Mnesia ويستمر بعد إعادة التشغيل.

**تعديل قرين**

1. **انقر** على زر "تعديل" في صف القرين
2. **عدل** حقول النموذج حسب الحاجة
3. **انقر** على "تحديث القرين"

**ملاحظة**: إذا قمت بتغيير معرف القرين، سيتم حذف القرين القديم وإنشاء قرين جديد.

**حذف قرين**

1. **انقر** ع��ى زر "حذف" في صف القرين
2. **أكد** الحذف (سيتم أيضًا إزالة جميع الطرق التي تستخدم هذا القرين)

**مؤشرات حالة القرين**

| الحالة | اللون | الوصف |
|---|---|---|
| `active` | أخضر  | القرين متصل ويقوم بتوجيه الرسائل |
| `aspup` | أصفر  | متصل ولكن ليس نشطًا بعد ASP |
| `down` | أحمر  | القرين غير متصل |

---

# تبويب طرق نقطة الاتصال

تكوين قواعد التوجيه بناءً على نقاط الاتصال الوجهة.

**أعمدة جدول الطرق**

| العمود | الوصف | المثال |
| --- | --- | --- |
| **نقطة الاتصال الوجهة** | نقطة الاتصال المستهدفة (بتنسيق zone.area.id) | `1.2.3` (100) |
| **القناع** | قناع الشبكة لمطابقة نقطة الاتصال | `/8`، (مطابقة دقيقة) `/14` (نطاق) |
| **معرف القرين** | القرين المستهدف لهذه الطريقة | `1` |
| **اسم القرين** | اسم القرين المستهدف | `"STP_West"` |
| **الأولوية** | أولوية الطريقة (1 = الأعلى) | `1` |
| **الشبكة** | مؤشر الشبكة | `international` |
| **الإجراءات** | أزرار تعديل/حذف | - |

**إضافة طريقة نقطة الاتصال**

1. **انقر** على تبويب "طرق نقطة الاتصال"
2. **املأ** حقول النموذج:
   - **نقطة الاتصال الوجهة**: أدخل كـ `zone.area.id` (مثل، `1.2.3`) أو كعدد صحيح (0-16383)
   - **القناع**: اختر القناع `/14` للمطابقة الدقيقة، وقيم أقل للنطاقات
   - **معرف القرين**: اختر القرين المستهدف من القائمة المنسدلة
   - **الأولوية**: أدخل الأولوية (1 = الأعلى، الافتراضي)
   - **مؤشر الشبكة**: اختر `national` أو `international`
3. **انقر** على "إضافة طريقة"

**تنسيق نقطة الاتصال**: يمكنك إدخال نقاط الاتصال في تنسيقين:

- **تنسيق 3-8-3**: `zone.area.id` (مثل، `1.2.3`)
- **تنسيق عدد صحيح**: `0-16383` (مثل، `1100`)

يقوم النظام بتحويل التنسيقات تلقائيًا.

**فهم الأقنعة**

نقاط الاتصال هي قيم 14 بت (0-16383). يحدد القناع عدد البتات الأكثر أهمية التي يجب أن تتطابق:

| القناع | ��نقاط الاتصال المتطابق | حالة الاستخدام |
|---|---|---|
| `/14` | 1 (مطابقة دقيقة) | توجيه إلى وجهة محددة |
| `/13` | 2 نقاط اتصال | نطاق صغير |
| `/8` | 64 نقطة اتصال | نطاق متوسط |
| `/0` | جميع 16,384 نقطة اتصال | **طريق افتراضي/احتياطي** |

**أمثلة**:

- `PC 1000 /14` → يتطابق فقط مع PC 1000
- `PC 1000 /8` → يتطابق مع PC 1000-1063 (64 نقطة اتصال متتالية)
- `PC 0 /0` → يتطابق مع جميع نقاط الاتصال (طريق افتراضي)

**بطاقة مرجعية لقناع نقطة الاتصال**

تتضمن صفحة الويب مرجعًا تفاعليًا يعرض جميع قيم الأقنعة ونطاقاتها.

---

# تبويب طرق العنوان العالمي

تكوين قواعد التوجيه بناءً على عناوين العنوان العالمي SCCP.

**المتطلبات**: يجب تمكين توجيه العنوان العالمي في التكوين:

```
config :omniss7,
  enable_gt_routing: true
```

**أعمدة جدول الطرق**

| العمود | الوصف | المثال |
|---|---|---|
| **بادئة GT** | بادئة GT للجهة المتصلة (فارغ = احتياطي) | `"1234"`، `""` |
| **المصدر SSN** | SSN المطابقة على للجهة المتصلة (اختياري) | `6` (HLR)، `any` |
| **معرف القرين** | القرين المستهدف | `1` |
| **القرين** | اسم القرين | `"HLR_West (1)"` |
| **الوجهة SSN** | عند التوجيه (اختياري) SSN إعادة كتابة | `6`، `preserve` |
| **الأولوية** | أولوية الطريقة | `1` |
| **الوصف** | وصف الطريقة | `"أرقام الولايات المتحدة"` |
| **الإجراءات** | أزرار تعديل/حذف | - |

**إضافة طريقة عنوان عالمي**

1. على تبويب "طرق العنوان العالمي" **انقر**
2. حقول النموذج **املأ**:
   - **بادئة GT**: (مثل، `"1234"`) اترك فارغًا للطريق الاحتياطي، أو أدخل أرقام
   - **المصدر SSN**: اختياري - تصفية حسب SSN للجهة المتصلة
   - **معرف القرين**: اختر القرين المستهدف
   - **الوجهة SSN**: اختياري - إعادة كتابة SSN عند التوجيه
   - **الأولوية**: أولوية الطريقة (1 = الأعلى)
   - **الوصف**: وصف قابل للقراءة البشرية
3. على "إضافة طريقة" **انقر**

**طرق الاحتياطي**: إذا كانت بادئة GT فارغة، فإن الطريقة تعمل كطريق شامل للـ GTs التي لا تتطابق مع أي طريقة أخرى.

**قيم SSN الشائعة**

تتضمن الصفحة بطاقة مرجعية بقيم SSN الشائعة:

| SSN | عنصر الشبكة |
|---|---|
| 6 | HLR (سجل الموقع المنزلي) |
| 7 | VLR (سجل الموقع الزائر) |
| 8 | MSC (مركز تبديل الهاتف المحمول) |
| 9 | EIR (سجل هوية المعدات) |
| 10 | AUC (مركز المصادقة) |
| 142 | RANAP |
| 145 | gsmSCF (وظيفة التحكم في الخدمة) |
| 146 | SGSN |

**إعادة كتابة SSN**

- **المصدر** SSN للجهة المتصلة في الرسائل الواردة: المطابقة على **SSN**
- **الوجهة** SSN للجهة المتصلة عند التوجيه: إذا تم تعيينه، يعيد كتابة **SSN**
  - الأصلي SSN فارغ = الحفاظ على
  - SSN قيمة = الاستبدال بهذه

**حالة الاستخدام**: توجيه الرسائل مع SSN=7 إلى قرين، وإعادة كتابتها إلى SSN=6 (HLR) على الجانب الخارج (VLR).

---

# استمرارية جدول التوجيه

**جميع الطرق مخزنة في Mnesia تمر بعد إعادة تشغيل التطبيق وت��.**

**كيف تستمر الطرق**

1. **تغييرات واجهة المستخدم على الويب**: يتم حفظ جميع عمليات الإضافة/التعديل/الحذف على الفور في Mnesia
2. **إعادة تشغيل التطبيق**: يتم تحميل الطرق من Mnesia عند بدء التشغيل
3. **دمج runtime.exs**: يتم دمج الطرق الثابتة من `config/runtime.exs` مع طرق Mnesia (بدون تكرارات)

**أولوية الطرق**

عندما تتطابق عدة طرق مع وجهة:

1. **الأكثر تحديدًا أولاً**: القيم الأعلى للقناع (الأكثر تحديدًا) تأخذ الأولوية
2. **حقل الأولوية**: الأرقام الأقل للأولوية تمر أولاً (1 = أعلى أولوية)
3. **حالة القرين**: يتم استخدام الطرق فقط للأقران `النشطين`

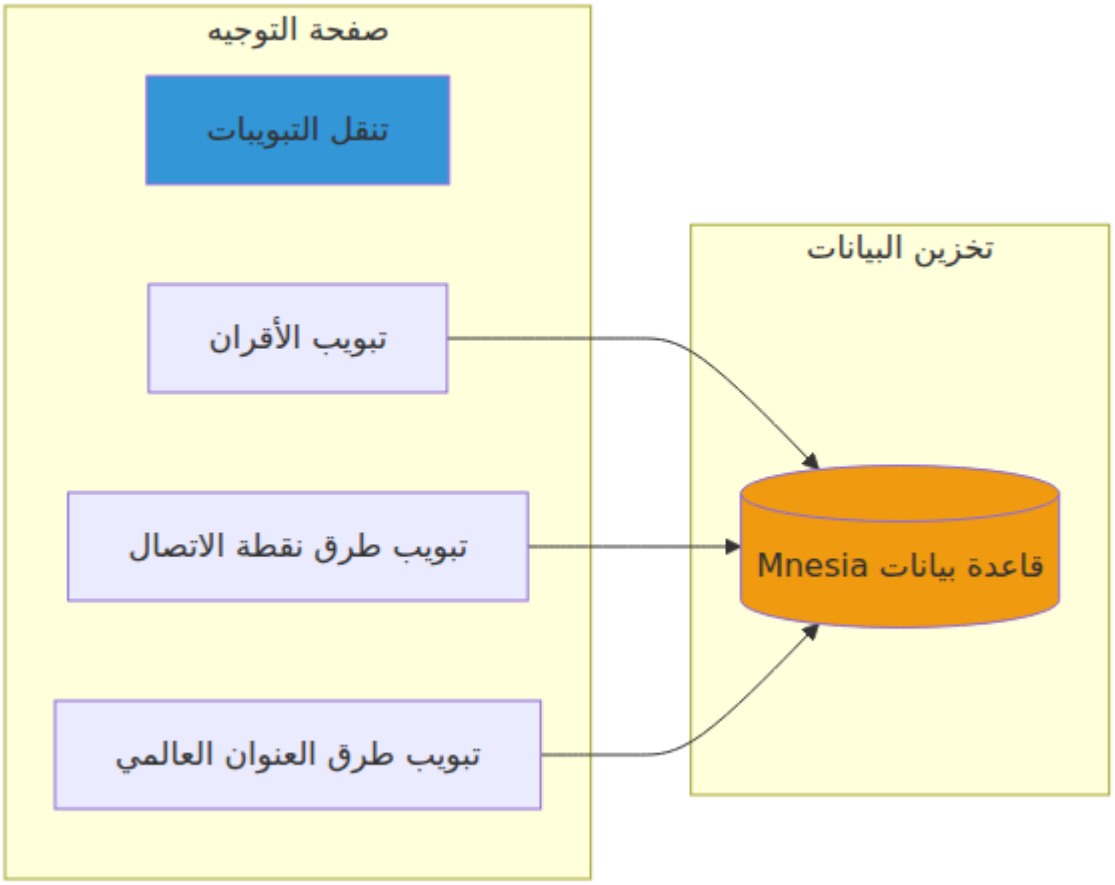
---

# المشتركين النشطين

**الصفحة**: `/subscribers`
**الوضع**: فقط HLR
**التحديث التلقائي**: كل 2 ثانيتين

يعرض تتبعًا في الوقت الحقيقي للمشتركين الذين أرسلوا طلبات UpdateLocation.

## ميزات الصفحة




## أعمدة جدول المشتركين

| العمود | الوصف | المثال |
|---|---|---|
| **IMSI** | IMSI المشترك | `"50557123456789"` |
| **رقم VLR** | عنوان GT VLR الحالي | `"555123155"` |
| **رقم MSC** | عنوان GT MSC الحالي | `"555123155"` |
| **تم التحديث في** | آخر طابع زمني لـ UpdateLocation | `"2025-10-25 14:23:45 UTC"` |
| **المدة** | الوقت منذ التسجيل | `"2h 15m 34s"` |

## ملخص الإحصائيات

عندما يكون هناك مشتركين، تعرض بطاقة ملخص:

- **إجمالي النشطين**: إجمالي عدد المشتركين المسجلين
- **VLRs الفريدة**: عدد عناوين VLR المتميزة
- **MSCs الفريدة**: عدد عناوين MSC المتميزة

## مسح المشتركين

**زر مسح الكل**: يزيل جميع المشتركين النشطين من المتعقب.

**التأكيد**: يتطلب تأكيدًا قبل المسح (لا يمكن التراجع عنه).

**حالة الاستخدام**: مسح سجلات المشتركين القديمة بعد صيانة الشبكة أو الاختبار.

## التحديث التلقائي

تقوم الصفحة بالتحديث تلقائيًا كل **2 ثانية** لعرض تحديثات المشتركين في الوقت الحقيقي.

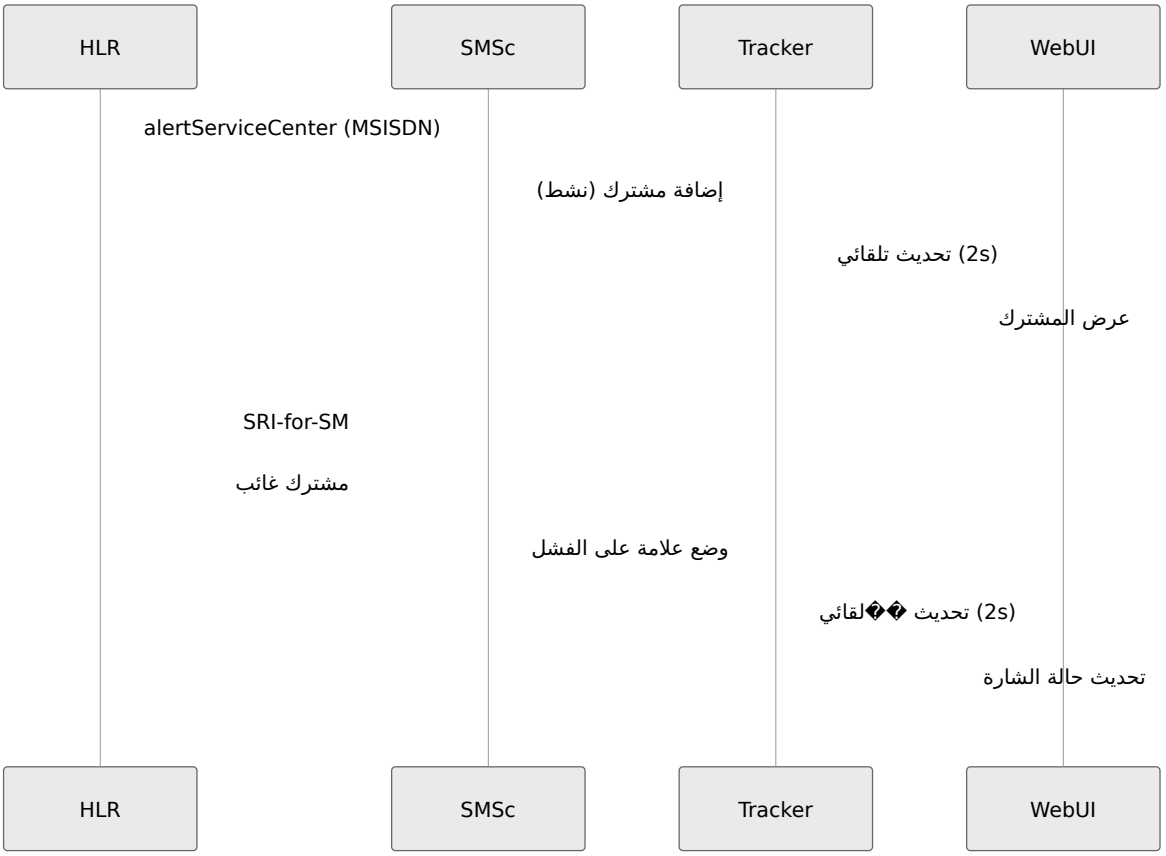
---

# SMSc مشتركي

**الصفحة**: `/smsc_subscribers`
**الوضع**: SMSc فقط
**التحديث التلقائي**: كل 2 ثانيتين

يعرض تتبعًا في الوقت الحقيقي للمشتركين بناءً على رسائل alertServiceCenter المستلمة من HLRs، وحالة تسليم الرسائل، وتتبع الفشل.

# ميزات الصفحة



## أعمدة جدول المشتركين

| العمود | الوصف | المثال |
|---|---|---|
| **MSISDN** | رقم هاتف المشترك | `"15551234567"` |
| **IMSI** | IMSI المشترك | `"001010123456789"` |
| **HLR GT** | HLR GT الذي أرسل alertServiceCenter | `"15551111111"` |
| **الرسائل المرسلة** | عدد رسائل MT-FSM المرسلة | `5` |
| **الرسائل المستلمة** | عدد رسائل MO-FSM المستلمة | `2` |
| **الحالة** | `نشط` أو `فشل` (ملونة) | `● نشط` |
| **آخر تحديث** | آخر طابع زمني للتحديث | `"2025-10-30 14:23:45 UTC"` |
| **المدة** | الوقت منذ آخر تحديث | `"15m 34s"` |

## مؤشرات الحالة

- ● **نشط** (أخضر): المشترك قابل للوصول، آخر alertServiceCenter تم استلامه بنجاح
- ○ **فشل** (أحمر): آخر محاولة تسليم فشلت خطأ (أو مشترك غائب SRI-for-SM)

## ملخص الإحصائيات

عندما يكون هناك مشتركين، تعرض بطاقة ملخص:

- **إجمالي المتعقبين**: إجمالي عدد المشتركين المتعقبين
- **نشط**: عدد المشتركين بحالة نشطة

- **فشل**: عدد المشتركين بحالة فشل
- **HLRs الفريدة**: عدد HLRs المتميزة التي ترسل التنبيهات

## إدارة المشتركين

**زر إزالة**: يزيل مشتركًا فرديًا من المتابعة.

**زر مسح الكل**: يزيل جميع المشتركين المتعقبين.

**التأكيد**: يتطلب مسح الكل تأكيدًا قبل المسح (لا يمكن التراجع عنه).

**حالة الاستخدام**:

- إزالة الإدخالات القديمة بعد مشاكل الشبكة
- مسح بيانات الاختبار بعد التطوير
- مراقبة HLRs التي ترسل التنبيهات

## عدادات الرسائل

يقوم المتعقب بزيادة العدادات تلقائيًا:

- **الرسائل المرسلة**: تزداد عندما تنجح MT-FSM ويتم إرسال SRI-for-SM
- **الرسائل المستلمة**: تزداد عندما يتم استلام MO-FSM من المشترك

## التحديث التلقائي

تقوم الصفحة بالتحديث تلقائيًا كل **2 ثانية** لعرض تحديثات المشتركين والحالة في الوقت الحقيقي.

---

# العمليات الشائعة

## البحث والتصفية

حاليًا�� لا تتضمن واجهة المستخدم على الويب وظيفة بحث/تصفية مدمجة. للعثور على طرق معينة:

1. استخدم وظيفة البحث في متصفحك (Ctrl+F / Cmd+F)

2. ابحث عن أسماء الأقران، نقاط الاتصال، أو بادئات GT

## العمليات الجماعية

لإجراء تغييرات جماعية على الطرق:

1. للوصول البرمجي REST **الخيار 1**: استخدم واجهة برمجة التطبيقات
2. وأعد تشغيل التطبيق `config/runtime.exs` **الخيار 2**: عدل
3. **الخيار 3**: استخدم واجهة المستخدم على الويب لإجراء تغييرات فردية على الطرق

## التصدير/الاستيراد

**ملاحظة**: لا تدعم واجهة المستخدم على الويب حاليًا تصدير أو استيراد جداول التوجيه. الطرق هي:

- Mnesia مخزنة في ملفات قاعدة بيانات
- مُعدة في `config/runtime.exs`

لعمل نسخة احتياطية من الطرق:

1. **Mnesia**: قم بعمل نسخة احتياطية من دليل `Mnesia.{node_name}/`
2. **التكوين**: تحكم في إصدار `config/runtime.exs`

---

# سلوك التحديث التلقائي

تمتل�ك الصفحات المختلفة فترات تحديث مختلفة:

| الصفحة | فترة التحديث | السبب |
|---|---|---|
| إدارة التوجيه | 5 ثوانٍ | تغييرات الطرق نادرة |
| المشتركين النشطين | 2 ثانيتين | تتغير حالة المشتركين بشكل متكرر |
| حالة M3UA | تختلف حسب الصفحة | مراقبة حالة الاتصال |

**اتصال WebSocket**: تستخدم جميع الصفحات اتصالات WebSocket لـ Phoenix LiveView للتحديثات في الوقت الحقيقي.

**انقطاع الشبكة**: إذا فقد اتصال WebSocket، ستقوم الصفحة بمحاولة إعادة الاتصال تلقائيًا.

---

# استكشاف الأخطاء وإصلاحها

## الصفحة لا تُحمّل

1. **تحقق من شهادة HTTPS**: تأكد من وجود `priv/cert/omnitouch.crt` و `.pem`
2. **تحقق من المنفذ 443**: تحقق من قواعد جدار الحماية للسماح بحركة مرور HTTPS
3. **تشغيل التطبيق**: تأكد من تشغيل التطبيق باستخدام `iex -S mix`
4. **وحدة تحكم المتصفح**: تحقق من أخطاء شهادة SSL (تحذيرات الشهادات الموقعة ذاتيًا)

## الطر�� لا تستمر

1. **تحقق من تخزين Mnesia**: تحقق من `mnesia_storage_type: :disc_copies` في التكوين
2. **دليل Mnesia**: تأكد من أن دليل Mnesia قابل للكتابة
3. **تحقق من السجلات**: ابحث عن أخطاء Mnesia في سجلات التطبيق

## التحديث التلقائي لا يعمل

1. **اتصال WebSocket**: تحقق من وحدة تحكم المتصفح لأخطاء WebSocket
2. **الشبكة**: تحقق من اتصال الشبكة المستقر
3. **إعادة تحميل الصفحة**: حاول تحديث الصفحة (F5)

---

# الوثائق ذات الصلة

- **دليل STP** - تكوين التوجيه بالتفصيل

- **دليل HLR** - إدارة المشتركين
- **دليل API** - واجهة برمجة التطبيقات REST للوصول البرمجي
- **مرجع التكوين** - جميع معلمات التكوين

---

# الملخص

إدارة بديهية وفي الوقت الحقيقي لجداول التوجيه OmniSS7 توفر واجهة المستخدم على الويب لـ وتتبع المشتركين:

 **تحديثات في الوقت الحقيقي** - التحديث التلقائي يبقي البيا��ات محدثة
 **تخزين مستمر** - تضمن Mnesia بقاء الطرق بعد إعادة التشغيل
 **واجهة مستخدم قائمة على الدور** - تتكيف الصفحات مع وضع التشغيل (STP/HLR/SMSc)
 **إدارة تفاعلية** - إضافة وتعديل وحذف الطرق دون إعادة التشغيل
 **مراقبة الحالة** - حالة الاتصال والقرين مباشرة

لإجراء عمليات متقدمة أو الأتمتة، راجع دليل API.