المعايير

يحتوي هذا الدليل على معايير الأداء لنظام SMS-C باستخدام Benchee.

المعايير المتاحة

1. معيار SMS الخام (`raw_sms_bench.exs`)

يقوم بمعايرة نقطة نهاية API `submit_message_raw` باستخدام PDUs SMS حقيقية.

الميزات:

- يستخدم PDUs SMS حقيقية (أضف PDUs الخاصة بك إلى قائمة `@sample_pdus` في الملف)
- يعطل الكشف عن التكرار عن طريق مسح بصمات الأصابع قبل كل تكرار
- يخرج تقارير لكل من وحدة التحكم و HTML

الاستخدام:

```
mix run benchmarks/raw_sms_bench.exs
```

الإخراج: `benchmarks/output/raw_sms_benchmark.html`

2. معيار API الرسائل (`message_api_bench.exs`)

يقوم بمعايرة عمليات API الرسائل المختلفة بما في ذلك الإدراج والاسترجاع والتوجيه.

الميزات:

- يختبر insert_message (بسيط ومع التوجيه)
- يختبر get_messages_for_smsc
- يختبر list_message_queues
- يملأ قاعدة البيانات مسبقًا ببيانات اختبار لسيناريوهات واقعية

الاستخدام:

```
mix run benchmarks/message_api_bench.exs
```

الإخراج: `benchmarks/output/message_api_benchmark.html`

التكوين

تستخدم جميع المعايير Benchee مع الإعدادات الافتراضية التالية:

- التسخين: 2 ثانية
- الوقت: 10 ثوانٍ
- وقت الذاكرة: 2 ثانية
- إحصائيات موسعة مفعلة
- تقارير HTML يتم إنشاؤها تلقائيًا

المخرجات

يتم إنشاء تقارير معيار HTML في `benchmarks/output/` وتحتوي على:

- مقاييس أداء مفصلة
- مخططات مقارنة
- إحصائيات استخدام الذاكرة
- تحليل إحصائي

وثائق عمليات SMS-C

← العودة إلى README الرئيسي

مرحبًا بكم في وثائق عمليات SMS-C. تغطي هذه الدليل الشامل جميع جوانب تكوين وتشغيل ومراقبة واستكشاف نظام SMS-C.

نظرة عامة على الوثائق

البدء

- **مرجع التكوين** - خيارات التكوين الكاملة والأمثلة

العمليات اليومية

- **دليل العمليات** - المهام اليومية، المراقبة، والصيانة
- **دليل توجيه SMS** - إدارة التوجيه والتكوين
- **بحث مشترك HSS** - واجهة Diameter Sh لاكتشاف المشتركين على الشبكة
- **الاتحاد الجغرافي** - توزيع متعدد المواقع يعتمد على DNS SRV مع اكتشاف HTTP
- **مرجع API** - وثائق API الكاملة مع الأمثلة

الأداء والمراقبة

- **تحسين الأداء** - تحسين للأحمال المختلفة
- **دليل القياسات** - قياسات Prometheus والمراقبة

استكشاف الأخطاء

- **دليل استكشاف الأخطاء** - المشكلات الشائعة والحلول

الامتثال والتنظيم

- **امتثال الاعتراض ANSSI R226** - المواصفات الفنية للاعتراض القانوني الفرنسية
  - تكامل الواجهة الأمامية متعددة البروتوكولات (IMS/SIP، SMPP، SS7/MAP)

- واجهات الاعتراض القانوني ETSI X1/X2/X3
- بنية تخزين من مستويين Mnesia + SQL
- مخطط CDR لاستعلامات الاعتراض القانوني
- قدرات التشفير والتحليل الكمي

روابط سريعة

المهام الشائعة

- إرسال رسالة
- إنشاء مسار
- التحقق من حالة الرسالة
- مراقبة صحة النظام
- التعامل مع فشل التسليم

أمثلة التكوين

- تخزين الرسائل والاحتفاظ بها
- إعد��د تصدير CDR
- ضوابط الخصوصية
- تكوين عالي الحجم
- الاتحاد الجغرافي
- التوجيه الجغرافي
- توازن الحمل
- إعداد Diameter Sh / HSS
- إعداد ENUM/NAPTR
- شحن OCS
- ترجمة الأرقام

المراقبة والتنبيهات

- المقاييس الرئيسية
- التنبيهات الموصى بها

- قوالب لوحة التحكم

نظرة عامة على بنية النظام

يعد SMS-C منصة توجيه رسائل موزعة وعالية الأداء مع المكونات الرئيسية التالية:

المكونات الأساسية

- **تخزين الرسائل** - تخزين سريع يعتمد على Mnesia مع احتفاظ قابل للتكوين وتصدير CDR
- **محرك التوجيه** - قواعد توجيه تعتمد على Mnesia مع مطابقة بادئة وتوازن الحمل
- **ترجمة الأرقام** - تطبيع الأرقام يعتمد على Regex مع ترتيب الأولويات
- **تكامل الشحن** - شحن OCS عبر الإنترنت مع سياسات قائمة على المسار
- **بحث ENUM** - توجيه الأرقام يعتمد على DNS مع التخزين المؤقت
- **تسجيل الأحداث** - تتبع دورة حياة الرسالة
- **تصدير CDR** - تصدير تلقائي إلى قاعدة بيانات SQL للفوترة/التحليلات على المدى الطويل

الواجهات الخارجية

- **REST API** - إرسال الرسائل وإدارتها (HTTPS)
- **واجهة الويب** - إدارة المسارات، متصفح الرسائل، المراقبة
- **Prometheus** - عرض القياسات للمراقبة
- **OCS** - تكامل الشحن/الفوترة
- **DNS** - عمليات بحث ENUM/NAPTR للتوجيه

التوزيع والموثوقية العالية

- **الاتحاد الجغرافي** - توزيع متعدد المتحكمين يعتمد على DNS SRV مع اكتشاف HTTP
- **طابور الإرسال** - طابور رسائل تلقائي وإعادة المحاول�ة عند عدم الوصول إلى المتحكمين البعيدين
- **توازن الحمل** - توزيع المسارات بناءً على الوزن

الوثائق ذات الصلة

- **معايير الأداء** - اختبار الأداء والنتائج
- **مرجع مخطط CDR** - مخطط قاعدة بيانات CDR الكامل مع أمثلة SQL

متطلبات النظام

المتطلبات الدنيا

- **CPU**: 2 نوى
- **RAM**: 4 جيجابايت
- **القرص**: 50 جيجابايت (يزداد مع الاحتفاظ بالرسائل)
- **نظام التشغيل**: Linux (موصى به)، macOS (تطوير)
- **Erlang/OTP**: 26.x أو أحدث
- **Elixir**: 1.15.x أو أحدث
- **قاعدة بيانات SQL**: MySQL 8.0+، MariaDB 10.5+، أو PostgreSQL 13+ (لتخزين CDR)

الإنتاج الموصى به

- **CPU**: 8+ نوى
- **RAM**: 16+ جيجابايت
- **القرص**: 500+ جيجابايت SSD
- **الشبكة**: 1 Gbps+
- **قاعدة بيانات SQL**: خادم مخصص مع النسخ المتماثل (لتخزين CDR)

منافذ الشبكة

- **80/443** - واجهة الويب (HTTP/HTTPS)
- **8443** - API (HTTPS) وتواصل أقران الاتحاد
- **9568** - قياسات Prometheus

الدعم والموارد

السجلات

- **سجلات التطبيق**: `/var/log/sms_c/` (الإنتاج) (التطوير) أو وحدة التحكم
- **سجلات واجهة الويب**: عارض السجلات في الوقت الحقيقي في `/logs`
- **سجلات الأحداث**: تتبع الأحداث لكل رسالة عبر API

التشخيصات

- **فحص الصحة**: `GET /api/status`
- **القياسات**: `GET http://localhost:9568/metrics` (تنسيق Prometheus)
- **حالة الواجهة الأمامية**: واجهة الويب في `/frontend_status`
- **طابور الرسائل**: واجهة الويب في `/message_queue`

الحصول على المساعدة

1. تحقق من دليل استكشاف الأخطاء
2. راجع سجلات التطبيق
3. تحقق من قياسات Prometheus للبحث عن الشذوذ
4. استخدم محاكي التوجيه لاختبار منطق التوجيه
5. افحص سجلات الأحداث لكل رسالة

معلومات الإصدار

تكون هذه الوثائق سارية اعتبارًا من:

- **آخر تحديث**: 30-10-2025
- **إصدار SMS-C**: أحدث إصدار تطوير
- **Elixir المدعوم**: 1.15.x - 1.17.x
- **Erlang/OTP المدعوم**: 26.x - 27.x

اتفاقيات الوثائق

طوال هذه الوثائق:

- **أمثلة التكوين** تظهر القيم النموذجية؛ قم بتعديلها لتناسب بيئتك
- **أمثلة API** تستخدم تنسيق سطر الأوامر `curl`
- **عناوين IP والنطاقات** هي أمثلة فقط؛ استبدلها بقيمك الفعلية
- **أسماء القياسات** تتبع اتفاقيات تسمية Prometheus
- **جميع الطوابع الزمنية** بتوقيت UTC ما لم يُذكر خلاف ذلك

البدء السريع

1. **التكوين**: قم بالتكوين عبر `config/runtime.exs` - انظر مرجع التكوين
2. **المسارات الأولية**: أنشئ قواعد التوجيه عبر واجهة الويب أو ملف التكوين - انظر دليل توجيه SMS
3. **إرسال رسالة اختبار**: استخدم API أو واجهة الويب - انظر مرجع API
4. **المراقبة**: قم بإعداد سحب Prometheus - انظر دليل القياسات

ملاحظات على الوثائق

تُحافظ هذه الوثائق جنبًا إلى جنب مع قاعدة شفرة SMS-C. يرجى، لإجراء تصحيحات أو تحسينات، تحديث ملفات markdown في دليل `docs/`.

وثائق الامتثال للتنصت ANSSI R226

الغرض من الوثيقة: توفر هذه الوثيقة المواصفات الفنية المطلوبة للحصول على تفويض ANSSI من قانون العقوبات الفرنسي لمركز خدمات الرسائل R226-7 و R226-3 بموجب المواد R226 القصيرة OmniMessage (SMSc).

التصنيف: وثائق الامتثال التنظيمي

السلطة المستهدفة: الوكالة الوطنية لأمن نظم المعلومات (ANSSI)

التنظيم: حماية خصوصية المراسلات والتنصت القانوني - R226

---

1. المواصفات الفنية التفصيلية

1.1 ورقة البيانات الفنية التجارية

اسم المنتج: مركز خدمات الرسائل القصيرة OmniMessage (SMSc)
نوع المنتج: مركز رسائل الاتصالات
الوظيفة الأساسية: توجيه الرسائل القصيرة، التخزين، والتسليم
بروتوكولات الشبكة: REST API (HTTPS)، بروتوكولات الرسائل القصيرة (SMPP، IMS، SS7/MAP عبر الواجهات الخارجية)
نموذج النشر: تطبيق خادم محلي
تكنولوجيا المكدس: Elixir/Erlang، إطار عمل Phoenix، Mnesia، MySQL/PostgreSQL

القدرات الأساسية

معالجة الرسائل:

- قائمة انتظار رسائل قصيرة مركزية مع REST API
- تصميم غير مرتبط بالبروتوكول يدعم واجهات SMPP، IMS، SS7/MAP
- محرك توجيه ديناميكي مع توجيه قائم على البادئات
- منطق إعادة المحاولة مع زيادة زمن الانتظار

- معالجة انتهاء صلاحية الرسائل وقائمة الرسائل الميتة
- وأرشفتها (CDR) توليد سجلات تفاصيل المكالمات
- الأداء: ~1,750 رسالة/ثانية معدل الإدخال، سعة 150 مليون رسالة/يوم

تخزين الرسائل:

- قاعدة بيانات Mnesia في الذاكرة مع إمكانية **قائمة انتظار الرسائل النشطة:** الاستمرارية على القرص
  - التخزين الأساسي: RAM للوصول السريع للغاية (زمن تأخير أقل من مللي ثانية)
  - النسخ الاحتياطي على القرص: `disc_copies` يكتب إلى القرص ضع لاستعادة النظام بعد التعطل
  - الاستعادة التلقائية: تبقى الرسائل بعد إعادة تشغيل النظام
  - الاحتفاظ: قابل للتكوين (الإعداد الافتراضي 24 ساعة)، ثم تنظيف تلقائي
- قاعدة بيانات MySQL/PostgreSQL (منفصلة عن **طويل الأجل:** **أرشيف CDR** قائمة انتظار الرسائل)
  - يتم كتابة CDRs عند تسليم الرسائل، انتهاء صلاحيتها، فشلها، أو رفضها
  - قاعدة بيانات SQL تستخدم فقط لتصدير/أرشفة CDR، وليس لعمليات الرسائل النشطة
  - لا تأثير على الأداء في توجيه الرسائل (كتابة غير متزامنة)
- **فوائد العمارة ذات الطبقتين:**
  - قائمة الانتظار النشطة: سريعة للغاية (1,750 رسالة/ثانية) بدون عنق زجاجة SQL
  - أرشيف CDR: احتفاظ طويل الأجل (أشهر/سنوات) للفوترة والتنصت القانوني
  - فصل نظيف: عمليات الرسائل لا تلمس SQL أبداً
- دعم العنقود للتوافر العالي (عبر العقد Mnesia تكرار)

واجهات الشبكة:

- **REST API:** HTTPS (المنفذ 8443) للتواصل مع الواجهات الخارجية
- **لوحة التحكم:** HTTPS (المنفذ 8086) للإدارة عبر الويب
- **بروتوكولات الواجهة:** SMPP، IMS، SS7/MAP (عبر تطبيقات البوابة الخارجية)
- **قاعدة البيانات:** MySQL/PostgreSQL لتخزين CDR

التوجيه والمعالجة:

- توجيه ديناميكي للرسائل القصيرة مع تحديثات تكوين في وقت التشغيل
- مطابقة قائمة على البادئات (أرقام الاتصال/المتصل)
- تصفية مصدر SMSc ونوع الرسائل
- توازن الحمل القائم على الأولوية والوزن
- ترجمة الأرقام وتطبيعها
- دعم استعلام DNS ENUM (تعيين رقم E.164)
- قدرات الرد التلقائي وإسقاط الرسائل
- التحكم في الشحن لكل مسار (تكامل CGRates)

تم توثيق العمارة والميزات الكاملة في README.md

1.2 قدرات التنصت

1.2.1 اكتساب الرسائل

التقاط الرسائل القصيرة:

- يقوم مركز خدمات الرسائل القصيرة OmniMessage بمعالجة جميع الرسائل القصيرة بين المشتركين والشبكات الخارجية
- الوصول الكامل إلى بيانات التعريف ومحتوى الرسائل بما في ذلك:
  - MSISDN (رقم الهاتف المحمول) المصدر
  - MSISDN (رقم الهاتف المحمول) الوجهة
  - IMSI (معرف المشترك الدولي للهاتف المحمول) المصدر
  - IMSI الوجهة
  - نص الرسالة (محتوى النص)
  - بيانات PDU (وحدة البيانات البروتوكولية) الخام
  - معلومات TP-DCS (نظام ترميز البيانات)
  - ترميز الرسالة (GSM7، UCS-2، 8-bit، Latin-1)
  - مؤشرات الرسائل متعددة الأجزاء وبيانات إعادة التجميع
  - معلومات رأس بيانات المستخدم (UDH)

اكتساب بيانات التعريف للرسائل:

- سجلات تفاصيل المكالمات الكاملة (CDR) المخزنة في قاعدة البيانات مع:
  - معرف الرسالة (معرف فريد)
  - رقم الاتصال (MSISDN المصدر)

    - رقم المتصل (MSISDN الوجهة)
    - الطابع الزمني للإرسال (عندما دخلت الرسالة النظام)
    - الطابع الزمني للتسليم (عندما تم تسليم الرسالة)
    - الطابع الزمن�� لانتهاء الصلاحية (عندما انتهت الرسالة إذا لم يتم تسليمها)
    - الحالة (تم التسليم، انتهت صلاحيتها، فشلت، رُفضت)
    - عدد محاولات التسليم
    - أجزاء الرسالة (المجمعة/متعددة الأجزاء SMS لرسائل)
    - معرف SMSc المصدر
    - معرف SMSc الوجهة
    - العقدة الأصلية (اسم عقدة مجموعة Erlang)
    - العقدة الوجهة (للنشر الموزع)
    - علامة الرسالة الميتة (مؤشر استنفاد إعادة المحاولة)

✅ تم توثيق مخطط CDR الكامل في CDR_SCHEMA.md

الوصول إلى قائمة انتظار الرسائل:

- مراقبة قائمة انتظار الرسائل في الوقت الحقيقي
- نقاط نهاية REST API لاسترجاع الرسائل
- استعلامات قاعدة البيانات للبحث التاريخي عن الرسائل
- قدرات التصفية بواسطة:
    - رقم الهاتف (المصدر/الوجهة)
    - بوابة SMSc
    - نطاق الوقت
    - حالة الرسالة
    - محاولات التسليم

✅ تم توثيق API الكامل في API_REFERENCE.md

1.2.2 قدرات معالجة ا��بيانات

عمارة تخزين الرسائل (نظام ذو طبقتين):

يستخدم SMSc عمارة تخزين متطورة ذات طبقتين تفصل بين معالجة الرسائل التشغيلية وأرشفة طويلة الأجل:

الطبقة 1: قائمة انتظار الرسائل النشطة (Mnesia)

- **الغرض:** عمليات توجيه وتسليم الرسائل في الوقت الحقيقي
- **التكنولوجيا:** قاعدة بيانات Mnesia الموزعة Erlang
- **وضع التخزين:** في الذاكرة مع نسخ احتياطية على القرص
  - التخزين الأساسي في RAM لأقصى سرعة
  - مزامنة تلقائية للقرص لاستعادة النظام بعد التعطل
  - تبقى الرسائل عبر إعادة تشغيل النظام
- **الأداء:** عمليات القراءة/الكتابة أقل من مللي ثانية
- **الاحتفاظ:** قصير الأجل (الإعداد الافتراضي 24 ساعة)، قابل للتكوين
- **التنظيف:** أرشفة تلقائية إلى قاعدة بيانات Mnesia، ثم حذف من CDR
- **العمليات:** جميع عمليات قائمة انتظار الرسائل (إدراج، تحديث، حالة التسليم، توجيه)
- **الميزة الحرجة:** لا يتم استعلام قاعدة البيانات SQL أبداً أثناء توجيه/تسليم الرسائل

الطبقة 2: أرشيف CDR (MySQL/PostgreSQL)

- **الغرض:** التخزين طويل الأجل للفوترة، التحليلات، والتنصت القانوني
- **التكنولوجيا:** قاعدة بيانات SQL التقليدية (MySQL أو PostgreSQL)
- **محفز الكتابة:** يتم كتابة CDRs فقط عندما تصل الرسائل إلى الحالة النهائية:
  - تم تسليم الرسالة بنجاح
  - انتهت صلاحية الرسالة (تجاوزت فترة الصلاحية)
  - فشلت الرسالة بشكل دائم
  - رُفضت الرسالة بواسطة قواعد التوجيه
- **وضع الكتابة:** كتابة غير متزامنة (لا تأثير على أداء توجيه الرسائل)
- **الاحتفاظ:** طويل الأجل (أشهر إلى سنوات)، قابل للتكوين حسب المتطلبات التنظيمية
- **العمليات:** استعلامات تاريخية، تقارير، امتثال، تنصت قانوني
- **الوصول:** استعلامات SQL، REST API (مستقبلاً)، تصدير CSV/JSON

فوائد العمارة المعمارية الرئيسية:

1. **الأداء:** عمليات التوجيه النشطة لا تلمس SQL (لا عنق زجاجة قاعدة البيانات)
2. **القابلية للتوسع:** تتعامل Mnesia مع 1,750+ رسالة/ثانية بدون عبء SQL
3. **الموثوقية:** يضمن وضع `disc_copies` عدم فقدان الرسائل عند التعطل
4. **الامتثال:** توفر قاعدة بيانات CDR مسار تدقيق دائم
5. **فصل الاهتمامات:** البيانات التشغيلية مقابل البيانات الأرشيفية مفصولة بوضوح

دورة حياة الرسالة:

```
1. تم تقديم الرسالة → تم تخزينها في Mnesia (RAM + نسخ احتياطية على القرص)
2. استعلام → تم توجيه الرسالة Mnesia (سريع للغاية)
3. تم تسليم الرسالة/انتهت صلاحيتها → تم كتابة CDR إلى SQL (غير متزامن)
4. بعد 24 ساعة → تم حذف الرسالة من Mnesia (عامل التنظيف)
5. تبقى CDR في SQL → متاحة لاستعلامات التنصت القانوني (سنوات)
```

الاحتفاظ واسترجاع البيانات:

- الاحتفاظ القابل للتكوين بجسم الرسالة أو حذفه من أجل الخصوصية
- الحفاظ على البيانات الثنائية (تخزين PDU الخام في كل من Mnesia وCDR)
- القدرة على البحث النصي الكامل (إذا تم تمكينه على قاعدة بيانات CDR)
- حقول CDR المفهرسة لاستعلامات التنصت القانوني السريعة

تتبع الواجهة الأمامية:

- تتبع في الوقت الحقيقي لواجهات SMSC الخارجية (SMPP، IMS، بوابات MAP)
- تسجيل الواجهة الأمامية مع مراقبة نبضات القلب
- تتبع حالة الصحة (نشط/منتهي)
- تاريخ التشغيل/التوقف
- تتبع عنوان IP واسم المضيف
- تسجيل تكوين محدد للواجهة الأمامية

1.2.3 قدرات التحليل

المراقبة في الوقت الحقيقي:

- لوحة معلومات واجهة المستخدم على الويب تظهر:
  - قائمة انتظار الرسائل النشطة
  - تقديم الرسائل وتسليمها
  - قرارات التوجيه واختيار البوابة
  - حالة بوابة الواجهة الأمامية
  - استخدام موارد النظام
- تكامل مقاييس Prometheus للمراقبة التشغيلية
- مقاييس الأداء (معدل النقل، زمن التأخير، معدلات النجاح)

 دليل المراقبة الكامل في OPERATIONS_GUIDE.md
 توثيق المقاييس في METRICS.md

التحليل التاريخي:

- قاعدة بيانات CDR قابلة للاستعلام بواسطة:
  - نطاق الوقت
  - رقم الطرف المتصل/المتصل به
  - حالة الرسالة
  - بوابة SMSC
  - محاولات التسليم
  - محتوى الرسالة (بحث نصي كامل إذا تم تمكينه)
- قدرات التحليل الإحصائي:
  - حجم الرسائل حسب الساعة/اليوم/الشهر
  - معدلات النجاح/الفشل حسب المسار
  - متوسط أوقات التسليم
  - تحليل الرسائل متعددة الأجزاء
  - أنماط التسليم الفاشلة

تتبع المشتركين:

- تاريخ الرسائل حسب رقم الهاتف (MSISDN)
- تتبع قائم على IMSI (عند توفره من واجهات IMS/MAP)
- تحليل أنماط المكالمات
- ارتباط الأطراف المتواصلة
- التحليل الزمني (تكرار الرسائل، أنماط التوقيت)

تحليلات الشبكة:

- مقاييس أداء المسار
- توفر وصحة البوابة
- تصور تدفق الرسائل
- توزيع عقد العنقود (نشر متعدد العقد)
- تحليل محاولات التسليم
- تحليل أنماط إعادة المحاولة

ذكاء الأرقام:

- تطبيع رقم E.164
- تحديد البلد/المنطقة من بادئة الرقم
- قواعد ترجمة وإعادة كتابة الأرقام
- استعلام DNS ENUM لذكاء التوجيه
- قرارات التوجيه القائمة على البادئات

 دليل ترجمة الأرقام في number_translation_guide.md
 دليل التوجيه في sms_routing_guide.md

1.3 قدرات التدابير المضادة

1.3.1 آليات حماية الخصوصية

سرية الاتصالات:

- HTTPS/TLS لتواصل REST API
- مصادقة قائمة على الشهادات
- تشفير اتصال قاعدة البيانات (دعم TLS)
- خيار حذف جسم الرسالة بعد التسليم

التحكم في الوصول:

- التحكم في الوصول إلى واجهة المستخدم على الويب
- آليات مصادقة API
- ضوابط وصول قاعدة البيانات
- مصادقة تسجيل الواجهة الأمامية

التدقيق ��تسجي:

- تسجيل كامل لأحداث النظام
- تسجيل تقديم/تسليم الرسائل
- تتبع تغييرات التكوين
- تسجيل الإجراءات الإدارية
- تسجيل منظم مع مستويات قابلة للتكوين

1.3.2 ميزات حماية البيانات

خصوصية الرسائل:

- خيار حذف جسم الرسالة بعد التسليم
- استبعاد جسم الرسالة من عرض واجهة المستخدم (اختياري)
- استبعاد جسم الرسالة من التصديرات (اختياري)
- من أجل الخصوصية NULL إلى CDR يمكن تعيين حقل جسم الرسالة

أمان قاعدة البيانات:

- دعم تشفير جداول MySQL (ENCRYPTION='Y')
- دعم تشفير البيانات الشفاف في PostgreSQL
- فصل أدوار الوصول إلى قاعدة البيانات
- حسابات مستخدمين للقراءة فقط للتحليلات
- وصول مقيد إلى محتوى الرسالة

تقوية النظام:

- الحد الأدنى من المنافذ الشبكية المكشوفة
- إدارة شهادات TLS
- تخزين تكوين آمن
- فصل التكوين بناءً على البيئة
- أمان العنقود مع بروتوكول توزيع Erlang

1.4 عمارة التخزين: تصميم ذو طبقتين Mnesia + SQL

نظرة عامة

عمارة تخزين فريدة ذات طبقتين OmniMessage يستخدم مركز خدمات الرسائل القصيرة مصممة خصيصًا لفصل معالجة الرسائل التشغيلية عالية الأداء عن التخزين طويل الأجل للامتثال والأرشفة.

الطبقة 1: قائمة انتظار الرسائل في الذاكرة Mnesia

ما هي Mnesia؟

- قاعدة بيانات موزعة مدمجة في وقت تشغيل Erlang/OTP

- تخزين هجين: أساسي في الذاكرة مع نسخ احتياطية تلقائية على القرص
- معاملات متوافقة مع ACID
- تكرار العنقود عبر عدة عقد

وضع التخزين: disc_copies

- **في الذاكرة الأساسية:** جميع الرسائل النشطة مخزنة في RAM
  - عمليات القراءة/الكتابة سريعة للغاية (أقل من مللي ثانية)
  - لا توجد عمليات إدخال/إخراج على القرص أثناء عمليات توجيه الرسائل العادية
  - يمكّن من معدل نقل 1,750+ رسالة/ثانية
- **نسخ احتياطي على القرص (تلقائي):** تقوم Mnesia بمزامنة RAM مع القرص
  - تتم الكتابات بشكل غير متزامن في الخلفية
  - يتم تحديث النسخة على القرص مع كل التزام للمعاملة
  - استعادة من التعطل: تعيد تشغيل النظام مع جميع الرسائل سليمة
  - الموقع: دليل `Mnesia.*/` في بيانات التطبيق

دورة حياة الرسالة في Mnesia:

1. تصل الرسالة عبر REST API → يتم إدراجها في Mnesia RAM + النسخ الاحتياطية على القرص
2. يستعلم محرك التوجيه Mnesia → استجابة فورية (الوصول إلى الذاكرة)
3. استعلام → تستعلم البوابة الخارجية عن الرسائل Mnesia (الوصول إلى الذاكرة)
4. تحديث → تقوم البوابة بتحديث حالة التسليم Mnesia (ذاكرة + قرص)
5. بعد التسليم/انتهاء الصلاحية → يتم وضع علامة على الرسالة للتنظيف
6. عامل التنظيف (الإعداد الافتراضي 24 ساعة) → يتم حذف الرسالة من Mnesia

الميزة الحرجة للأداء:

- **لا استعلامات قاعدة بيانات SQL** أثناء توجيه/تسليم الرسائل النشطة
- تم��مًا لمعالجة الرسائل التشغيلية SQL يتم تجاوز
- هذا يقضي على عنق الزجاجة التقليدي في SMS-C (إدخال/إخراج قاعدة البيانات)

الطبقة 2: قاعدة بيانات SQL لتصدير/أرشفة CDR

ما هو CDR (سجل تفاصيل المكالمات)؟

- سجل تدقيق دائم لبيانات التعريف ومحتوى الرسالة
- مكتوب إلى قاعدة بيانات PostgreSQL أو MySQL
- يستخدم للفوترة، التحليلات، الامتثال، والتنصت القانوني

متى يتم كتابة CDRs: يتم إنشاء سجلات CDR فقط عندما تصل الرسائل إلى حالة نهائية:

- ✅ تم تسليم الرسالة بنجاح
- ⏰ انتهت صلاحية الرسالة (تجاوزت فترة الصلاحية دون تسليم)
- ❌ فشلت الرسالة بشكل دائم (رقم غير صالح، خطأ في التوجيه)
- 🚫 تم رفض الرسالة (قواعد التوجيه، فشل التحقق)

كيف يتم كتابة CDRs:

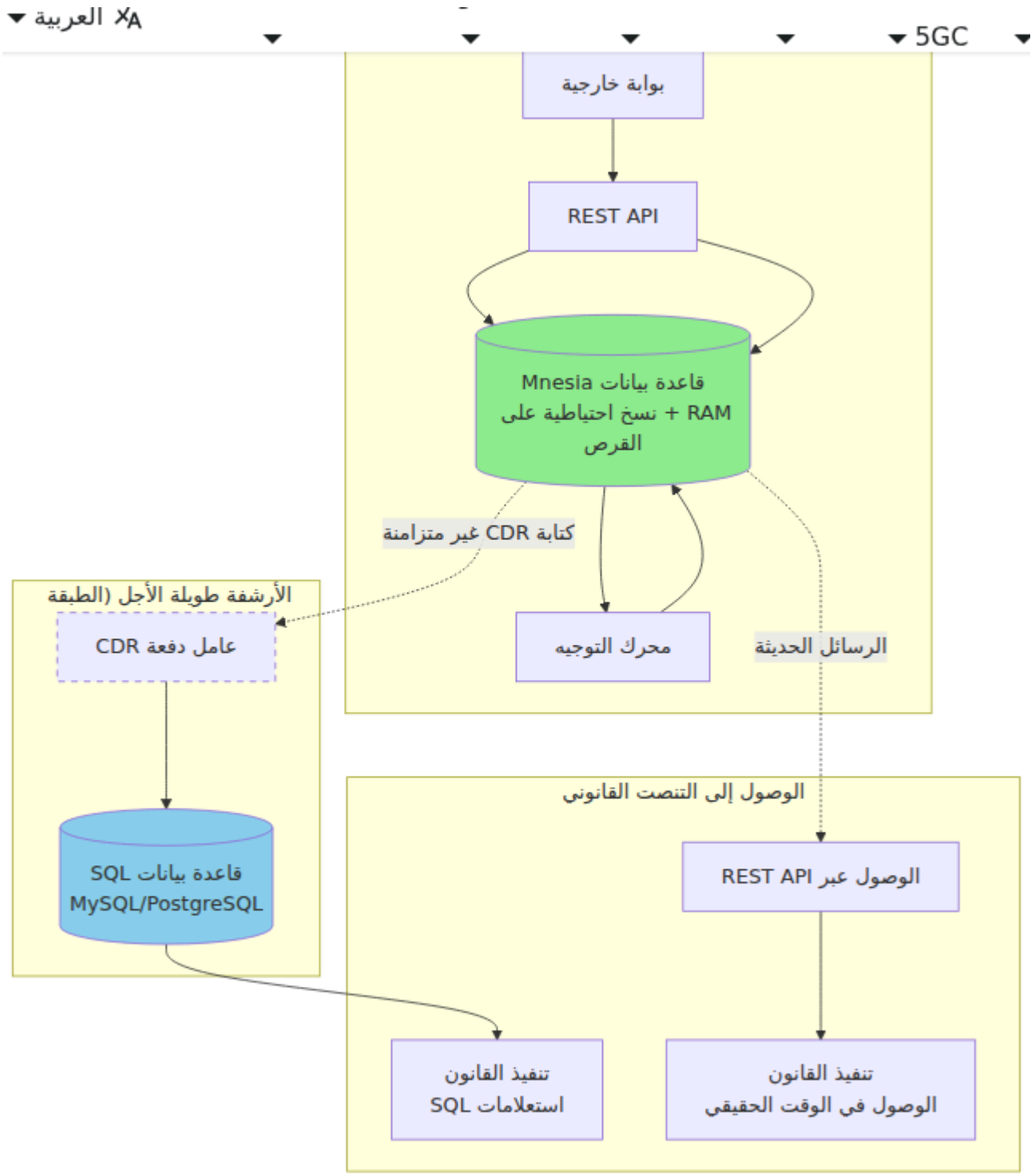
- **كتابة غير متزامنة:** يتم كتابة CDRs في عملية عامل خلفية
- **لا حظر:** لا تنتظر توجيه الرسائل لكتابة SQL
- **إدراج مجمع:** يتم تجميع عدة CDRs (الإعداد الافتراضي 100) وكتابتها معًا
- **فترة التفريغ:** 100 مللي ثانية (قابل للتكوين)
- **معالجة الأخطاء:** يتم تسجيل الكتابات الفاشلة لـ CDR، وتستمر معالجة الرسالة

```
# التكوين في config/runtime.exs
config :sms_c,
  batch_insert_batch_size: 100,          # حجم الدفعة لكتابات CDR
  batch_insert_flush_interval_ms: 100    # فترة التفريغ
```

الغرض من قاعدة بيانات SQL:

- ❌ لا تستخدم لـ: عمليات قائمة انتظار الرسائل النشطة
- ❌ لا تستخدم لـ: قرارات توجيه الرسائل
- ❌ لا تستخدم لـ: تسليم الرسائل في الوقت الحقيقي
- ✅ تستخدم فقط لـ: أرشفة CDR طويلة الأجل واستعلامات تاريخية
- ✅ تستخدم فقط لـ: استعلامات التنصت القانوني (شهور/سنوات من التاريخ)
- ✅ تستخدم فقط لـ: تقارير الفوترة والتحليلات

مخطط العمارة



الأسطورة:

- الخطوط الصلبة: عمليات متزامنة (في الوقت الحقيقي)
- الخطوط المتقطعة: عمليات غير متزامنة (في الخلفية)
- الأخضر: الطبقة عالية الأداء (في الذاكرة)
- الأزرق: الطبقة الأرشيفية (SQL المستمر)

تداعيات التنصت القانوني

الرسائل الحديثة (< 24 ساعة):

- متاحة عبر Mnesia (استعلامات REST API)
- ا��ترجاع سريع للغاية
- محتوى الرسالة الكامل متاح
- المراقبة في الوقت الحقيقي ممكنة

الرسائل التاريخية (> 24 ساعة):

- متاحة عبر قاعدة بيانات SQL (جدول CDR)
- أداء استعلام SQL القياسي
- بيانات التعريف الكاملة للرسالة متاحة دائمًا
- محتوى الرسالة متاح (ما لم يتم تمكين وضع الخصوصية)

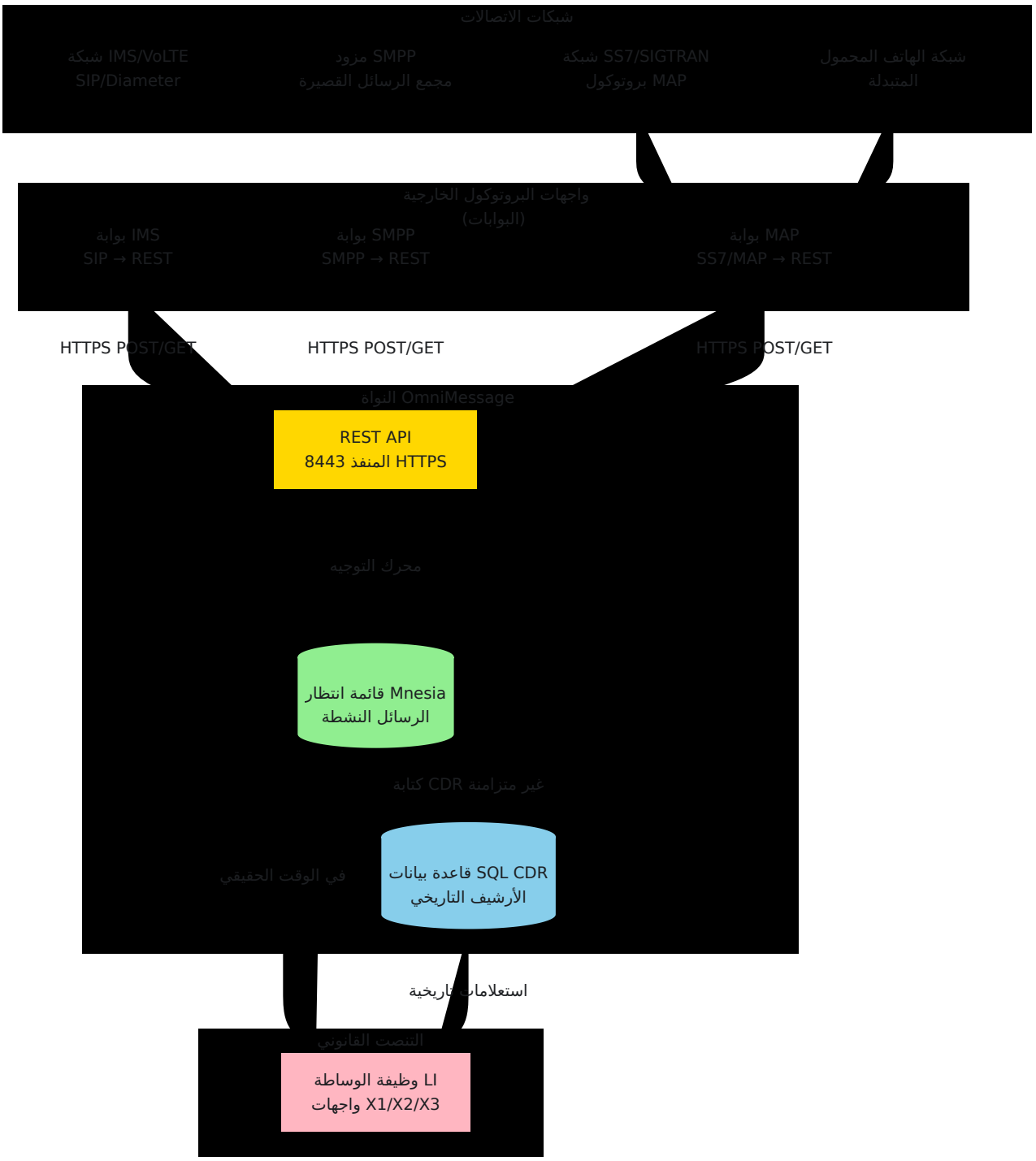
فوائد الامتثال:

1. بقاء الرسائل سليمة بعد الأعطال `disc_copies` **لا فقدان للبيانات:** يضمن وضع
2. لسنوات في قاعدة بيانات CDRs **مسار تدقيق دائم:** يتم الاحتفاظ بـ SQL
3. **الأداء:** استعلامات التنصت القانوني لا تؤثر على توجيه الرسائل
4. كلاهما متاح (SQL) الرسائل التاريخية + (Mnesia) **المرونة:** الرسائل الحديثة

1.5 عمارة تكامل الواجهة الأمامية متعددة البروتوكولات

تصميمًا أساسيًا غير مرتبط بالبروتوكول OmniMessage يستخدم مركز خدمات الرسائل القصيرة موحد. تسمح هذه العمارة REST API يتفاعل مع بوابات خارجية محددة بالبروتوكول (واجهات) عبر للتنصت القانوني بالتقاط الرسائل بغض النظر عن البروتوكول الذي تم استخدامه لإرسالها أو استلامها.

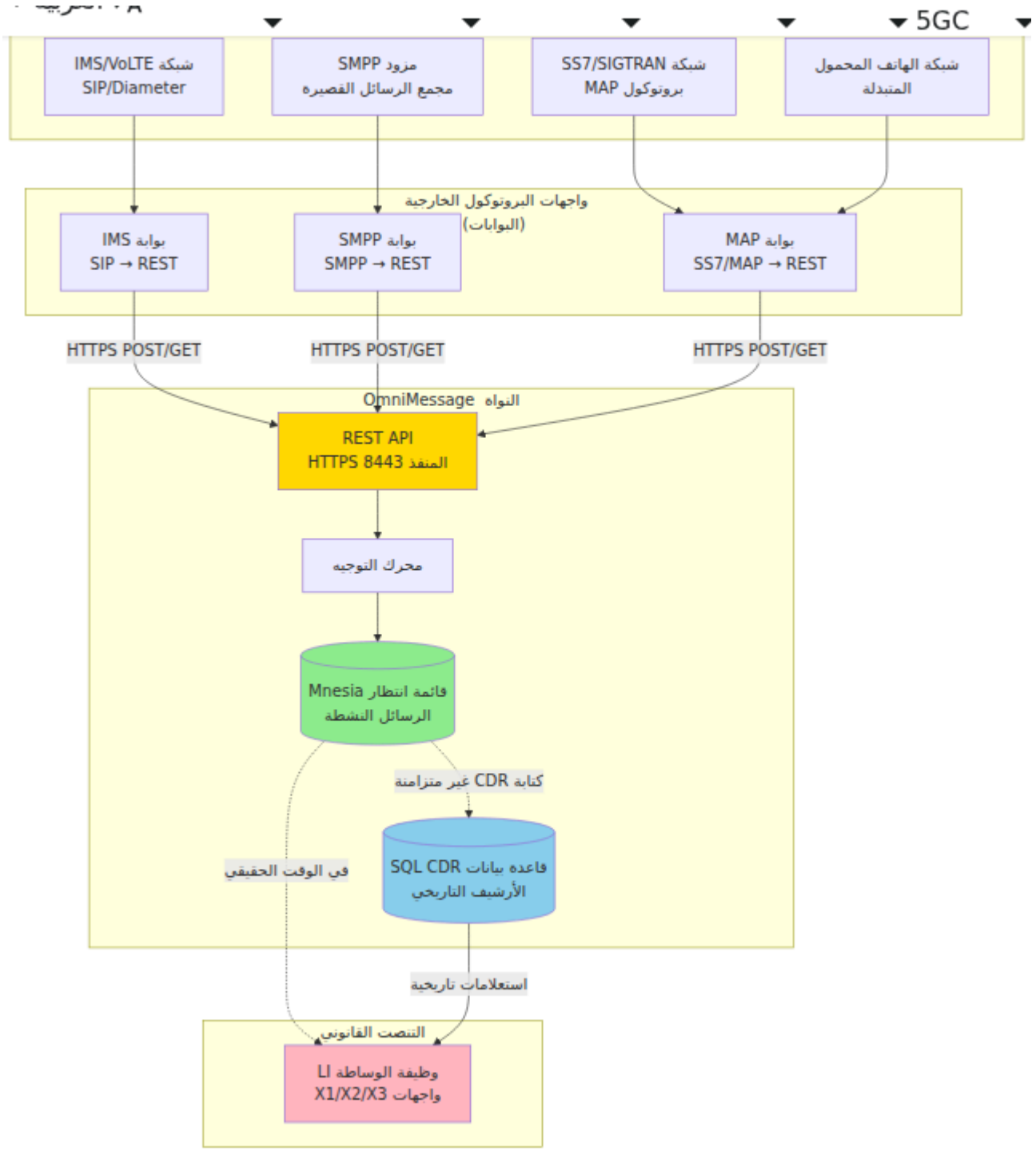
نظرة عامة على العمارة



تفاصيل تكامل بروتوكول الواجهة الأمامية

1. تكامل الواجهة الأمامية IMS/SIP

تستخدم شبكات IMS بروتوكول SIP لتبادل الرسائل القصيرة عبر IP. تقوم بوابة IMS بترجمة بين SIP وREST API لمركز خدمات الرسائل القصيرة.

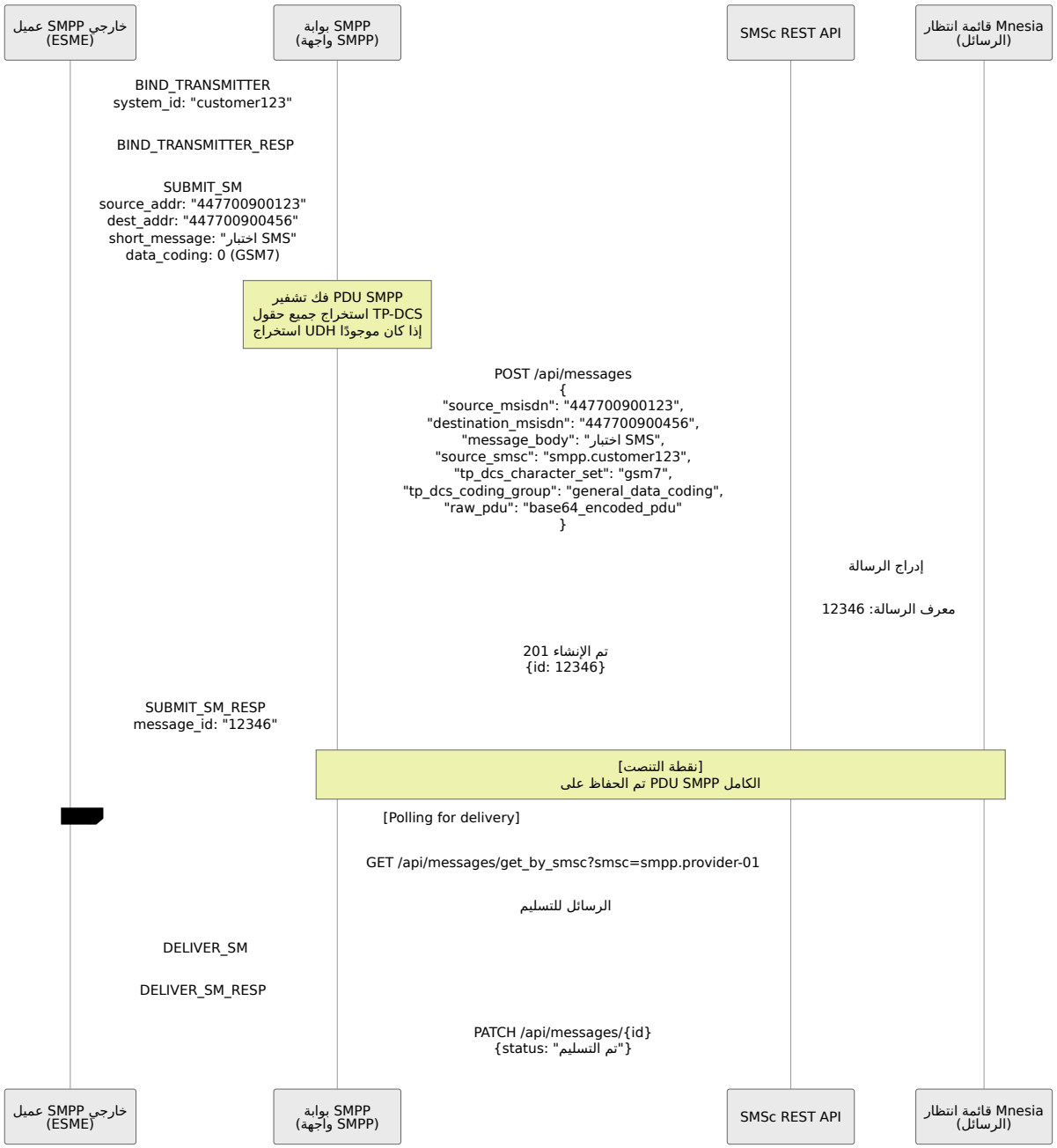


بيانات التنصت الخاصة بـ IMS:

- IMSI المصدر/الوجهة (من تسجيل IMS)
- رؤوس SIP P-Asserted-Identity
- Call-ID SIP للتتبع
- موقع شبكة IMS (P-Access-Network-Info)
- ملفات تعريف المشتركين من HSS IMS

2. تكامل الواجهة الأمامية SMPP

هو البروتوكول القياسي الصناعي لمجمعي الرسائل القصيرة ومقدمي الخدمات. تقوم بوابة SMPP بترجمة الرسائل القائمة على SMPP إلى استدعاءات PDU REST API.

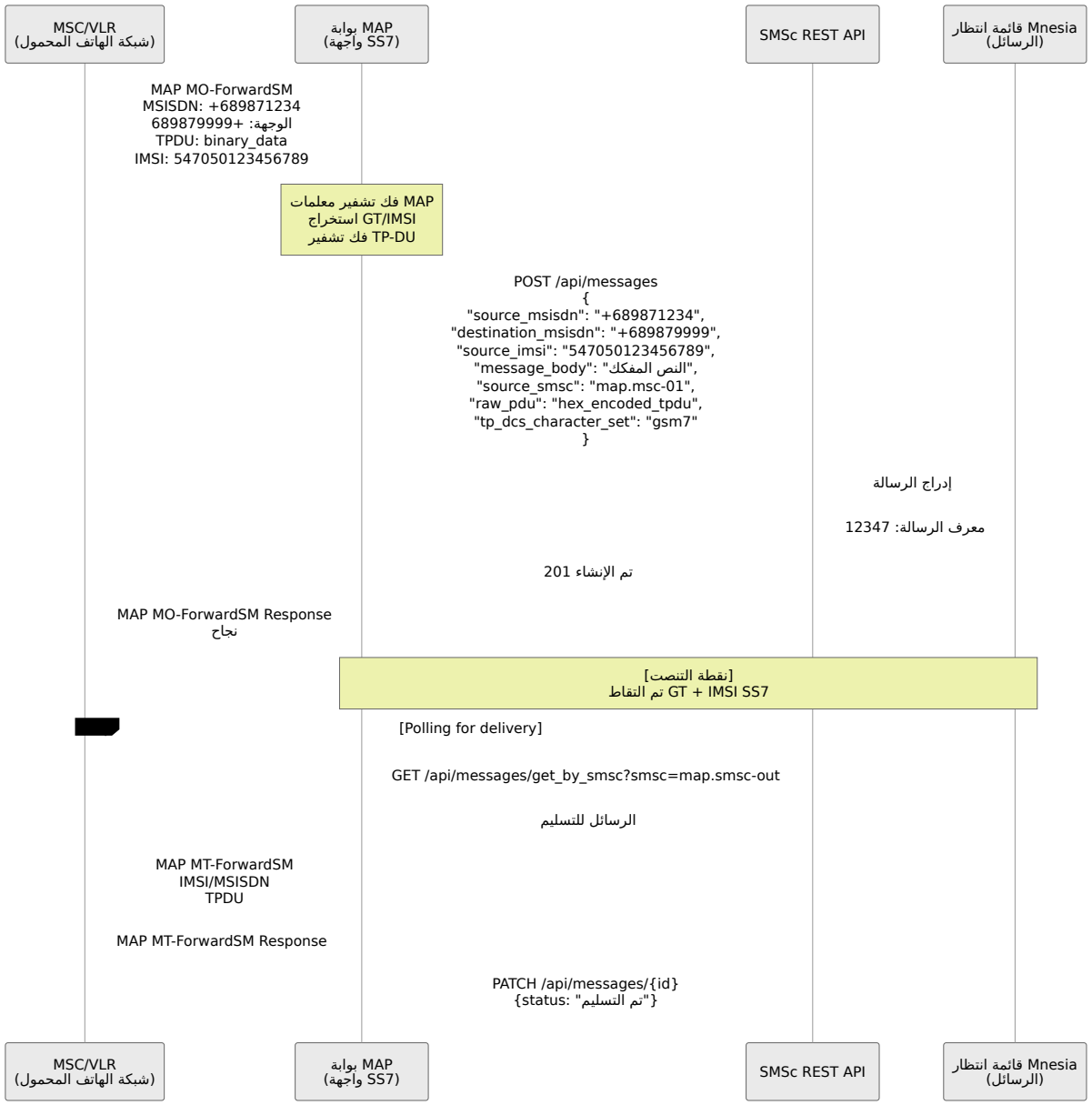


بيانات التنصت الخاصة بـ SMPP:

- PDU SMPP (تم الحفاظ على التنسيق الثنائي) الكامل
- تفاصيل نظام ترميز البيانات (DCS)
- رأس بيانات المستخدم (UDH) للرسائل المجمعة
- معرف نظام ESME (تحديد العميل)
- معلومات خطة ترقيم TON/NPI
- علامات التسليم المسجلة

3. تكامل الواجهة الأمامية SS7/MAP

تستخدم الشبكات القديمة المتبدلة بروتوكول MAP للرسائل القصيرة. تقوم ب��ابة SS7 MAP بترجمة بين الإشارات SS7 وREST API.



بيانات التنصت الخاصة بـ SS7/MAP:

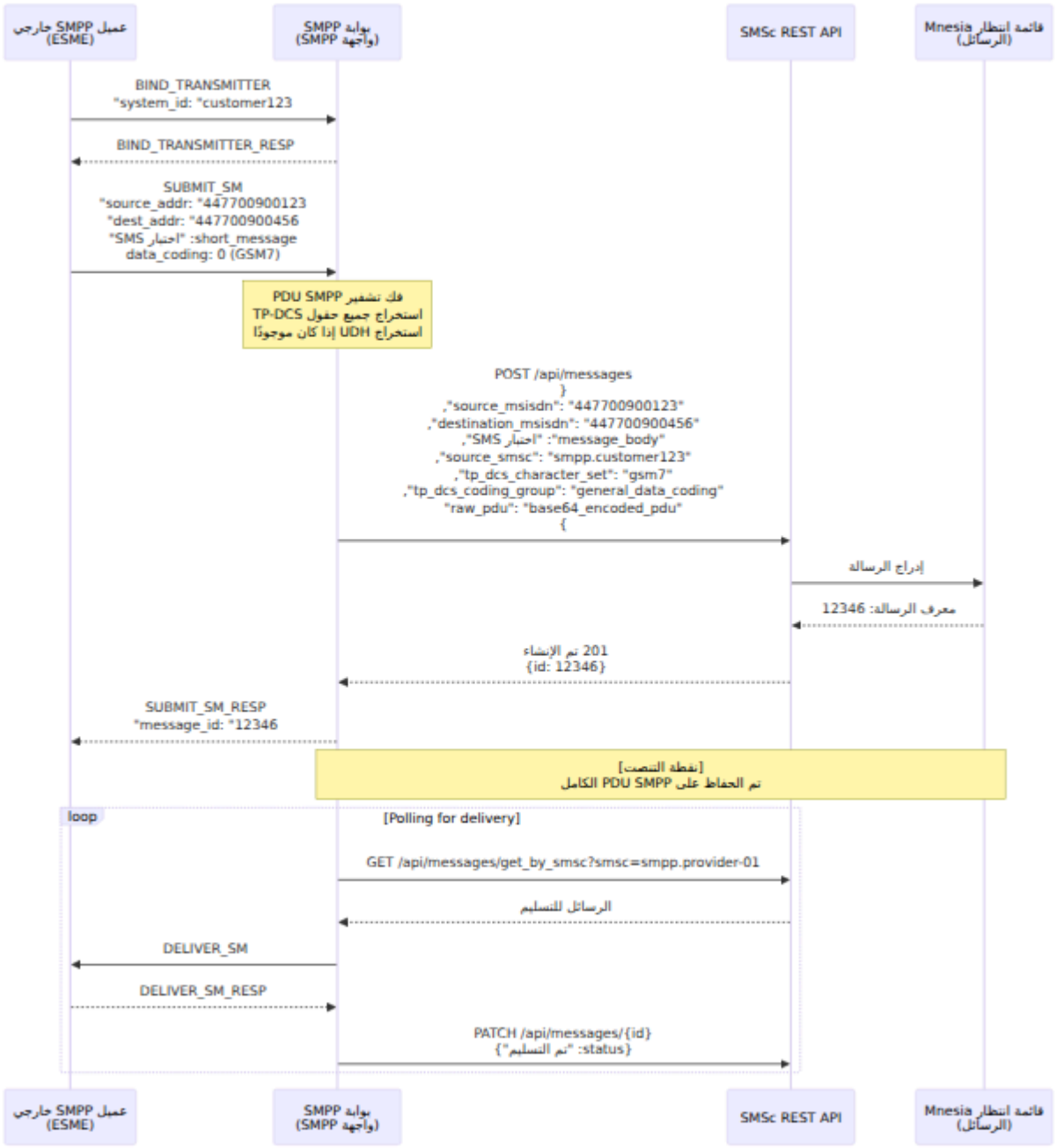
- IMSI من رسائل MAP
- عناوين Global Title (GT)
- عنوان MSC/VLR (تحديد عنصر الشبكة)
- عناوين الأطراف المتصلة SCCP
- رموز العمليات MAP
- تنسيق TP-User-Data الثنائي

التنصت الموحد عبر جميع البروتوكولات

الفائدة الرئيسية للتنصت القانوني: بغض النظر عن البروتوكول الذي تم استخدامه (IMS/SIP، SMPP، أو SS7/MAP)، تتجمع جميع الرسائل في النواة SMSc، مع هيكل بيانات موحد، مما يمكّن:

1. **المراقبة غير المرتبطة بالبروتوكول:** نقطة تنصت واحدة تلتقط جميع أنواع الرسائل
2. **تنسيق CDR الموحد:** جميع البروتوكولات تكتب إلى نفس مخطط CDR
3. **التتبع عبر البروتوكولات:** تتبع الرسائل عبر حدود البروتوكول
4. **الحفاظ على البيانات الكاملة:** يتم الحفاظ على الحقول الخا��ة بالبروتوكول في CDR

ملخص تدفق البيانات:



تحديد البروتوكول في CDR:

- يشير حقل `source_smsc` إلى بروتوكول الواجهة الأمامية (مثل "ims.gateway-01"، "smpp.customer123"، "map.msc-01")
- يمكّن من التصفية والتحليل حسب نوع البروتوكول
- يمكن لاستعلامات التنصت القانوني استهداف بروتوكولات معينة أو جميع البروتوكولات

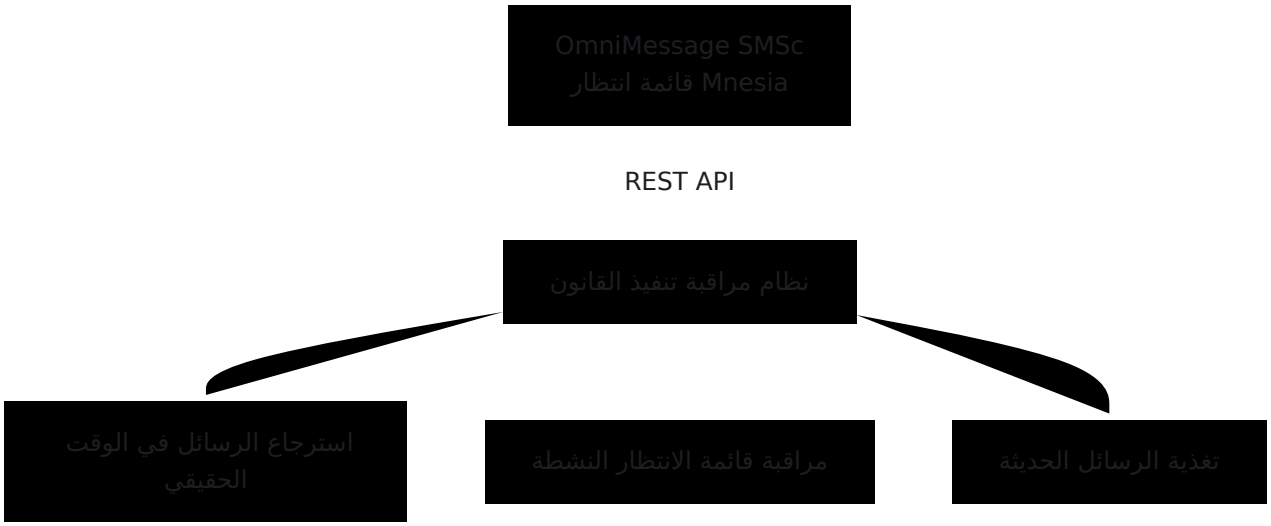
1.6 العمارة الفنية للتنصت القانوني

نقاط تكامل التنصت القانوني

توفر العمارة ذات الطبقتين نقاط وصول متعددة للتنصت القانوني، محسّنة لكل من المراقبة في الوقت الحقيقي (Mnesia) والتحليل التاريخي (SQL).

1. الوصول عبر REST API للرسائل الحديثة (Mnesia):

الوصول إلى الرسائل النشطة في قائمة انتظار Mnesia (عادةً آخر 24 ساعة):



نقاط نهاية API للتنصت في الوقت الحقيقي:

- `GET /api/messages` - قائمة الرسائل النشطة مع التصفية
- `GET /api/messages/{id}` - الحصول على تفاصيل رسالة معينة (من Mnesia)
- `GET /api/messages/get_by_smsc?smsc=X` - الحصول على الرسائل حسب البوابة
- جميع الاستعلامات تضرب Mnesia (في الذاكرة) للحصول على استجابة فورية

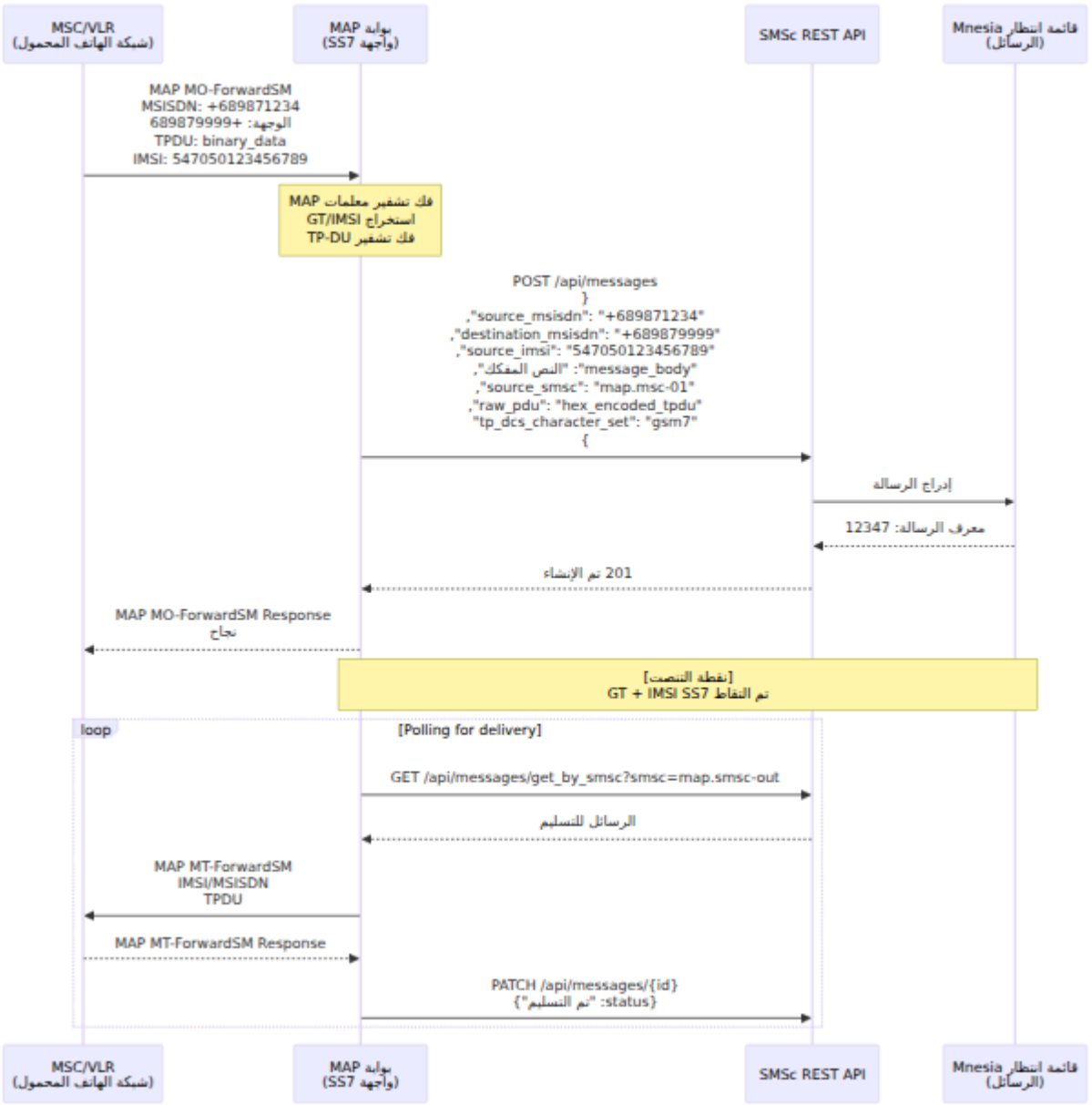
ملاحظة: تستعلم هذه النقاط النهاية قائمة انتظار الرسائل النشطة في Mnesia، مما يوفر الوصول إلى الرسائل التي تتم معالجتها حاليًا أو تم تسليمها مؤخرًا (ضمن فترة الاحتفاظ).

معلمات الاستعلام:

- تصفية بواسطة MSISDN المصدر/الوجهة
- تصفية بواسطة نطاق الوقت
- تصفية بواسطة بوابة SMSC
- تصفية بواسطة حالة الرسالة
- دعم الفرز والترتيب

2. الوصول المباشر إلى قاعدة بيانات CDR للرسائل التاريخية (SQL):

جميع الرسائل التي تم تسليمها/انتهت) SQL الوصول إلى الرسائل المؤرشفة في قاعدة بيانات صلاحيتها/فشلت):



الوصول المباشر إلى SQL:

- بيانات اعتماد قاعدة بيانات للقراءة فقط للأنظمة المصرح بها
- الوصول إلى استعلام SQL إلى جدول `cdrs` (مسار تدقيق دائم)
- عميل SQL قياسي (mysql، psql، DBeaver، إلخ) **طريقة الوصول:**
- فقط الرسائل المؤرشفة (ليس قائمة الانتظار النشطة) **مصدر البيانات:**
- حقول مفهرسة للبحث الفعال:
  - `calling_number` (مفهرس) - رقم الهاتف المصدر
  - `called_number` (مفهرس) - رقم الهاتف الوجهة
  - `message_id` (مفهرس) - معرف الرسالة الفريد

- `submission_time` (مفهرس) - عندما دخلت الرسالة النظام
- `status` (مفهرس) - الحالة النهائية للتسليم
- `dest_smsc` (مفهرس) - البوابة المستخدمة للتسليم

ملاحظة: تحتوي قاعدة بيانات CDR على سجلات دائمة لجميع الرسائل المعالجة. هذا هو المصدر الرئيسي لاستعلامات التنصت القانوني التاريخية (شهور/سنوات من البيانات).

3. تغذية الرسائل في الوقت الحقيقي (PubSub):

- تكامل Phoenix PubSub للأحداث في الوقت الحقيقي
- إشعارات تقديم الرسائل
- إشعارات تسليم الرسائل
- أحداث تغيير حالة الرسالة
- تصفية الأحداث الق��بلة للتكوين حسب المعايير
- دعم WebSocket للمراقبة الحية

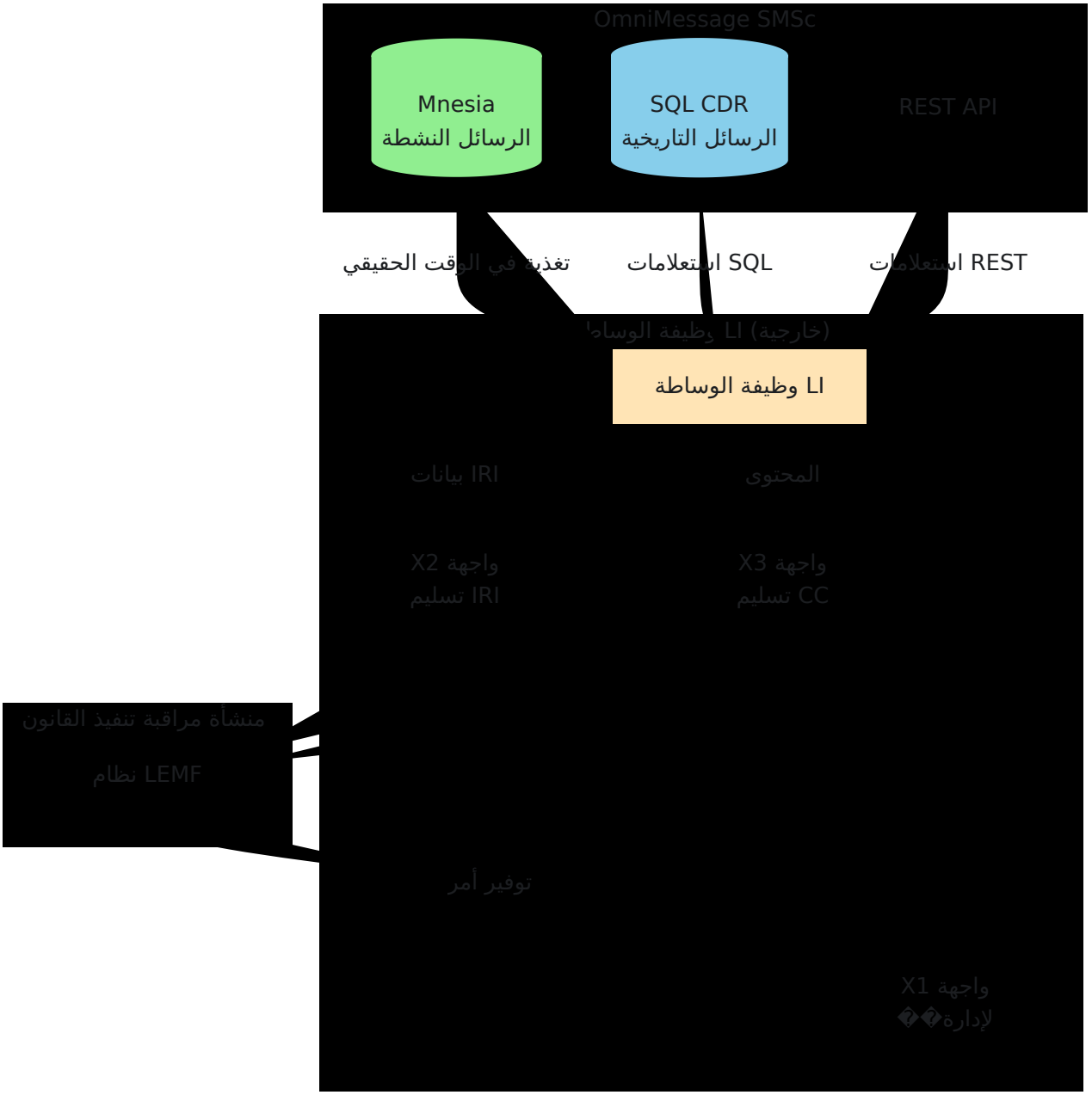
4. واجهة تصدير دفعي:

- تصدير CSV لسجلات CDR
- تصدير JSON للوصول البرمجي
- حقول تصدير قابلة للتكوين
- تصديرات قائمة على نطاق الوقت
- تصديرات واعية للخصوصية (استبعاد محتوى الرسالة الاختياري)

واجهات التنصت القانونية وفقًا لمعايير ETSI

يوفر مركز خدمات الرسائل القصيرة OmniMessage الأساس لتنفيذ واجهات التنصت القانونية المتوافقة مع ETSI. بينما لا تنفذ النواة SMSc واجهات X1/X2/X3 بشكل أصلي، فإنها توفر جميع نقاط الوصول اللازمة التي يمكن دمجها مع أنظمة وظيفة الوساطة للتنصت القانوني الخارجية (LIMF).

واجهات LI القياسية ETSI:



وصف الواجهات:

X1 - واجهة الإدارة وظيفة:

- **الغرض:** توفير أوامر التنصت وتتبع الأهداف من تنفيذ القانون إلى نظام التنصت
- **الاتجاه:** LEMF → LIMF (ثنائي الاتجاه)
- **الوظائف:**
  - تفعيل/إلغاء تفعيل التنصت لأهداف محددة (MSISDNs، IMSIs)
  - تعيين مدة التنصت وفترة الصلاحية
  - تكوين معايير التصفية (أرقام الهواتف، نوافذ الوقت)
  - استرجاع حالة التنصت
- **التكامل مع SMSc:**

- يحتفظ LIMF (قاعدة بيانات الأوامر) بقائمة الأهداف
- يستعلم LIMF من SMSc CDR/API عن الرسائل المطابقة
- يقوم LIMF بالتصفية بناءً على المعايير المقدمة من X1

واجهة X2 - تسليم IRI (المعلومات المتعلقة بالتنصت):

- **الغرض:** تسليم بيانات التعريف المتعلقة بالرسالة إلى تنفيذ القانون
- **الاتجاه:** LIMF → LEMF (اتجاه واحد)
- **تنسيق البيانات:** متوافق مع ETSI TS 102 232-x XML/ASN.1
- **المحتوى من SMSc CDR:**
  - معرف الرسالة
  - رقم الاتصال (المصدر MSISDN)
  - رقم المتصل (الوجهة MSISDN)
  - IMSI (المصدر والوجهة، إذا كانت متاحة)
  - الطابع الزمني للإرسال
  - الطابع الزمني للتسليم
  - حالة الرسالة (تم التسليم/فشل/انتهت صلاحيتها)
  - محاولات التسليم
  - معلومات بوابة SMSC (المصدر/الوجهة)
  - موقع الشبكة (إذا كان متاحًا)
- **التكامل مع SMSc:**
  - يستعلم LIMF عن أرقام الهواتف المستهدفة من قاعدة بيانات CDR
  - يقوم LIMF بتحويل سجلات CDR إلى تنسيق IRI متوافق مع ETSI
  - يسلم LIMF IRI إلى LEMF عبر X2

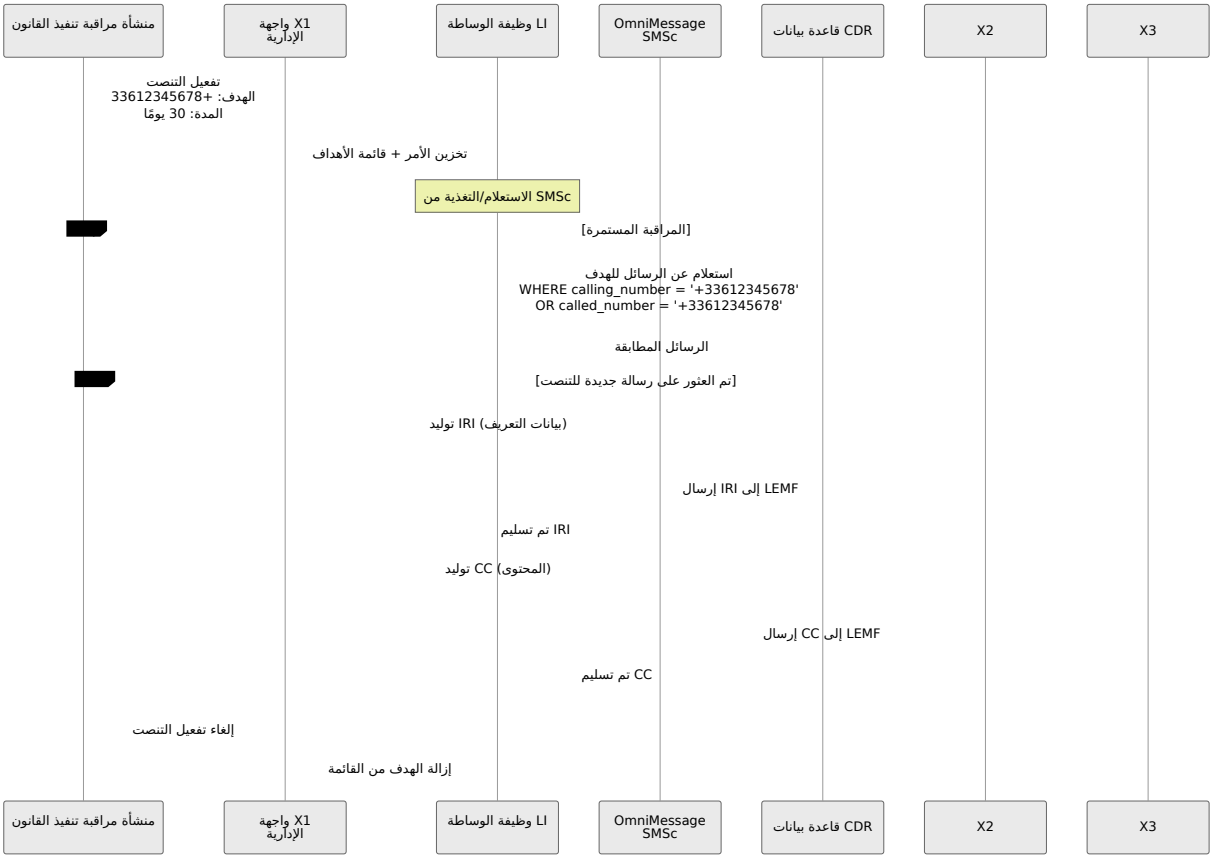
واجهة X3 - تسليم CC (محتوى الاتصال):

- **الغرض:** تسليم محتوى الرسالة الفعلي إلى تنفيذ القانون
- **الاتجاه:** LIMF → LEMF (اتجاه واحد)
- **تنسيق البيانات:** متوافق مع ETSI TS 102 232-x
- **المحتوى من SMSc:**
  - جسم الرسالة (محتوى النص)
  - PDU الخام (بيانات SMS الثنائية)
  - معلومات الترميز
  - أجزاء الرسائل متعددة الأجزاء

- معلومات TP-DCS

- **التكامل مع SMSc:**
  - يسترجع LIMF محتوى الرسالة من حقل `message_body` في CDR
  - يسترجع LIMF بيانات PDU الخام إذا كانت متاحة
  - يقوم LIMF بتعبئة المحتوى في تنسيق CC متوافق مع ETSI
  - يسلم LIMF CC إلى LEMF عبر X3

عمارة التنفيذ:



تعيين بيانات SMSc إلى واجهات LI:

حقل بيانات SMSc	X2 (IRI)	X3 (CC)	عمود جدول CDR
معرف الرسالة	✅ معرف الارتباط	✅ مرجع	`message_id`
رقم الاتصال	✅ الطرف A	-	`calling_number`
رقم المتصل	✅ الطرف B	-	`called_number`
وقت الإرسال	✅ الطابع الزمني	-	`submission_time`
وقت التسليم	✅ الاكتمال	-	`delivery_time`
الحالة	✅ النتيجة	-	`status`
جسم الرسالة	-	✅ المحتوى	`message_body`
PDU الخام	-	✅ ثنائي	(Mnesia/CDR)
المصدر SMSC	✅ عنصر الشبكة	-	`source_smsc`
الوجهة SMSC	✅ عنصر الشبكة	-	`dest_smsc`
IMSI	✅ معرف المشترك	-	(عبر الواجهات)

خيارات تكامل LIMF:

الخيار 1: عمارة الاستعلام

- يستعلم LIMF دوريًا من قاعدة بيانات CDR (كل 1-60 ثانية)
- X1 يصفّي حسب أرقام الهواتف المستهدفة من قائمة الأوامر SQL استعلام
- تعقيد منخفض، سهل التنفيذ
- LI تأخير طفيف بين تسليم الرسالة وتسليم

الخيار 2: عمارة التغذية في الوقت الحقيقي

- ينشر SMSc أحداث الرسائل
- في تدفق الرسائل في الوقت الحقيقي LIMF يشترك

- يقوم LIMF بالتصفية بناءً على قائمة الأهداف
- زمن تأخير قريب من الصفر للتنصت القانوني
- يتطلب تطوير تكامل مخصص

الخيار 3: ��لعمارة الهجينة

- في الوقت الحقيقي (< 24 ساعة) PubSub الرسائل الحديثة: تدفق
- CDR الرسائل التاريخية: استعلام قاعدة بيانات
- توازن مثالي بين زمن التأخير والموثوقية

آليات تحفيز التنصت

التنصت القائم على الهدف:

- مطابقة رقم الهاتف (MSISDN)
- استهداف قائم على IMSI (عند توفره)
- قوائم مراقبة قابلة للتكوين
- وجهات نظر قاعدة البيانات لعزل الأهداف
- تصفية API بواسطة معرفات الأهداف

التنصت القائم على الحدث:

- جميع الرسائل إلى/من أرقام محددة
- الرسائل عبر بوابات SMSC محددة
- الرسائل ذات الخصائص المحددة (متعددة الأجزاء، تسليم فاشل، إلخ)
- التوجيه الجغرافي (عبر ENUM أو مطابقة البادئات)

التنصت القائم على الوقت:

- تصفية نطاق التاريخ/الوقت في استعلامات CDR
- فرض فترة الاحتفاظ
- الأرشفة التلقائية للرسائل القديمة
- سياسات احتفاظ بالبيانات قابلة للتكوين

أمثلة استعلام SQL للتنصت القانوني:

```sql
-- الحصول على جميع الرسائل للرقم المستهدف
SELECT * FROM cdrs
WHERE calling_number = '+33612345678'
   OR called_number = '+33612345678'
ORDER BY submission_time DESC;

-- الحصول على الرسائل في نافذة زمنية محددة
SELECT * FROM cdrs
WHERE (calling_number = '+33612345678' OR called_number =
'+33612345678')
  AND submission_time BETWEEN '2025-11-01 00:00:00' AND '2025-11-
30 23:59:59'
ORDER BY submission_time;

-- الحصول على المحادثة بين طرفين
SELECT * FROM cdrs
WHERE (calling_number = '+33612345678' AND called_number =
'+33687654321')
   OR (calling_number = '+33687654321' AND called_number =
'+33612345678')
ORDER BY submission_time;
```

---

2. قدرات التشفير والتحليل

2.1 نظرة عامة على القدرات التشفيرية

يطبق مركز خدمات الرسائل القصيرة OmniMessage آليات تشفير لتأمين الاتصالات وحماية البيانات الحساسة. توثق هذه القسم جميع القدرات التشفيرية وفقًا لمتطلبات ANSSI.

2.2 تشفير طبقة النقل

2.2.1 تنفيذ TLS/SSL

البروتوكولات المدعومة:

- TLS 1.2 (RFC 5246)
- TLS 1.3 (RFC 8446) - موصى به

- SSL 2.0/3.0: غير مدعوم (ثغرات معروفة)
- TLS 1.0/1.1: ملغى (غير موصى به)

التنفيذ:

- مكتبة SSL/TLS Erlang/OTP (تم التحقق منها تشفيرياً)
- خادم الويب Cowboy مع دعم TLS
- نقاط نهاية HTTPS لإطار عمل Phoenix

أطقم التشفير:

يستخدم النظام اختيار مجموعة التشفير الآمن الافتراضي لـ Erlang/OTP، والتي تشمل:

المفضل - TLS 1.3:

- TLS_AES_256_GCM_SHA384
- TLS_AES_128_GCM_SHA256
- TLS_CHACHA20_POLY1305_SHA256

المدعوم - TLS 1.2:

- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES128-GCM-SHA256

ميزات الأمان:

- سرية التقدم المثالي (PFS) عبر تبادل المفاتيح ECDHE/DHE
- مجموعات Diffie-Hellman قوية (2048 بت كحد أدنى)
- دعم تشفير المنحنى البيضاوي
- دعم إشارة اسم الخادم (SNI)

إدارة الشهادات:

- دعم الشهادات X.509
- أحجام مفاتيح RSA: 2048 بت كحد أدنى، 4096 بت موصى بها
- دعم ECDSA

- التحقق من سلسلة الشهادات
- شهادات موقعة ذاتيًا (للتطوير فقط)
- تكام�� CA الخارجي

موقع تكوين TLS:

```
# config/runtime.exs
config :api_ex,
  api: %{
    enable_tls: true,
    tls_cert_path: "priv/cert/omnitouch.crt",
    tls_key_path: "priv/cert/omnitouch.pem"
  }
```

** مرجع التكوين الكامل في CONFIGURATION.md**

التطبيقات:

- HTTPS لـ REST API (المنفذ 8443)
- HTTPS (المنفذ 8086) للوحة التحكم على الويب
- اتصالات قاعدة البيانات (MySQL/PostgreSQL عبر TLS)

2.3 تشفير البيانات في حالة السكون

2.3.1 تشفير قاعدة البيانات

تشفير MySQL/MariaDB:

- دعم التشفير على مستوى الجدول
- خوارزمية التشفير AES-256
- تشفير البيانات الشفاف (TDE)

```
-- تمكين التشفير لجدول CDR
ALTER TABLE cdrs ENCRYPTION='Y';
```

تشفير PostgreSQL:

- دعم التشفير الشفاف للبيانات
- تشفير على مستوى نظام الملفات
- تشفير على مستوى العمود (امتداد pgcrypto)

2.3.2 تخزين القرص Mnesia

قاعدة بيانات Mnesia:

- تخزين نسخ القرص لاستمرارية الرسائل
- يوصى بتشفير مستوى نظام الملفات (LUKS، dm-crypt)
- حماية الذاكرة عبر عزل VM Erlang

2.3.3 تشفير نظام الملفات

تخزين البيانات الحساسة:

- ملفات التكوين: يوصى بتشفير نظام الملفات
- المفاتيح الخاصة: أذونات الملفات (0600) + تشفير نظام الملفات
- ملفات السجل: تشفير قابل للتكوين لسجلات الأرشفة
- تصديرات CDR: تخزين مشفر للتصديرات الحساسة

تخزين المفاتيح:

- يتم تخزين شهادات TLS والمفاتيح في `priv/cert/`
- مخازن المفاتيح المعتمدة على الملفات مع أذونات مقيدة
- إجراءات دوران المفاتيح الآمنة

2.4 المصادقة والتحكم في الوصول

2.4.1 مصادقة API

أمان REST API:

- تشفير النقل عبر HTTPS/TLS إلزامي
- مصادقة قائمة على الرؤوس (رأس SMSc الأمامية لتحديد الواجهة)
- التحكم في الوصول بناءً على IP (على مستوى جدار الحماية)
- مصادقة العميل المعتمدة على الشهادات (اختياري)

تسجيل الواجهة الأمامية:

- تحديد فريد للواجهة الأمامية (اسم المضيف، IP، الاسم، النوع)
- مصادقة قائمة على نبضات القلب
- إدارة جلسات قائمة على انتهاء الصلاحية (مهلة 90 ثانية)
- تتبع ومراقبة الواجهة الأمامية

2.4.2 مصادقة قاعدة البيانات

التحكم في الوصول إلى قاعدة البيانات:

- مصادقة اسم المستخدم/كلمة المرور
- دعم اتصال TLS/SSL
- قيود اتصال بناءً على IP
- التحكم في الوصول القائم على الدور (RBAC)

التكوين:

```
# config/runtime.exs
config :sms_c, SmsC.Repo,
  username: "omnitouch",
  password: "omnitouch2024",  # يجب استخدام كلمات مرور قوية في الإنتاج
  hostname: "localhost",
  ssl: true  # تمكين TLS لاتصالات قاعدة البيانات
```

توصيات التحكم في الوصول:

```
-- إنشاء مستخدم للقراءة فقط للوصول إلى تنفيذ القانون
CREATE USER 'li_readonly'@'%' IDENTIFIED BY 'secure_password';
GRANT SELECT ON sms_c.cdrs TO 'li_readonly'@'%';

-- إنشاء مستخدم محدود بدون وصول إلى محتوى الرسالة
CREATE USER 'analytics'@'%' IDENTIFIED BY 'secure_password';
GRANT SELECT (id, message_id, calling_number, called_number,
              source_smsc, dest_smsc, submission_time,
delivery_time,
              status, delivery_attempts)
ON sms_c.cdrs TO 'analytics'@'%';
```

2.5 تفاصيل الخوارزميات التشفيرية

2.5.1 خوارزميات التجزئة

المتاحة في Erlang/OTP:

- SHA-256، SHA-384، SHA-512 (موصى بها)
- SHA-1 (ملغاة، فقط للتوافق مع الإصدارات القديمة)
- MD5 (ملغاة، لا تستخدم للأمان)
- BLAKE2 (الحديثة OTP متاحة في إصدارات)

الاستخدام:

- بصمات الرسائل (الكشف عن التكرار)
- التحقق من سلامة البيانات
- سلامة سجل التدقيق

2.5.2 التشفير المتماثل

الخوارزميات المتاحة:

- AES (معيار التشفير المتقدم)
  - AES-128-GCM
  - AES-256-GCM
  - AES-128-CBC
  - AES-256-CBC

- ChaCha20-Poly1305

أحجام المفاتيح:

- 128 بت (الحد الأدنى)
- 256 بت (موصى بها)

الاستخدام:

- تشفير جلسة TLS
- تشفير قاعدة البيانات في حالة السكون
- تشفير جسم الرسالة الاختياري

2.5.3 التشفير غير المتماثل

الخوارزميات المدعومة:

- RSA (2048 بت موصى بها، 4096 بت كحد أدنى)
- ECDSA (خوارزمية توقيع رقمي باس��خدام المنحنيات البيضاوية)
  - P-256، P-384، P-521
- Ed25519 (EdDSA)

الاستخدام:

- مصادقة شهادة TLS
- التوقيعات الرقمية
- تبادل المفاتيح

2.6 أمان بروتوكول SMS

2.6.1 ترميز رسالة SMS

دعم ترميز الأحرف:

- GSM 7-bit (ترميز SMS القياسي)
- UCS-2 (Unicode، 16-bit)
- 8 بيانات ثنائية-bit
- Latin-1

TP-DCS (نظام ترميز البيانات):

- إشارة فئة الرسالة
- علامات الضغط
- تحديد مجموعة الترميز
- تحديد مجموعة الأحرف

أصلي SMS لا تشفير:

- لا يوفر بروتوكول SMS تشفيرًا من طرف إلى طرف
- محتوى الرسالة متاح على مستوى SMSc
- يمكّن من التنصت القانوني كما هو مطلوب

2.6.2 اعتبارات أمان البروتوكول

بروتوكول SMPP (واجهة خارجية):

- مصادقة اسم المستخدم/كلمة المرور على مستوى SMPP
- دعم TLS متاح (SMPP عبر TLS)
- مصادقة الربط

بروتوكول IMS (واجهة خارجية):

- رسائل قائمة على SIP
- آليات مصادقة SIP
- التكامل مع أمان شبكة IMS الأساسية

بروتوكول SS7/MAP (واجهة خارجية):

- أمان شبكة SS7
- مصادقة بروتوكول MAP
- أمان طبقة SCCP/TCAP

ملاحظة: يتم تنفيذ الأمان الخاص بالبروتوكول في بوابات الواجهة الخارجية، وليس في النواة SMSc.

2.7 قدرات التحليل ومقاومة الثغرات الأمنية

2.7.1 أدوات تحليل البروتوكول

قدرات تصحيح الأخطاء المدمجة:

- نظام تسجيل شامل
- تتبع تدفق الرسائل
- تسجيل طلبات/استجابات API
- تسجيل استعلامات قاعدة البيانات
- تتبع الأخطاء والاستثناءات

التكامل الخارجي:

- إخراج تسجيل قياسي (stdout/ملفات)
- دعم التقاط PCAP لتحليل الشبكة
- تسجيل استعلامات قاعدة البيانات للأدلة
- تصدير مقاييس Prometheus

2.7.2 اعتبارات تقييم الثغرات الأمنية

القيود المعروفة:

- غير مشفر بطبيعته (حسب التصميم، يمكّن من التنصت القانوني) SMS بروتوكول
- بيانات اعتماد قاعدة البيانات في ملفات التكوين (يجب استخدام إدارة الأسرار)
- دعم الشهادات الموقعة ذاتيًا (للتطوير/الاختبار فقط)

توصيات تقوية الأمان:

- استخدام مجموعات تشفير TLS قوية
- تنفيذ تشفير اتصال قاعدة البيانات
- استخدام إدارة الأسرار الخارجية (Vault، AWS Secrets Manager)
- تحديثات أمان منتظمة لـ Erlang/OTP والاعتمادات
- قيود جدار الحماية على منافذ API
- قائمة بيضاء IP للوصول إلى الواجهة الأمامية

اختبار الأمان:

- مسح الثغرات الأمنية للاعتمادات بانتظام
- دعم اختبار الاختراق
- التحقق من تكوين TLS
- تدقيق أمان قاعدة البيانات
- مراجعة التحكم في الوصول

2.8 بنية إدارة المفاتيح

2.8.1 توليد المفاتيح

توليد شهادة TLS:

```
# توليد مفتاح خاص (RSA 4096-bit)
openssl genrsa -out omnitouch.pem 4096

# توليد طلب توقيع الشهادة
openssl req -new -key omnitouch.pem -out omnitouch.csr

# شهادة موقعة ذاتيًا (للتطوير)
openssl x509 -req -days 365 -in omnitouch.csr -signkey
omnitouch.pem -out omnitouch.crt

# الإنتاج: الحصول على شهادة من CA موثوق
```

توليد الأرقام العشوائية:

- CSPRNG (مولد الأر��ام العشوائية الآمن تشفيرياً) في Erlang/OTP
- مجموعة الطاقة النظامية (/dev/urandom)
- عشوائية قوية لمفاتيح الجلسة، المعرفات، الرموز

2.8.2 تخزين المفاتيح وحمايتها

تخزين المفتاح الخاص:

- نظام الملفات مع أذونات مقيدة (0600)
- مخزنة في دليل `priv/cert/`
- تنسيق PEM (مشفر اختياريًا)
- إجراءات النسخ الاحتياطي الآمنة

دوران المفاتيح:

- إجراءات تجديد شهادة TLS (موصى بها سنويًا)
- دوران بيانات اعتماد قاعدة البيانات
- دوران رموز API (إذا تم تنفيذها)

2.8.3 توزيع المفاتيح

توزيع الشهادات:

- التثبيت اليدوي في `priv/cert/`
- مراجع ملفات التكوين
- دعم بروتوكول ACME ممكن (Let's Encrypt)

توزيع المفاتيح المتماثلة:

- تبادل المفاتيح خارج النطاق لبيانات اعتماد قاعدة البيانات
- اتفاقية مفاتيح Diffie-Hellman في TLS
- عدم نقل المفاتيح في نص واضح

2.9 الامتثال والمعايير

توثق هذه القسم الامتثال للمعايير الدولية للاتصالات، والأطر التنظيمية، والمواصفات الأمنية المعمول بها لمعالجة SMS عبر جميع البروتوكولات المدعومة.

2.9.1 الامتثال لـ SMS عبر بروتوكول SS7/MAP

معايير 3GPP وETSI:

- **3GPP TS 23.040:** مواصفة - التنفيذ الفني لخدمة الرسائل القصيرة (SMS) بروتوكول SMS الأساسية
- **3GPP TS 23.038:** ترميز الأحرف - الأبجديات والمعلومات الخاصة باللغة (GSM7، UCS-2)
- **3GPP TS 29.002:** مواصفة جزء التطبيق المحمول (MAP) - الإشارات لـ SS7 SMS
- **3GPP TS 23.003:** تنسيقات - العناوين والتعريف، الترقيم MSISDN، IMSI
- **ETSI TS 100 901:** مواصفة خدمة الرسائل القصيرة من نقطة إلى نقطة
- **ETSI TS 100 902:** مواصفة خدمة الرسائل القصيرة للبث الخلوي

معايير إشارات SS7:

- **ITU-T Q.711-Q.716:** جزء التحكم في اتصال الإشارات (SCCP)
- **ITU-T Q.771-Q.775:** جزء قدرات المعاملات (TCAP)
- **ITU-T Q.701-Q.710:** جزء نقل الرسائل (MTP) 3-1 المستويات
- **ETSI EN 300 356:** نظام الإشارات رقم 7 - جزء المستخدم ISDN (ISUP)

معايير الأمان لـ SS7/MAP:

- **GSMA FS.07:** ضوابط الأمان الأساسية - Diameterو SS7 أمان إشارات
- **GSMA FS.11:** إرشادات مراقبة أمان SS7
- **3GPP TS 33.117:** كتالوج متطلبات ضمان الأمان العامة
- **ETSI TS 133 210:** أمان نطاق الشبكة - أمان طبقة الشبكة IP

التنصت القانوني لـ SS7/MAP:

- **ETSI TS 101 671:** واجهة التسليم للتنصت القانوني على(LI) التنصت القانوني ؛ حركة المرور في الاتصالات
- **ETSI TS 102 232-1:** مواصفة التسليم - الجزء 1: واجهة(LI) التنصت القانوني ؛ التسليم لإدارة LI
- **3GPP TS 33.107:** بنية ووظائف التنصت القانوني لشبكات 3G

2.9.2 الامتثال لـ SMS عبر بروتوكول IMS

معايير IMS 3GPP:

- **3GPP TS 23.228:** نظام الوسائط المتعددة IP (IMS) - 2 بنية المرحلة
- **3GPP TS 24.229:** بروتوكول التحكم في المكالمات متعددة الوسائط IP - إجراءات SIPو SDP
- **3GPP TS 24.341:** دعم SMS عبر الشبكات IP - 3 مواصفة المرحلة
- **3GPP TS 23.204:** دعم خدمة الرسائل القصيرة (SMS) عبر الوصول IP العام 3GPP - 2 المرحلة
- **3GPP TS 29.228:** واجهات Cxو Dx لنظام الوسائط المتعددة (IM)

معايير أمان IMS:

- **3GPP TS 33.203:** أمان الوصول للخدمات القائمة علىG أمان 3 ؛ IP (IMS AKA)

- **3GPP TS 33.210:** أمان طبقة الشبكة(NDS) ؛ أمان نطاق الشبكةGأمان 3 IP (IPsec)
- **3GPP TS 33.310:** أمان نطاق الشبكة (NDS) ؛ إطار المصادقة (AF)
- **ETSI TS 133 203:** أمان الوصول للخدمات القائمة على IP

معايير بروتوكول SIP:

- **RFC 3261:** SIP: بروتوكول بدء الجلسات - المواصفة الأساسية
- **RFC 3428:** تمديد SIP للرسائل الفورية - طريقة MESSAGE
- **RFC 3325:** الامتدادات الخاصة بـ SIP للهوية المؤكدة
- **RFC 5765:** قضايا الأمان والحلول في أنظمة الند للند

التنصت القانوني لـ IMS:

- **ETSI TS 102 232-5:** التنصت القانوني (LI)5 - الجزء LI :؛ مواصفة التسليم المستقل عن الخدمة لـ خدمات نظام الوسائط المتعددة
- **3GPP TS 33.107:** متطلبات بنية التنصت القانوني

2.9.3 الامتثال لبروتوكول SMPP

مواصفة SMPP:

- **SMPP v3.4:** مواصفة بروتوكول الرسائل القصيرة من نظير إلى نظير - معيار صناعي
- **SMPP v5.0:** الموسع مع ميزات محسّنة SMPP بروتوكول

إرشادات أمان SMPP:

- **TLS عبر SMPP:** أمان طبقة النقل لاتصالات SMPP (SMPP عبر TLS)
- **مصادقة الربط SMPP:** مصادقة معرف النظام وكلمة المرور
- **التحكم في الوصول بناءً على IP:** قيود على مستوى الشبكة لاتصالات SMPP

معايير التوافق:

- **GSM 03.40 (ETSI TS 100 901):** التنفيذ الفني لـ SMS من نقطة إلى نقطة (PP)
- **GSM 03.38 (ETSI TS 100 900):** الأبجديات والمعلومات الخاصة باللغة
- **GSM 04.11 (ETSI TS 100 942):** دعم SMS من نقطة إلى نقطة على واجهة الراديو المحمول

امتثال ترميز الرسائل:

- **ITU-T T.50:** الأبجدية الدولية رقم 5 (IA5)
- **ISO/IEC 8859-1:** ترميز الأحرف Latin-1
- **ISO/IEC 10646:** مجموعة الأحرف العالمية (UCS-2/UTF-16)

2.9.4 الامتثال لمعايير التشفير

TLS وأمان الشبكة:

- **NIST SP 800-52:** إرشادات لاختيار وتكوين واستخدام تنفيذات TLS
- **NIST SP 800-131A:** الانتقال إلى استخدام خوارزميات التشفير وأطوال المفاتيح
- **RFC 7525:** توصيات للاستخدام الآمن لـ TLS وDTLS
- **RFC 8446:** الإصدار 1.3 (TLS) بروتوكول أمان طبقة النقل

معايير خوارزميات التشفير:

- **FIPS 197:** معيار التشفير المتقدم (AES)
- **FIPS 180-4:** معيار التجزئة الآمن (عائلة SHA-2)
- **NIST SP 800-38D:** توصية لأساليب تشغيل خوارزميات التشفير: وضع GCM
- **RFC 7539:** لبروتوكولات IETF Poly1305 وChaCha20

إدارة المفاتيح:

- **NIST SP 800-57:** توصية لإدارة المفاتيح
- **RFC 5280:** بنية مفتاح البنية العامة X.509 لشهادات الإنترنت وملفات CRL

2.10 مقاومة التحليل

2.10.1 مبادئ التصميم

الدفاع ضد التحليل:

- لا خوارزميات تشفير مخصصة/ملكية
- خوارزميات معيارية، تمت مراجعتها من قبل الأقران فقط
- تحديثات أمان منتظمة لمكتبات التشفير
- إلغاء خوارزميات ضعيفة
- استخدام التشفير المعتمد (GCM، Poly1305)

2.10.2 الأمان التشغيلي

دوران المفاتيح:

- إجراءات تجديد شهادة TLS
- دوران مفاتيح الجلسة (لكل جلسة لـ TLS)
- سياسات دوران بيانات اعتما�ه قاعدة البيانات

المراقبة والكشف:

- تسجيل محاولات المصادقة الفاشلة
- مراقبة انتهاء صلاحية الشهادات
- تسجيل فشل عملية المصافحة TLS
- اكتشاف الشذوذ لفشل التشفير
- تنبيه أحداث الأمان

---

3. التحكم في التنصت والتفويض

3.1 التحكم في الوصول للتنصت القانوني

تفويض إداري:

- الوصول إلى مسؤول النظام مطلوب للتكوين
- ضوابط وصول على مستوى قاعدة البيانات لاستعلامات CDR
- مقيد بواسطة IP/المصادقة الوصول إلى API
- تسجيل تدقيق لجميع الوصول

تكامل الإطار القانوني:

- تتبع أوامر التنصت (تكامل النظام الخارجي)
- قوائم تفويض معرفات الأهداف (وجهات نظر قاعدة البيانات)
- استعلامات محدودة زمنياً (عبارات WHERE SQL)
- فرض تلقائي عبر سياسات الوصول

3.2 الاحتفاظ بالبيانات والخصوصية

سياسات الاحتفاظ:

- الاحتفاظ بالرسائل النشطة: قابل للتكوين (الإعداد الافتراضي 24 ساعة في Mnesia)
- احتفاظ CDR: قابل للتكوين (6 أشهر إلى 2 سنوات نموذجية)
- الأرشفة التلقائية من Mnesia إلى SQL
- الحذف التلقائي لـ CDRs القديمة (استنادًا إلى cron)

حماية الخصوصية:

- خيار حذف جسم الرسالة بعد التسليم
- استبعاد جسم الرسالة من واجهة المستخدم/التصديرات
- تشفير قاعدة البيانات في حالة السكون
- تسجيل الوصول والمراقبة
- مبدأ الحد الأدنى من جمع البيانات

التكوين:

```
# config/runtime.exs
config :sms_c,
  # الاحتفاظ برسالة Mnesia قبل الأرشفة
  message_retention_hours: 24,

  # حذف جسم الرسالة بعد التسليم من أجل الخصوصية
  delete_message_body_after_delivery: false,  # تعيين true لوضع الخصوصية

  # التحكم في كتابة CDR
  cdr_enabled: true,

  # إعدادات الأرشفة الدفعة
  batch_insert_batch_size: 100,
  batch_insert_flush_interval_ms: 100
```

 انظر CONFIGURATION.md لجميع إعدادات الاحتفاظ

3.3 واجهات التسليم لجهات تنفيذ القانون

الواجهات القياسية:

1. الوصول عبر REST API:

- نقاط نهاية HTTPS لاسترجاع الرسائل
- تبادل البيانات بتنسيق JSON
- المصادقة

مرجع واجهة برمجة تطبيقات SMS-C

العودة إلى فهرس الوثائق | **الملف التعريفي الرئيسي ←**

مرجع كامل لجميع نقاط نهاية واجهة برمجة تطبيقات SMS-C REST مع أمثلة للطلبات/الاستجابات.

جدول المحتويات

- نظرة عامة على واجهة برمجة التطبيقات
- المصادقة
- تنسيقات الاستجابة الشائعة
- نقطة نهاية الحالة
- واجهة برمجة تطبيقات قائمة الرسائل
- واجهة برمجة تطبيقات SMS PDU الخام
- واجهة برمجة تطبيقات إدارة الموقع
- واجهة برمجة تطبيقات تسجيل الواجهة الأمامية
- واجهة برمجة تطبيقات تسجيل الأحداث
- واجهة برمجة تطبيقات رسا��ل MMS
- واجهة برمجة تطبيقات أحداث SS7
- رموز الأخطاء
- تحديد معدل الطلبات
- أفضل الممارسات

نظرة عامة على واجهة برمجة التطبيقات

تقدم واجهة برمجة تطبيقات SMS-C REST وصولاً برمجياً إلى وظائف تقديم الرسائل وتوجيهها وإدارتها.

عنوان URL الأساسي

```
https://api.example.com:8443/api
```

المنفذ الافتراضي: 8443 (قابل للتكوين)
البروتوكول: HTTPS (يتطلب TLS في الإنتاج)

نوع المحتوى

تستخدم جميع الطلبات والاستجابات JSON:

```
Content-Type: application/json
```

إصدار واجهة برمجة التطبيقات

الإصدار الحالي من واجهة برمجة التطبيقات هو 1 (ضمني). ستستخدم الإصدارات المستقبلية ترميز الإصدار في عنوان URL:

```
https://api.example.com:8443/api/v2/...
```

المصادقة

شهادات عميل TLS (موصى بها)

يجب أن تستخدم عمليات النشر في الإنتاج مصادقة شهادة عميل TLS:

```
curl --cert client.crt --key client.key \
  https://api.example.com:8443/api/status
```

مصادقة مفتاح واجهة برمجة التطبيقات

مصادقة مفتاح واجهة برمجة التطبيقات المخصصة عبر رأس `X-API-Key`:

```
curl -H "X-API-Key: your_api_key_here" \
  https://api.example.com:8443/api/status
```

قائمة بيضاء لعناوين IP

تقييد الوصول إلى واجهة برمجة التطبيقات لعناوين IP الموثوقة على مستوى جدار الحماية.

تنسيقات الاستجابة الشائعة

استجابة النجاح

```
{
  "data": {
    ...
  }
}
```

استجابة الخطأ

```
{
  "errors": {
    "detail": "رسالة خطأ تصف ما حدث خطأ"
  }
}
```

استجابة القائمة

```
{
  "data": [
    {...},
    {...}
  ]
}
```

نقطة نهاية الحالة

نقطة فحص الصحة لمراقبة وتحميل الموازنات.

الحصول على حالة واجهة برمجة التطبيقات

الطلب:

```
GET /api/status
```

الاستجابة (200 OK):

```
{
  "status": "ok",
  "application": "OmniMessage",
  "timestamp": "2025-10-30T12:34:56Z"
}
```

مثال:

```
curl https://api.example.com:8443/api/status
```

حالات الاستخدام:

- فحوصات صحة موازن التحميل
- مراقبة اتصال النظام
- التحقق من توفر الخدمة

واجهة برمجة تطبيقات قائمة الرسائل

نقاط نهاية أساسية لتقديم الرسائل وإدارتها.

قائمة الرسائل

استرجاع الرسائل من قائمة الانتظار.

الطلب:

```
GET /api/messages
```

رؤوس اختيارية:

- `smsc: frontend_name` - تصفية حسب SMSC الوجهة
- `include-unrouted: true|false|1|0` - تضمين الرسائل بدون تسجيل الموقع (الافتراضي: false)
  - `false` (افتراضي): إرجاع الرسائل التي تحتوي على توجيه صريح أو تسجيل موقع فقط
  - `true`: تضمين الرسائل بدون تسجيل الموقع (وضع متوافق مع الإصدارات السابقة)

معلمات الاستعلام:

- `status` - تصفية حسب الحالة: `pending`, `delivered`, `expired`, `dropped`
- `source_smsc` - تصفية حسب SMSC المصدر
- `dest_smsc` - تصفية حسب SMSC الوجهة
- `limit` - تحديد النتائج (الافتراضي: 100، الحد الأقصى: 1000)
- `offset` - إزاحة الصفحات

الاستجابة (200 OK):

```
{
  "data": [
    {
      "id": 12345,
      "source_msisdn": "+15551234567",
      "destination_msisdn": "+447700900000",
      "message_body": "Hello World",
      "source_smsc": "api_client",
      "dest_smsc": "uk_gateway",
      "status": "pending",
      "send_time": "2025-10-30T12:00:00Z",
      "deliver_time": null,
      "delivery_attempts": 0,
      "inserted_at": "2025-10-30T12:00:00Z"
    }
  ]
}
```

أمثلة:

الحصول على الرسائل المعلقة لـ SMSC معين (فقط مع توجيه صريح أو موقع):

```
curl -H "smsc: uk_gateway" \
  https://api.example.com:8443/api/messages
```

الحصول على الرسائل المعلقة بما في ذلك الرسائل غير الموجهة (متوافق مع الإصدارات السابقة):

```
curl -H "smsc: uk_gateway" \
  -H "include-unrouted: true" \
  https://api.example.com:8443/api/messages
```

الحصول على جميع الرسائل التي تم تسليمها:

```
curl "https://api.example.com:8443/api/messages?
status=delivered&limit=50"
```

الحصول على رسالة واحدة

استرجاع تفاصيل رسالة معينة.

الطلب:

```
GET /api/messages/:id
```

الاستجابة (200 OK):


```
{
  "data": {
    "id": 12345,
    "source_msisdn": "+15551234567",
    "destination_msisdn": "+447700900000",
    "message_body": "Hello World",
    "source_smsc": "api_client",
    "dest_smsc": "uk_gateway",
    "source_imsi": null,
    "dest_imsi": null,
    "message_parts": 1,
    "message_part_number": 1,
    "tp_data_coding_scheme": "00",
    "tp_user_data_header": null,
    "status": "pending",
    "send_time": "2025-10-30T12:00:00Z",
    "deliver_time": null,
    "expires": "2025-10-31T12:00:00Z",
    "deadletter": false,
    "delivery_attempts": 0,
    "charge_failed": false,
    "deliver_after": "2025-10-30T12:00:00Z",
    "raw_data_flag": false,
    "raw_sip_flag": false,
    "raw_pdu": null,
    "inserted_at": "2025-10-30T12:00:00Z",
    "updated_at": "2025-10-30T12:00:00Z"
  }
}
```

مثال:

```
curl https://api.example.com:8443/api/messages/12345
```

تقديم رسالة (متزامنة)

تقديم رسالة واستلام معرف الرسالة على الفور.

الطلب:

```
POST /api/messages
Content-Type: application/json
```

الجسم:

```
{
  "source_msisdn": "+15551234567",
  "destination_msisdn": "+447700900000",
  "message_body": "Hello World",
  "source_smsc": "api_client"
}
```

الحقول الاختيارية:

- `dest_smsc` - تجاوز قرار التوجيه
- `send_time` - جدولة للتسليم في المستقبل (ISO 8601)
- `message_parts` - إجمالي الأجزاء لرسالة متعددة الأجزاء
- `message_part_number` - رقم الجزء (مؤشر من 1)
- `tp_data_coding_scheme` - SMS DCS (00" :الافتراضي")
- `source_imsi` - IMSI المشترك المصدر
- `dest_imsi` - IMSI المشترك الوجهة

الاستجابة (201 تم الإنشاء):

```
{
  "data": {
    "id": 12345,
    "source_msisdn": "+15551234567",
    "destination_msisdn": "+447700900000",
    "message_body": "Hello World",
    "source_smsc": "api_client",
    "dest_smsc": "uk_gateway",
    "status": "pending",
    "send_time": "2025-10-30T12:00:00Z",
    "inserted_at": "2025-10-30T12:00:00Z"
  }
}
```

مثال:

```
curl -X POST https://api.example.com:8443/api/messages \
  -H "Content-Type: application/json" \
  -d '{
    "source_msisdn": "+15551234567",
    "destination_msisdn": "+447700900000",
    "message_body": "Hello World",
    "source_smsc": "api_client"
  }'
```

الأداء: ~70 رسالة/ثانية، 14 مللي ثانية متوسط وقت الاستجابة

استخدم عندما:

- تحتاج إلى معرف الرسالة على الفور
- معالجة الرسائل/ثانية
- تتطلب تأكيدًا فوريًا

تقديم رسالة (غير متزامنة)

تقديم رسالة مع معدل عالٍ (معالجة دفعات).

الطلب:

```
POST /api/messages/create_async
Content-Type: application/json
```

الجسم: نفس نقطة النهاية المتزامنة

الاستجابة (202 مقبول):

```
{
  "data": {
    "status": "accepted",
    "message": "تمت إضافة الرسالة إلى قائمة الانتظار للمعالجة"
  }
}
```

مثال:

```
curl -X POST
https://api.example.com:8443/api/messages/create_async \
  -H "Content-Type: application/json" \
  -d '{
    "source_msisdn": "+15551234567",
    "destination_msisdn": "+447700900000",
    "message_body": "رسالة إشعار جماعي",
    "source_smsc": "bulk_api"
  }'
```

الأداء: ~4,650 رسالة/ثانية، 0.22 مللي ثانية متوسط وقت الاستجابة

الكمون: تظهر الرسالة في قاعدة البيانات خلال 100 مللي ثانية (قاب�ة للتكوين)

استخدم عندما:

- الرسائل الجماعية ذات الحجم الكبير (> 100 رسالة/ثانية)
- لا تحتاج إلى معرف الرسالة في استجابة واجهة برمجة التطبيقات
- معدل الإنتاج أكثر أهمية من التأكيد الفوري

تحديث الرسالة

تحديث جزئي لحقول الرسالة.

الطلب:

```
PATCH /api/messages/:id
Content-Type: application/json
```

الجسم:

```
{
  "dest_smsc": "alternate_gateway",
  "deliver_after": "2025-10-30T14:00:00Z"
}
```

الحقول القابلة للتحديث:

- `dest_smsc` - تغيير الوجهة
- `deliver_after` - تأخير التسليم
- `message_body` - تحديث نص الرسالة
- `status` - تغيير الحالة

الاستجابة (200 OK):

```
{
  "data": {
    "id": 12345,
    "dest_smsc": "alternate_gateway",
    "deliver_after": "2025-10-30T14:00:00Z",
    ...
  }
}
```

مثال:

```
curl -X PATCH https://api.example.com:8443/api/messages/12345 \
  -H "Content-Type: application/json" \
  -d '{
    "dest_smsc": "backup_gateway"
  }'
```

وضع علامة على الرسالة كتم تسليمها

وضع علامة على رسالة على أنها تم تسليمها بنجاح.

الطلب:

```
POST /api/messages/:id/mark_delivered
Content-Type: application/json
```

الجسم:

```
{
  "dest_smsc": "uk_gateway"
}
```

الاستجابة (200 OK):

```
{
  "data": {
    "id": 12345,
    "status": "delivered",
    "deliver_time": "2025-10-30T12:05:30Z",
    "dest_smsc": "uk_gateway",
    ...
  }
}
```

مثال:

```
curl -X POST
https://api.example.com:8443/api/messages/12345/mark_delivered \
  -H "Content-Type: application/json" \
  -d '{
    "dest_smsc": "uk_gateway"
  }'
```

حالة الاستخدام: يتم استدعاؤها بواسطة الأنظمة الأمامية بعد التسليم الناجح

زيادة محاولة التسليم

زيادة عداد إعادة المحاولة وتطبيق التراجع الأسي.

الطلب:

```
PUT /api/messages/:id
```

الاستجابة (200 OK):

```
{
  "data": {
    "id": 12345,
    "delivery_attempts": 2,
    "deliver_after": "2025-10-30T12:08:00Z",
    ...
  }
}
```

حساب التراجع:

```
deliver_after = now + 2^(delivery_attempts) minutes
```

مثال:

```
curl -X PUT https://api.example.com:8443/api/messages/12345
```

حالة الاست��دام: يتم استدعاؤها بواسطة الواجهة الأمامية بعد فشل التسليم لجدولة إعادة المحاولة

حذف الرسالة

إزالة الرسالة من قائمة الانتظار.

الطلب:

```
DELETE /api/messages/:id
```

الاستجابة (204 لا توجد محتويات)

مثال:

```
curl -X DELETE https://api.example.com:8443/api/messages/12345
```

تحذير: يؤدي حذف الرسائل إلى إزالتها بشكل دائم. استخدم بحذر.

الخام SMS PDU واجهة برمجة تطبيقات

خام لتحقيق أقصى توافق مع الأنظمة (وحدة بيانات البروتوكول) PDU كـ SMS تقديم رسائل القديمة.

خام (متزامن) SMS تقديم

الطلب:

```
POST /api/messages_raw
Content-Type: application/json
```

الجسم:

```
{
  "pdu": "0001000B916407007009F0000004D4F29C0E",
  "source_smsc": "legacy_system"
}
```

تنسيق PDU: SMS TPDU (وحدة بيانات البروتوكول للنقل) مشفرة بتنسيق Hex

الاستجابة: (201 تم الإنشاء)

```
{
  "data": {
    "id": 12346,
    "source_msisdn": "+447700900000",
    "destination_msisdn": "+447700900000",
    "message_body": "اختبار",
    "source_smsc": "legacy_system",
    "raw_pdu": "0001000B916407007009F0000004D4F29C0E",
    ...
  }
}
```

مثال:

```
curl -X POST https://api.example.com:8443/api/messages_raw \
  -H "Content-Type: application/json" \
  -d '{
    "pdu": "0001000B916407007009F0000004D4F29C0E",
    "source_smsc": "legacy_system"
  }'
```

تقديم SMS خام (غير متزامن)

الطلب:

```
POST /api/messages_raw/async
Content-Type: application/json
```

الجسم: نفس المتزامن

الاستجابة (202 مقبول):

```
{
  "data": {
    "status": "accepted",
    "message": "تمت إضافة PDU للمعالجة إلى قائمة الانتظار"
  }
}
```

مثال:

```
curl -X POST https://api.example.com:8443/api/messages_raw/async \
  -H "Content-Type: application/json" \
  -d '{
    "pdu": "0001000B916407007009F0000004D4F29C0E",
    "source_smsc": "legacy_gateway"
  }'
```

معالجة PDU

يقوم النظام تلقائيًا بـ:

1. فك تشفير PDU باستخدام معايير SMS (3GPP TS 23.040)
2. استخراج أرقام الهواتف، نص الرسالة، DCS
3. اكتشاف تقارير التسليم (CP-ACK، RP-ACK، إلخ)
4. إجراء بحث IMSI إلى MSISDN إذا لزم الأمر
5. تطبيق قواعد التوجيه
6. تخزين PDU الأصلي للرجوع إليه

كشف تقرير التسليم:

- CP-ACK، CP-ERROR - تأكيدات بروتوكول الاتصال
- RP-ACK، RP-ERROR، RP-SMMA - استجابات بروتوكول الترحيل
- يتم تسجيل تقارير التسليم ولكن لا يتم تخزينها كرسائل

واجهة برمجة تطبيقات إدارة الموقع

إدارة معلومات موقع المشترك لتسليم الرسائل الموجهة إلى الهاتف المحمول.

قائمة المواقع

الطلب:

```
GET /api/locations
```

الاستجابة (200 OK):

```
{
  "data": [
    {
      "id": 1,
      "msisdn": "+15551234567",
      "imsi": "001001000000001",
      "location": "msc1.region1.example.com",
      "ran_location": "cell_tower_12345",
      "imei": "123456789012345",
      "ims_capable": true,
      "csfb": false,
      "registered": true,
      "expires": "2025-10-30T13:00:00Z",
      "user_agent": "Samsung Galaxy",
      "inserted_at": "2025-10-30T12:00:00Z",
      "updated_at": "2025-10-30T12:00:00Z"
    }
  ]
}
```

مثال:

```
curl https://api.example.com:8443/api/locations
```

الحصول على الموقع

الطلب:

```
GET /api/locations/:id
```

الاستجابة (200 OK):

```
{
  "data": {
    "id": 1,
    "msisdn": "+15551234567",
    "imsi": "001001000000001",
    ...
  }
}
```

مثال:

```
curl https://api.example.com:8443/api/locations/1
```

إنشاء/تحديث الموقع

إنشاء موقع جديد أو تحديث الموقع الموجود بناءً على IMSI (معرف فريد).

الطلب:

```
POST /api/locations
Content-Type: application/json
```

الجسم:

```
{
  "msisdn": "+15551234567",
  "imsi": "001001000000001",
  "location": "msc1.region1.example.com",
  "ran_location": "cell_tower_12345",
  "imei": "123456789012345",
  "ims_capable": true,
  "csfb": false,
  "registered": true,
  "expires": "2025-10-30T13:00:00Z",
  "user_agent": "Samsung Galaxy"
}
```

الحقول المطلوبة:

- `imsi` - معرف المشترك الفريد
- `msisdn` - رقم الهاتف

الحقول الاختيارية:

- `location` - عنوان MSC/VLR
- `ran_location` - معرف برج/قطاع الخلية
- `imei` - معرف الجهاز
- `ims_capable` - قدرة IMS VoLTE
- `csfb` - علامة التراجع إلى الدائرة
- `registered` - مسجل حاليًا
- `expires` - انتهاء التسجيل
- `user_agent` - طراز/معلومات الجهاز

الاستجابة (200 تم الإنشاء أو 201 OK):

```
{
  "data": {
    "id": 1,
    "msisdn": "+15551234567",
    ...
  }
}
```

مثال:

```
curl -X POST https://api.example.com:8443/api/locations \
  -H "Content-Type: application/json" \
  -d '{
    "msisdn": "+15551234567",
    "imsi": "001001000000001",
    "location": "msc1.region1.example.com",
    "ims_capable": true,
    "registered": true
  }'
```

حالة الاستخدام: يتم استدعاؤها بواسطة أنظمة إدارة الحركة المشترك (HSS، MME، إلخ) عندما يسجل

تحديث الموقع

الطلب:

```
PATCH /api/locations/:id
Content-Type: application/json
```

الجسم: تحديث جزئي مع أي حقول موقع

الاستجابة (200 OK):

```
{
  "data": {
    "id": 1,
    ...
  }
}
```

مثال:

```
curl -X PATCH https://api.example.com:8443/api/locations/1 \
  -H "Content-Type: application/json" \
  -d '{
    "location": "msc2.region2.example.com",
    "ran_location": "cell_tower_67890"
  }'
```

حذف الموقع

الطلب:

```
DELETE /api/locations/:id
```

الاستجابة (204 لا توجد محتويات)

مثال:

```
curl -X DELETE https://api.example.com:8443/api/locations/1
```

حالة الاستخدام: يتم استدعاؤها عندما يقوم المشترك بإلغاء التسجيل أو انتهاء الوقت

واجهة برمجة تطبيقات تسجيل الواجهة الأمامية

تتبع وإدارة اتصالات SMSC للواجهة الأمامية.

قائمة جميع الواجهات الأمامية

الطلب:

```
GET /api/frontends
```

الاستجابة (200 OK):

```
{
  "data": [
    {
      "id": 1,
      "frontend_name": "uk_gateway_1",
      "frontend_type": "smpp",
      "ip_address": "10.0.1.50",
      "hostname": "gateway1.uk.example.com",
      "uptime_seconds": 86400,
      "configuration": {
        "max_throughput": 1000,
        "bind_type": "transceiver"
      },
      "status": "active",
      "expires_at": "2025-10-30T12:02:00Z",
      "last_seen_at": "2025-10-30T12:00:30Z",
      "inserted_at": "2025-10-29T12:00:00Z",
      "updated_at": "2025-10-30T12:00:30Z"
    }
  ]
}
```


مثال:

```
curl https://api.example.com:8443/api/frontends
```

قائمة الواجهات الأمامية النشطة فقط

الطلب:

```
GET /api/frontends/active
```

الاستجابة (200 OK): نفس التنسيق، فقط الواجهات الأمامية النشطة

مثال:

```
curl https://api.example.com:8443/api/frontends/active
```

حالة الاستخدام: الحصول على قائمة الوجهات المتاحة للتوجيه

الحصول على إحصائيات الواجهة الأمامية

الطلب:

```
GET /api/frontends/stats
```

الاستجابة (200 OK):

```
{
  "data": {
    "active_count": 5,
    "expired_count": 2,
    "unique_frontends": 7,
    "total_registrations": 1523
  }
}
```

مثال:

```
curl https://api.example.com:8443/api/frontends/stats
```

الحصول على تاريخ الواجهة الأمامية

الطلب:

```
GET /api/frontends/history/:name
```

الاستجابة (200 OK):

```
{
  "data": [
    {
      "id": 1,
      "frontend_name": "uk_gateway_1",
      "status": "active",
      "inserted_at": "2025-10-30T12:00:00Z",
      ...
    },
    {
      "id": 2,
      "frontend_name": "uk_gateway_1",
      "status": "expired",
      "inserted_at": "2025-10-29T12:00:00Z",
      ...
    }
  ]
}
```

مثال:

```
curl
https://api.example.com:8443/api/frontends/history/uk_gateway_1
```

تسجيل الواجهة الأمامية

تسجيل أو تحديث اتصال الواجهة الأمامية.

الطلب:

```
POST /api/frontends/register
Content-Type: application/json
```

الجسم:

```
{
  "frontend_name": "uk_gateway_1",
  "frontend_type": "smpp",
  "ip_address": "10.0.1.50",
  "hostname": "gateway1.uk.example.com",
  "uptime_seconds": 86400,
  "configuration": {
    "max_throughput": 1000,
    "bind_type": "transceiver",
    "system_id": "gateway1"
  }
}
```

الحقول المطلوبة:

- `frontend_name` - معرف فريد للواجهة الأمامية
- `frontend_type` - النوع: `smpp`، `sip`، `http`، إلخ.

الحقول الاختيارية:

- `ip_address` - عنوان IP للواجهة الأمامية
- `hostname` - اسم المضيف للواجهة الأمامية
- `uptime_seconds` - وقت التشغيل منذ البداية
- `configuration` - كائن تكوين مخصص

الاستجابة (201 تم الإنشاء):

```
{
  "data": {
    "id": 1,
    "frontend_name": "uk_gateway_1",
    "status": "active",
    "expires_at": "2025-10-30T12:01:30Z",
    ...
  }
}
```

مثال:

```
curl -X POST https://api.example.com:8443/api/frontends/register \
  -H "Content-Type: application/json" \
  -d '{
    "frontend_name": "uk_gateway_1",
    "frontend_type": "smpp",
    "ip_address": "10.0.1.50",
    "hostname": "gateway1.uk.example.com"
  }'
```

مهلة التسجيل: 90 ثانية (يجب على الواجهات الأمامية إعادة التسجيل كل 60-90 ثانية)

حالة الاستخدام: يتم استدعاؤها بشكل دوري بواسطة الأنظمة الأمامية للحفاظ على الحالة النشطة

واجهة برمجة تطبيقات تسجيل الأحداث

تتبع أحداث دورة حياة الرسالة.

الحصول على أحداث الرسالة

الطلب:

```
GET /api/events/:message_id
```

الاستجابة (200 OK):

```
{
  "data": [
    {
      "event_epoch": 1698672000,
      "name": "message_inserted",
      "description": "تم إدخال الرسالة في قائمة الانتظار",
      "event_source": "node1@server.example.com"
    },
    {
      "event_epoch": 1698672001,
      "name": "message_routed",
      "description": "تم توجيهها إلى uk_gateway عبر route_id=42",
      "event_source": "node1@server.example.com"
    },
    {
      "event_epoch": 1698672005,
      "name": "message_delivered",
      "description": "تم التسليم بنجاح",
      "event_source": "node2@server.example.com"
    }
  ]
}
```

مثال:

```
curl https://api.example.com:8443/api/events/12345
```

أنواع الأحداث:

- `message_inserted` - تم إنشاء الرسالة
- `message_routed` - تم اتخاذ قرار التوجيه
- `message_delivered` - التسليم الناجح
- `message_failed` - فشل التسليم
- `message_dropped` - تم إسقاطه بواسطة التوجيه
- `auto_reply_sent` - تم تفعيل الرد التلقائي
- `number_translated` - تم تطبيق تحويل الرقم
- `routing_failed` - لم يتم العثور على مسار
- `charging_failed` - خطأ في نظام الشحن

تسجيل حدث

الطلب:

```
POST /api/events
Content-Type: application/json
```

الجسم:

```
{
  "message_id": 12345,
  "name": "custom_event",
  "description": "وصف الحدث المخصص",
  "event_source": "external_system"
}
```

الاستجابة (201 تم الإنشاء):

```
{
  "data": {
    "message_id": 12345,
    "name": "custom_event",
    "description": "وصف الحدث المخصص",
    "event_source": "external_system",
    "event_epoch": 1698672010
  }
}
```

مثال:

```
curl -X POST https://api.example.com:8443/api/events \
  -H "Content-Type: application/json" \
  -d '{
    "message_id": 12345,
    "name": "external_delivery_confirmed",
    "description": "تم التأكيد من قبل النظام السفلي"
  }'
```

احتفاظ الأحداث: 7 أيام (قابل للتكوين)

واجهة برمجة تطبيقات رسائل MMS

إدارة رسائل خدمة الرسائل متعددة الوسائط (MMS).

قائمة رسائل MMS

الطلب:

```
GET /api/mms_messages
```

الاستجابة (200 OK): إضافية MMS مع حقول SMS مشابهة لرسائل

إنشاء رسالة MMS

الطلب:

```
POST /api/mms_messages
Content-Type: application/json
```

الجسم:

```
{
  "source_msisdn": "+15551234567",
  "destination_msisdn": "+447700900000",
  "subject": "صورة",
  "content_type": "image/jpeg",
  "content_location": "https://cdn.example.com/media/12345.jpg",
  "message_size": 524288
}
```

الاستجابة (201 تم الإنشاء): كائن رسالة MMS كامل

واجهة برمجة تطبيقات أحداث SS7

تتبع أحداث إشارة SS7.

قائمة أحداث SS7

الطلب:

```
GET /api/ss7_events
```

الاستجابة (200 OK):

```
{
  "data": [
    {
      "id": 1,
      "event_type": "MAP_UPDATE_LOCATION",
      "imsi": "001001000000001",
      "msisdn": "+15551234567",
      "timestamp": "2025-10-30T12:00:00Z",
      ...
    }
  ]
}
```

إنشاء حدث SS7

الطلب:

```
POST /api/ss7_events
Content-Type: application/json
```

الجسم:

```
{
  "event_type": "MAP_UPDATE_LOCATION",
  "imsi": "001001000000001",
  "msisdn": "+15551234567"
}
```

الاستجابة (201 تم الإنشاء): كائن الحدث الكامل

رموز الأخطاء

رموز حالة HTTP

الرمز	المعنى	الوصف
200	حسنًا	الطلب ناجح
201	تم الإنشاء	تم إنشاء المورد بنجاح
202	مقبول	تم قبول الطلب للمعالجة
204	لا توجد محتويات	تم الحذف بنجاح
400	طلب غير صالح	تنسيق الطلب غير صالح
401	غير مصرح	تتطلب المصادقة
403	محظور	أذونات غير كافية
404	غير موج��د	المورد غير موجود
422	كيان غير قابل للمعالجة	أخطاء التحقق
429	عدد كبير جدًا من الطلبات	تم تجاوز حد المعدل
500	خطأ في الخادم الداخلي	خطأ في الخادم
503	الخدمة غير متاحة	غير متاحة مؤقتًا

تنسيق استجابة الخطأ

```
{
  "errors": {
    "detail": "فشل التحقق: destination_msisdn مطلوب"
  }
}
```

رسائل الخطأ الشائعة

الخطأ	السبب	الحل
"destination_msisdn مطلوب"	حقل مطلوب مفقود	تضمين destination_msisdn في الطلب
"تنسيق رقم الهاتف غير صالح"	رقم مشوه	استخدم تنسيق E.164: +15551234567
"الرسالة طويلة جدًا"	تتجاوز حد الحجم	قسمها إلى أجزاء متعددة
"لم يتم العثور على مسار"	فشل التوجيه	تحقق من تكوين التوجيه
"فشل الشحن"	خطأ في OCS	تحقق من اتصال نظام الشحن
"لم يتم العثور على الرسالة"	معرف رسالة غير صالح	تحقق من وجود المعرف
"الواجهة الأمام��ة غير مسجلة"	SMSC غير معروف	سجل الواجهة الأمامية أولاً

تحديد معدل الطلبات

الحدود الافتراضية

نقطة النهاية	الحد	النافذة
POST /api/messages	100 req/sec	لكل IP
POST /api/messages/create_async	1000 req/sec	لكل IP
POST /api/messages_raw	100 req/sec	لكل IP
GET /api/*	1000 req/sec	لكل IP

رؤوس حد المعدل

```
X-RateLimit-Limit: 100
X-RateLimit-Remaining: 95
X-RateLimit-Reset: 1698672060
```

تجاوز حد المعدل

الاستجابة (429 عدد كبير جدًا من الطلبات):

```
{
  "errors": {
    "detail": "تم تجاوز حد المعدل. حاول مرة أخرى بعد 5 ثوان."
  }
}
```

أفضل الممارسات

تقديم الرسائل

1. **استخدم غير متزامن للدفعات**: استخدم `/create_async` لأكثر من 100 رسالة/ثانية
2. **تضمين source_smsc**: حدد نظامك دائمًا
3. **تحقق من الأرقام**: استخدم تنسيق E.164 (رمز البلد+)
4. **تعامل مع الأخطاء**: نفذ منطق إعادة المحاولة لأخطاء 5xx
5. **تحقق من التوجيه**: اختبر المسارات قبل التقديم بالجملة

تكامل الواجهة الأمامية

1. **سجل بانتظام**: أعد التسجيل كل 60 ثانية
2. **استفسر عن الرسائل**: استعلم باستخدام رأس `smsc` لرسائلك
3. **بحكمة استخدم include-unrouted**: بشكل افتراضي، يتم إرجاع الرسائل التي تحتوي على توجيه صريح أو تسجيل موقع فقط. قم بتعيين `include-unrouted: true` فقط إذا كنت بحاجة إلى سلوك متوافق مع الإصدارات السابقة لاستلام جميع الرسائل غير الموجهة
4. **وضع علامة على التسليم**: اتصل دائمًا mark_delivered بعد النجاح
5. **زيادة عند الفشل**: استخدم نقطة النهاية PUT لمنطق إعادة المحاولة
6. **راقب الأحداث**: تحقق من سجل الأحداث لمشاكل التسليم

الأداء

1. **تجميع الاتصالات**: إعادة استخدام اتصالات HTTP
2. **طلبات الدفعة**: تجميع رسائل متعددة لكل طلب
3. **المعالجة المتوازية**: إجراء مكالمات واجهة برمجة التطبيقات المتزامنة
4. **راقب المقاييس**: راقب Prometheus لاختناقات
5. **تعيين مهلات**: استخدم مهلة 30 ثانية لمكالمات واجهة برمجة التطبيقات

الأمان

1. **استخدم TLS**: استخدم دائمًا HTTPS في الإنتاج

2. **تحقق من الشهادات**: لا تتخطى التحقق من الشهادات
3. **تدوير مفاتيح واجهة برمجة التطبيقات**: قم بتغيير المفاتيح بانتظام
4. **قائمة بيضاء لعناوين IP**: تقييد المصادر المعروفة
5. **سجل نشاط واجهة برمجة التطبيقات**: راقب الأنماط المشبوهة

معالجة الأخطاء

1. **إعادة المحاولة لأخطاء 5xx**: تكون أخطاء الخادم عادةً مؤقتة
2. **لا تعيد المحاولة لأخطاء 4xx**: تحتاج أخطاء العميل إلى إصلاحات في التعليمات البرمجية
3. **التراجع الأسي**: انتظر لفترة أطول بين إعادة المحاولات
4. **قاطع الدائرة**: توقف بعد الفشل المتكرر
5. **تنبيه على الأنماط**: راقب معدلات الأخطاء

مثال على التكامل (Python)

```python
import requests
import time

class SMSCClient:
    def __init__(self, base_url, api_key=None):
        self.base_url = base_url
        self.session = requests.Session()
        if api_key:
            self.session.headers.update({"X-API-Key": api_key})

    def submit_message(self, from_num, to_num, text,
async_mode=False):
        endpoint = "/messages/create_async" if async_mode else
"/messages"
        url = f"{self.base_url}{endpoint}"

        payload = {
            "source_msisdn": from_num,
            "destination_msisdn": to_num,
            "message_body": text,
            "source_smsc": "python_client"
        }

        try:
            response = self.session.post(url, json=payload,
timeout=30)
            response.raise_for_status()
            return response.json()["data"]
        except requests.exceptions.RequestException as e:
            print(f"خطأ في واجهة برمجة التطبيقات: {e}")
            return None

    def get_pending_messages(self, smsc_name,
include_unrouted=False):
        url = f"{self.base_url}/messages"
        headers = {"smsc": smsc_name}

        # تضمين الرسائل غير الموجهة إذا تم الطلب (وضع متوافق مع الإصدارات
السابقة)
        if include_unrouted:
            headers["include-unrouted"] = "true"
```

```
        try:
            response = self.session.get(url, headers=headers,
timeout=30)
            response.raise_for_status()
            return response.json()["data"]
        except requests.exceptions.RequestException as e:
            print(f"خطأ في واجهة برمجة التطبيقات: {e}")
            return []

    def mark_delivered(self, message_id, smsc_name):
        url = f"
{self.base_url}/messages/{message_id}/mark_delivered"
        payload = {"dest_smsc": smsc_name}

        try:
            response = self.session.post(url, json=payload,
timeout=30)
            response.raise_for_status()
            return True
        except requests.exceptions.RequestException as e:
            print(f"خطأ في واجهة برمجة التطبيقات: {e}")
            return False

# الاستخدام
client = SMSCClient("https://api.example.com:8443/api",
api_key="your_key")

# تقديم رسالة واحدة
result = client.submit_message("+15551234567", "+447700900000",
"Hello")
print(f"معرف الرسالة: {result['id']}")

# تقديم رسائل جماعية (غير متزامنة)
for i in range(1000):
    client.submit_message("+15551234567", f"+44770090{i:04d}",
f"Bulk {i}", async_mode=True)

# حلقة استعلام الواجهة الأمامية
while True:
    # الحصول على الرسائل مع ��وجيه صريح أو تسجيل الموقع
    messages = client.get_pending_messages("my_gateway")

    # لسلوك متوافق مع الإصدارات السابقة include_unrouted=True أو استخدم
```

```
    # messages = client.get_pending_messages("my_gateway",
include_unrouted=True)

    for msg in messages:
        # تسليم الرسالة عبر بروتوكولك
        success = deliver_via_smpp(msg)

        if success:
            client.mark_delivered(msg["id"], "my_gateway")
        else:
            # زيادة لإعادة المحاولة
            requests.put(f"
{client.base_url}/messages/{msg['id']}")

    time.sleep(5)  # استعلام كل 5 ثوان
```

سجل تغييرات واجهة برمجة التطبيقات

الإصدار 1 (الحالي)

- الإصدار الأول
- قائمة الرسائل CRUD
- تقديم PDU الخام
- إدارة الموقع
- تسجيل الواجهة الأمامية
- تسجيل الأحداث

الميزات المخطط لها

- تقديم الرسائل دفعة واحدة (طلب واحد، رسائل متعددة)
- قوالب الرسائل
- واجهة برمجة التطبيقات للتسليم المجدول
- Webhooks في الوقت الحقيقي للأحداث
- GraphQL نقطة نهاية واجهة برم��ة التطبيقات
- OAuth2 مصادقة

لأي أسئلة أو مشاكل تتعلق بواجهة برمجة التطبيقات، تحقق من دليل استكشاف الأخطاء أو اتصل بالدعم.

مرجع مخطط CDR (سجل تفاصيل المكالمات)

← الوثيقة الرئيسية | العودة إلى فهرس الوثائق

مرجع كامل لجدول قاعدة بيانات CDR المستخدم لتخزين الرسائل على المدى الطويل، والفوترة، والتحليلات.

جدول المحتويات

- نظرة عامة
- مخطط الجدول
- وصف الحقول
- أمثلة SQL
- الفهارس
- أنواع البيانات حسب قاعدة البيانات
- اعتبارات الخصوصية
- سياسة الاحتفاظ والأرشفة
- تكامل الفوترة

نظرة عامة

يخزن جدول `cdrs` سجلات تفاصيل المكالمات لجميع رسائل SMS. يتم المعالجة بواسطة النظام عندما CDRs كتابة:

- يتم تسليم الرسائل بنجاح
- تنتهي صلاحية الرسائل دون تسليم
- تفشل الرسائل بشكل دائم
- يتم رفض الرسائل

توفر CDRs تخزينًا طويل الأمد منفصلًا عن قاعدة بيانات Mnesia التشغيلية، مما يمكّن من:

- الفوترة وإصدار الفواتير
- التحليلات والتقارير
- الامتثال والتدقيق
- تاريخ الرسائل بعد فترة الاحتفاظ في Mnesia

مخطط الجدول

MySQL / MariaDB

```
CREATE TABLE cdrs (
  id BIGINT AUTO_INCREMENT PRIMARY KEY,

  -- تعريف الرسالة
  message_id BIGINT NOT NULL,

  -- أرقام الهواتف
  calling_number VARCHAR(255) NOT NULL,
  called_number VARCHAR(255) NOT NULL,

  -- توجيه SMSC
  source_smsc VARCHAR(255),
  dest_smsc VARCHAR(255),

  -- معلومات العقدة (للنشر المجمّع)
  origin_node VARCHAR(255),
  destination_node VARCHAR(255),

  -- الطوابع الزمنية
  submission_time DATETIME NOT NULL,
  delivery_time DATETIME,
  expiry_time DATETIME,

  -- الحالة والبيانات الوصفية
  status VARCHAR(50) NOT NULL,
  delivery_attempts INT DEFAULT 0,
  message_parts INT,
  deadletter BOOLEAN DEFAULT FALSE,

  -- نص الرسالة الاختياري (تحكمات الخصوصية)
  message_body TEXT,

  -- الطوابع الزمنية للتدقيق
  inserted_at DATETIME NOT NULL,
  updated_at DATETIME NOT NULL,

  -- الفهارس
  INDEX idx_cdrs_message_id (message_id),
```

```
  INDEX idx_cdrs_calling_number (calling_number),
  INDEX idx_cdrs_called_number (called_number),
  INDEX idx_cdrs_status (status),
  INDEX idx_cdrs_submission_time (submission_time),
  INDEX idx_cdrs_dest_smsc (dest_smsc)
);
```

PostgreSQL

```sql
CREATE TABLE cdrs (
  id BIGSERIAL PRIMARY KEY,

  -- تعريف الرسالة
  message_id BIGINT NOT NULL,

  -- أرقام الهواتف
  calling_number VARCHAR(255) NOT NULL,
  called_number VARCHAR(255) NOT NULL,

  -- توجيه SMSC
  source_smsc VARCHAR(255),
  dest_smsc VARCHAR(255),

  -- معلومات العقدة (للنشر المجمّع)
  origin_node VARCHAR(255),
  destination_node VARCHAR(255),

  -- الطوابع الزمنية
  submission_time TIMESTAMP NOT NULL,
  delivery_time TIMESTAMP,
  expiry_time TIMESTAMP,

  -- الحالة والبيانات الوصفية
  status VARCHAR(50) NOT NULL,
  delivery_attempts INTEGER DEFAULT 0,
  message_parts INTEGER,
  deadletter BOOLEAN DEFAULT FALSE,

  -- نص الرسالة الاختياري (تحكمات الخصوصية)
  message_body TEXT,

  -- الطوابع الزمنية للتدقيق
  inserted_at TIMESTAMP NOT NULL,
  updated_at TIMESTAMP NOT NULL
);

-- الفهارس
CREATE INDEX idx_cdrs_message_id ON cdrs(message_id);
CREATE INDEX idx_cdrs_calling_number ON cdrs(calling_number);
CREATE INDEX idx_cdrs_called_number ON cdrs(called_number);
```

```
CREATE INDEX idx_cdrs_status ON cdrs(status);
CREATE INDEX idx_cdrs_submission_time ON cdrs(submission_time);
CREATE INDEX idx_cdrs_dest_smsc ON cdrs(dest_smsc);
```

وصف الحقول

المفتاح الأساسي

الحقل	النوع	قابل للإلغاء	الوصف
`id`	BIGINT	لا	المفتاح الأساسي المتزايد تلقائيًا لسجل CDR

تعريف الرسالة

الحقل	النوع	قابل للإلغاء	الوصف
`message_id`	BIGINT	لا	معرف الرسالة الفريد من قائمة رسائل SMS-C. يشير إلى معرف الرسالة الأصلية في Mnesia.

أرقام الهواتف

الحقل	النوع	قابل للإلغاء	الوصف
`calling_number`	VARCHAR(255)	لا	المصدر (رقم الهاتف MSISDN المحمول) للمرسل. عادةً ما يك��ن بتنسيق E.164 (مثل: 15551234567+).
`called_number`	VARCHAR(255)	لا	الوجهة (رقم الهاتف MSISDN المحمول) للمستلم. عادةً ما يكون بتنسيق E.164 (مثل: 15551234567+).

توجيه SMSC

الحقل	النوع	قابل للإلغاء	الوصف
`source_smsc`	VARCHAR(255)	نعم	المصدر الذي SMSC اسم أو معرف قدم الرسالة. NULL إذا تم تقديمها عبر API أو واجهة غير SMSC أخرى.
`dest_smsc`	VARCHAR(255)	نعم	الوجهة التي SMSC اسم أو معرف سلمت (أو حاولت تسليم) الرسالة. NULL إذا لم يتم توجيه الرسالة أبدًا.

معلومات العقدة

للنشر المجمّع، تتبع أي العقدة تعاملت مع الرسالة:

الحقل	النوع	قابل للإلغاء	الوصف
`origin_node`	VARCHAR(255)	نعم	اسم عقدة Erlang حيث تم استلام الرسالة في الأصل (مثل: "sms@node1.example.com"). مفيد لاستكشاف الأخطاء وتحليل توزيع الحمل.
`destination_node`	VARCHAR(255)	نعم	اسم عقدة Erlang حيث تم تسليم الرسالة منه (إذا كانت مختلفة عن الأصل). NULL للنشر أحادي العقدة أو إذا لم يتم تسليم الرسالة.

الطوابع الزمنية

تخزن جميع الطوابع الزمنية بتوقيت UTC:

الحقل	النوع	قابل للإلغاء	الوصف
`submission_time`	DATETIME	لا	متى تم تقديم الرسالة لأول مرة إلى SMS-C. يستخدم كوقت البدء لحسابات الفوترة.
`delivery_time`	DATETIME	نعم	متى تم تسليم الرسالة بنجاح. NULL إذا انتهت صلاحية الرسالة، أو فشلت، أو تم رفضها.
`expiry_time`	DATETIME	نعم	متى انتهت صلاحية الرسالة (أصبحت غير قابلة للتسليم). NULL إذا تم تسليم الرسالة أو لا تزال قيد الانتظار.

حساب مدة التسليم:

```
TIMESTAMPDIFF(SECOND, submission_time, delivery_time) AS
delivery_duration_seconds
```

الحالة والبيانات الوصفية

الحقل	النوع	قابل للإلغاء	الوصف
`status`	VARCHAR(50)	لا	الحالة النهائية للرسالة. القيم الصالحة: `delivered`, `expired`, `failed`, `rejected`
`delivery_attempts`	INT	لا	عدد محاولات التسليم التي تمت قبل الحالة النهائية. الافتراضي: 0. النطاق: 0-255 عادةً.
`message_parts`	INT	نعم	عدد أجزاء SMS للرسائل المجمعة. 1 للرسائل ذات الجزء الواحد، 2+ للرسائل متعددة الأجزاء. NULL إذا كان غير معروف.
`deadletter`	BOOLEAN	لا	ما إذا كانت الرسالة قد تم نقلها إلى قائمة الرسائل الميتة. TRUE تشير إلى أن الرسالة لم تتمكن من التسليم واستنفدت جميع المحاولات. الافتراضي: FALSE

قيم الحالة:

الحالة	الوصف	قابل للفوترة	وقت التسليم
`delivered`	تم تسليمها بنجاح إلى المستلم	نعم	محدد
`expired`	تجاوزت فترة الصلاحية دون تسليم	يعتمد على سياسة الفوترة	NULL
`failed`	فشل دائم في التسليم (رقم غير صالح، إلخ)	يعتمد على سياسة الفوترة	NULL
`rejected`	تم رفضها بواسطة قواعد التوجيه أو التحقق	لا	NULL

نص الرسالة

الحقل	النوع	قابل للإلغاء	الوصف
`message_body`	TEXT	نعم	محتوى الرسالة الفعلي. يمكن أن يكون NULL إذا كانت خاصية `delete_message_body_after_delivery` مفعلة لأغراض الخصوصية. الحد الأقصى للطول عادةً 65,535 حرفًا) يختلف حسب قاعدة البيانات لنوع TEXT).

أوضاع الخصوصية:

- **الاحتفاظ الكامل:** يتم تخزين نص الرسالة في CDR للامتثال/الأرشفة
- **وضع الخصوصية:** يتم تعيين نص الرسالة إلى NULL عندما تكون `delete_message_body_after_delivery: true`
- **وضع الامتثال:** يتم تخزين النص مشفرًا أو مُجزأً (يتطلب تنفيذًا مخصصًا)

الطوابع الزمنية للتدقيق

الحقل	النوع	قابل للإلغاء	الوصف
`inserted_at`	DATETIME	لا	متى تم إدخال سجل CDR لأول مرة في قاعدة البيانات. عادةً ما يكون نفس الوقت أو بعده بقليل من delivery_time/expiry_time.
`updated_at`	DATETIME	لا	متى تم تحديث سجل CDR آخر مرة. نفس وقت inserted_at إذا لم يتم تحديثه أبدًا.

أمثلة SQL

الاستعلامات الأساسية

البحث عن جميع CDRs لرقم هاتف محدد:

```sql
SELECT * FROM cdrs
WHERE calling_number = '+15551234567'
   OR called_number = '+15551234567'
ORDER BY submission_time DESC
LIMIT 100;
```

عد الرسائل حسب الحالة:

```sql
SELECT status, COUNT(*) as count
FROM cdrs
GROUP BY status;
```

متوسط وقت التسليم للرسائل التي تم تسليمها:

```sql
SELECT AVG(TIMESTAMPDIFF(SECOND, submission_time, delivery_time))
AS avg_delivery_seconds
FROM cdrs
WHERE status = 'delivered'
  AND delivery_time IS NOT NULL;
```

استعلامات الفوترة

حجم الرسائل اليومية حسب SMSC الوجهة:

```sql
SELECT
  DATE(submission_time) AS date,
  dest_smsc,
  COUNT(*) AS message_count,
  SUM(CASE WHEN status = 'delivered' THEN 1 ELSE 0 END) AS
delivered_count,
  SUM(message_parts) AS total_segments
FROM cdrs
WHERE submission_time >= DATE_SUB(NOW(), INTERVAL 30 DAY)
GROUP BY DATE(submission_time), dest_smsc
ORDER BY date DESC, message_count DESC;
```

الرسائل القابلة للفوترة لعميل (حسب بادئة رقم الاتصال):

```sql
SELECT
  DATE(submission_time) AS date,
  COUNT(*) AS message_count,
  SUM(message_parts) AS total_segments,
  SUM(message_parts) * 0.01 AS total_cost
FROM cdrs
WHERE calling_number LIKE '+1555%'
  AND status = 'delivered'
  AND submission_time >= '2025-10-01'
  AND submission_time < '2025-11-01'
GROUP BY DATE(submission_time);
```

تحليل أداء التوجيه:

```sql
SELECT
  dest_smsc,
  COUNT(*) AS total_messages,
  SUM(CASE WHEN status = 'delivered' THEN 1 ELSE 0 END) AS
delivered,
  ROUND(100.0 * SUM(CASE WHEN status = 'delivered' THEN 1 ELSE 0
END) / COUNT(*), 2) AS delivery_rate_pct,
  AVG(delivery_attempts) AS avg_attempts,
  AVG(TIMESTAMPDIFF(SECOND, submission_time, delivery_time)) AS
avg_delivery_seconds
FROM cdrs
WHERE submission_time >= DATE_SUB(NOW(), INTERVAL 7 DAY)
  AND dest_smsc IS NOT NULL
GROUP BY dest_smsc
ORDER BY delivery_rate_pct DESC;
```

استعلامات التحليلات

الرسائل حسب ساعة اليوم (نمط الحركة):

```sql
SELECT
  HOUR(submission_time) AS hour,
  COUNT(*) AS message_count
FROM cdrs
WHERE submission_time >= DATE_SUB(NOW(), INTERVAL 7 DAY)
GROUP BY HOUR(submission_time)
ORDER BY hour;
```

تحليل الرسائل متعددة الأجزاء:

```sql
SELECT
  message_parts,
  COUNT(*) AS message_count,
  AVG(TIMESTAMPDIFF(SECOND, submission_time, delivery_time)) AS
avg_delivery_seconds
FROM cdrs
WHERE message_parts IS NOT NULL
  AND status = 'delivered'
GROUP BY message_parts
ORDER BY message_parts;
```

تحليل الرسائل الفاشلة:

```sql
SELECT
  called_number,
  COUNT(*) AS failure_count,
  AVG(delivery_attempts) AS avg_attempts,
  MAX(submission_time) AS last_failure
FROM cdrs
WHERE status IN ('failed', 'expired')
  AND submission_time >= DATE_SUB(NOW(), INTERVAL 7 DAY)
GROUP BY called_number
HAVING failure_count >= 5
ORDER BY failure_count DESC;
```

استعلامات الامتثال والتدقيق

البحث عن جميع الرسائل بين طرفين في نطاق زمني:

```sql
SELECT
  submission_time,
  calling_number,
  called_number,
  status,
  message_body,
  delivery_time
FROM cdrs
WHERE (
    (calling_number = '+15551234567' AND called_number =
'+15559876543')
    OR
    (calling_number = '+15559876543' AND called_number =
'+15551234567')
  )
  AND submission_time >= '2025-10-01'
  AND submission_time < '2025-11-01'
ORDER BY submission_time;
```

تنفيذ سياسة الاحتفاظ (حذف CDRs القديمة):

```sql
-- البحث عن السجلات الأقدم من فترة الاحتفاظ (مثال: سنتين)
SELECT COUNT(*) FROM cdrs
WHERE submission_time < DATE_SUB(NOW(), INTERVAL 2 YEAR);

-- حذف السجلات القديمة (استخدم بحذر!)
DELETE FROM cdrs
WHERE submission_time < DATE_SUB(NOW(), INTERVAL 2 YEAR)
LIMIT 10000;  -- حذف دفعي لتجنب القفل
```

تحليل الكتلة

توزيع الرسائل عبر العقد:

```sql
SELECT
  origin_node,
  COUNT(*) AS message_count,
  SUM(CASE WHEN status = 'delivered' THEN 1 ELSE 0 END) AS
delivered_count
FROM cdrs
WHERE submission_time >= DATE_SUB(NOW(), INTERVAL 1 DAY)
GROUP BY origin_node;
```

الفهارس

تم إنشاء الفهارس التالية لتحسين الاستعلامات الشائعة:

اسم الفهرس	الأعمدة	الغرض
`PRIMARY`	`id`	المفتاح الأساسي، يضمن سجل فريد
`idx_cdrs_message_id`	`message_id`	البحث عن CDR بواسطة معرف الرسالة الأصلية
`idx_cdrs_calling_number`	`calling_number`	العثور على الرسائل من مرسل محدد
`idx_cdrs_called_number`	`called_number`	العثور على الرسائل إلى مستلم محدد
`idx_cdrs_status`	`status`	تصفية حسب حالة التسليم
`idx_cdrs_submission_time`	`submission_time`	استعلامات قائمة على الوقت، فترات الفوترة
`idx_cdrs_dest_smsc`	`dest_smsc`	تحليل أداء التوجيه

توصيات فهرس إضافية

للنشر عالي الحجم، ضع في اعتبارك هذه الفهارس الإضافية:

فهرس مركب لاستعلامات الفوترة:

```
CREATE INDEX idx_cdrs_billing ON cdrs(calling_number,
submission_time, status);
```

فهرس مركب لتحليل التوجيه:

```
CREATE INDEX idx_cdrs_route_perf ON cdrs(dest_smsc,
submission_time, status);
```

فهرس مركب لعمليات البحث عن الامتثال:

```
CREATE INDEX idx_cdrs_party_time ON cdrs(calling_number,
called_number, submission_time);
```

فهرس نصي كامل لعمليات البحث عن نص الرسالة (MySQL):

```
ALTER TABLE cdrs ADD FULLTEXT INDEX idx_cdrs_message_body_ft
(message_body);

-- الاستخدام:
SELECT * FROM cdrs
WHERE MATCH(message_body) AGAINST('keyword' IN NATURAL LANGUAGE
MODE);
```

أنواع البيانات حسب قاعدة البيانات

تعيينات نوع الحقل عبر قواعد البيانات المدعومة:

الحقل	MySQL/MariaDB	PostgreSQL	الملاحظات
`id`	BIGINT AUTO_INCREMENT	BIGSERIAL	عدد صحيح 64 بت، يتزايد تلقائيًا
`message_id`	BIGINT	BIGINT	عدد صحيح 64 بت
حقول السلسلة	VARCHAR(255)	VARCHAR(255)	سلسلة ذات طول متغير، بحد أقصى 255 حرفًا
`message_body`	TEXT	TEXT	نص كبير، حتى 65,535 بايت (MySQL)، غير محدود (PostgreSQL)
الطوابع الزمنية	DATETIME	TIMESTAMP	يوصى باستخدام الطوابع الزمنية بتوقيت UTC
الأعداد الصحيحة	INT	INTEGER	عدد صحيح موقّع 32 بت
القيم المنطقية	BOOLEAN (TINYINT(1))	BOOLEAN	يخزن MySQL كـ 0/1

اعتبارات الخصوصية

قد تحتوي جدول CDR على معلومات شخصية حساسة (أرقام الهواتف، محتوى الرسائل). ضع في اعتبارك تدابير الخصوصية التالية:

1. خصوصية نص الرسالة

خيارات التكوين في `config/runtime.exs`:

```
config :sms_c,
  # حذف نص الرسالة بعد التسليم الناجح
  delete_message_body_after_delivery: true,

  # إخفاء نص الرسالة في واجهة المستخدم
  hide_message_body_in_ui: true,

  # إخفاء نص الرسالة في الصادرات
  hide_message_body_in_export: true
```

2. إخفاء أرقام الهواتف

لتحليلات لا تتطلب الأرقام الكاملة:

```
-- إخفاء آخر 4 أرقام من أرقام الهواتف
SELECT
  CONCAT(SUBSTRING(calling_number, 1, LENGTH(calling_number) - 4),
'XXXX') AS masked_calling,
  CONCAT(SUBSTRING(called_number, 1, LENGTH(called_number) - 4),
'XXXX') AS masked_called,
  COUNT(*) AS message_count
FROM cdrs
GROUP BY masked_calling, masked_called;
```

3. تشفير قاعدة البيانات

تفعيل التشفير أثناء الراحة لخادم قاعدة البيانات:

MySQL:

```
-- تفعيل تشفير الجدول
ALTER TABLE cdrs ENCRYPTION='Y';
```

PostgreSQL: استخدم تشفير البيانات الشفاف (TDE) في PostgreSQL أو تشفير مستوى النظام الملفات.

4. ضوابط الوصول

تقييد الوصول إلى جدول CDR:

```
-- إنشاء مستخدم فوترة للقراءة فقط
CREATE USER 'billing_ro'@'%' IDENTIFIED BY 'secure_password';
GRANT SELECT ON sms_c.cdrs TO 'billing_ro'@'%';

-- إنشاء مستخدم تحليلات محدود (لا يمكنه الوصول إلى نص الرسالة)
CREATE USER 'analytics'@'%' IDENTIFIED BY 'secure_password';
GRANT SELECT (id, message_id, calling_number, called_number,
source_smsc,
              dest_smsc, submission_time, delivery_time, status,
              delivery_attempts, message_parts)
ON sms_c.cdrs TO 'analytics'@'%';
```

سياسة الاحتفاظ والأرشفة

سياسات الاحتفاظ

حدد فترات الاحتفاظ بناءً على المتطلبات التنظيمية والتجارية:

الأساس التنظيمي	فترة الاحتفاظ النموذجية	الصناعة
قوانين الولايات، FCC	18-24 شهرًا	الاتصالات (الولايات المتحدة)
GDPR، ePrivacy	أشهر - 2 سنوات 6	الاتصالات (الاتحاد الأوروبي)
SOX، SEC	5-7 سنوات	المالية
HIPAA	6 سنوات	الرعاية الصحية

استراتيجية الأرشفة

1. تقسيم حسب التاريخ (MySQL 8.0+، PostgreSQL 11+)

```sql
-- تقسيم MySQL حسب الشهر
ALTER TABLE cdrs PARTITION BY RANGE (TO_DAYS(submission_time)) (
  PARTITION p202510 VALUES LESS THAN (TO_DAYS('2025-11-01')),
  PARTITION p202511 VALUES LESS THAN (TO_DAYS('2025-12-01')),
  PARTITION p202512 VALUES LESS THAN (TO_DAYS('2026-01-01')),
  PARTITION p_future VALUES LESS THAN MAXVALUE
);

-- حذف الجزء القديم (أرشفة سريعة)
ALTER TABLE cdrs DROP PARTITION p202510;
```

2. الأرشفة إلى التخزين البارد

```sql
-- تصدير CDRs القديمة إلى جدول الأرشيف
CREATE TABLE cdrs_archive LIKE cdrs;

INSERT INTO cdrs_archive
SELECT * FROM cdrs
WHERE submission_time < DATE_SUB(NOW(), INTERVAL 2 YEAR);

-- التحقق وحذف من الجدول الرئيسي
DELETE FROM cdrs
WHERE submission_time < DATE_SUB(NOW(), INTERVAL 2 YEAR);
```

3. نص برمجي لتنظيف تلقائي

```
#!/bin/bash
# cleanup_old_cdrs.sh - تشغيل عبر cron

MYSQL_USER="cleanup_user"
MYSQL_PASS="secure_password"
MYSQL_DB="sms_c"
RETENTION_DAYS=730  # سنتين

# أرشفة السجلات القديمة
mysql -u"$MYSQL_USER" -p"$MYSQL_PASS" "$MYSQL_DB" <<EOF
INSERT INTO cdrs_archive
SELECT * FROM cdrs
WHERE submission_time < DATE_SUB(NOW(), INTERVAL $RETENTION_DAYS
DAY)
LIMIT 100000;

DELETE FROM cdrs
WHERE submission_time < DATE_SUB(NOW(), INTERVAL $RETENTION_DAYS
DAY)
LIMIT 100000;
EOF
```

إدخال Cron:

```
# تشغيل يوميًا في الساعة 2 صباحًا
0 2 * * * /usr/local/bin/cleanup_old_cdrs.sh >>
/var/log/sms_c/cleanup.log 2>&1
```

تكامل الفوترة

مخطط بطاقة الأسعار

إنشاء جدول أسعار منفصل للفوترة:

```sql
CREATE TABLE billing_rates (
  id INT AUTO_INCREMENT PRIMARY KEY,
  destination_prefix VARCHAR(20) NOT NULL,
  description VARCHAR(255),
  rate_per_message DECIMAL(10, 6) NOT NULL,
  rate_per_segment DECIMAL(10, 6) NOT NULL,
  currency VARCHAR(3) DEFAULT 'USD',
  effective_date DATE NOT NULL,
  expiry_date DATE,
  INDEX idx_prefix (destination_prefix),
  INDEX idx_dates (effective_date, expiry_date)
);

-- أسعار مثال
INSERT INTO billing_rates (destination_prefix, description,
rate_per_message, rate_per_segment, effective_date) VALUES
('+1', '01-01-2025' ,0.0050 ,0.0050 ,'الولايات المتحدة/كندا'),
('+44', '01-01-2025' ,0.0080 ,0.0080 ,'المملكة المتحدة'),
('+61', '01-01-2025' ,0.0100 ,0.0100 ,'أستراليا'),
('+', '01-01-2025' ,0.0150 ,0.0150 ,'افتراضي دولي');
```

استعلام الفوترة

انضم إلى CDRs مع الأسعار للفوترة:

```
SELECT
  DATE(c.submission_time) AS date,
  c.dest_smsc AS route,
  LEFT(c.called_number,
    CASE
      WHEN c.called_number LIKE '+1%' THEN 2
      WHEN c.called_number LIKE '+%' THEN
LENGTH(SUBSTRING_INDEX(c.called_number, '', 4))
      ELSE 0
    END
  ) AS destination_prefix,
  COUNT(*) AS message_count,
  SUM(c.message_parts) AS segment_count,
  COALESCE(r.rate_per_segment, 0.015) AS rate,
  SUM(c.message_parts) * COALESCE(r.rate_per_segment, 0.015) AS
total_cost
FROM cdrs c
LEFT JOIN billing_rates r ON c.called_number LIKE
CONCAT(r.destination_prefix, '%')
  AND c.submission_time >= r.effective_date
  AND (r.expiry_date IS NULL OR c.submission_time < r.expiry_date)
WHERE c.status = 'delivered'
  AND c.submission_time >= '2025-10-01'
  AND c.submission_time < '2025-11-01'
GROUP BY date, route, destination_prefix
ORDER BY date DESC, total_cost DESC;
```

تصدير أنظمة الفوترة

تصدير CSV:

```
mysql -u billing_ro -p -D sms_c -e "
SELECT
  id,
  message_id,
  calling_number,
  called_number,
  dest_smsc,
  submission_time,
  delivery_time,
  status,
  message_parts
FROM cdrs
WHERE submission_time >= '2025-10-01'
  AND submission_time < '2025-11-01'
  AND status = 'delivered'
" --batch --silent | sed 's/\t/,/g' > billing_export_202510.csv
```

انظر أيضًا

- دليل التكوين - تكوين إعدادات تصدير CDR
- دليل العمليات - إجراءات صيانة CDR
- مرجع API - استعلام CDRs عبر واجهة REST API

مرجع تكوين SMS-C

العودة إلى فهرس الوثائق | **الملف التعريفي الرئيسي ←**

مرجع كامل لجميع خيارات تكوين SMS-C مع أمثلة لسيناريوهات النشر الشائعة.

جدول المحتويات

- ملفات التكوين
- تكوين قاعدة البيانات
- تكوين API
- تكوين واجهة الويب
- تكوين الاتحاد
- تكوين قائمة الانتظار للرسائل
- تكوين الشحن
- تكوين Diameter Sh (HSS)
- تكوين ENUM
- تكوين ترجمة الأرقام
- تكوين التوجيه
- تكوين تحسين الأداء
- تكوين السجلات
- سيناريوهات التكوين الشائعة

ملفات التكوين

يستخدم SMS-C ثلاث ملفات تكوين رئيسية:

config/config.exs

تكوين ثابت يتم تحميله في وقت الترجمة. يحتوي على:

- القيم الافتراضية على مستوى التطبيق
- تكوين السجل

- إعدادات التطوير/الاختبار
- معلمات تحسين الأداء

config/runtime.exs

تكوين وقت التشغيل يتم تحميله عند بدء التشغيل. يحتوي على:

- إعدادات اتصال قاعدة البيانات
- تكوين الكتلة
- تكامل الخدمة الخارجية (OCS، ENUM)
- المسارات الأولية وقواعد الترجمة
- إعدادات خاصة بالبيئة

config/prod.exs (اختياري)

تجاوزات خاصة بالإنتاج.

أفضل ممارسة: استخدم متغيرات البيئة في `runtime.exs` للقيم الحساسة مثل كلمات المرور ومفاتيح API.

تكوين تخزين SQL CDR

يستخدم SMS-C **Mnesia** للبيانات التشغيلية (قائمة الانتظار للرسائل، قواعد التوجيه، ترجمات الأرقام) ويدعم قواعد بيانات **SQL** الخارجية للتخزين طويل الأمد لسجلات تفاصيل المكالمات (CDR)، والتحليلات، والفوترة.

قواعد بيانات SQL المدعومة

يدعم النظام قواعد بيانات SQL التالية لتصدير CDR:

قاعدة البيانات	الإصدار	المحول	ال��نفذ الافتراضي	الأفضل لـ
MySQL	8.0+	`Ecto.Adapters.MyXQL`	3306	غرض عام، موثوقية مثبتة
MariaDB	10.5+	`Ecto.Adapters.MyXQL`	3306	متوافق مع MySQL، مفتوح المصدر
PostgreSQL	13+	`Ecto.Adapters.Postgres`	5432	ميزات متقدمة، دعم JSON

ملاحظة: يتم استخدام Mnesia تلقائيًا للبيانات التشغيلية (قائمة الانتظار للرسائل، التوجيه، والتخزين CDR **فقط** لتصدير SQL الترجمات) ولا تتطلب أي تكوين. يتم استخدام قاعدة البيانات طويل الأمد.

تكوين MySQL / MariaDB

```
# config/runtime.exs
config :sms_c, SmsC.Repo,
  adapter: Ecto.Adapters.MyXQL,
  username: System.get_env("DB_USERNAME") || "sms_user",
  password: System.get_env("DB_PASSWORD") || "secure_password",
  hostname: System.get_env("DB_HOSTNAME") || "localhost",
  port: String.to_integer(System.get_env("DB_PORT") || "3306"),
  database: System.get_env("DB_NAME") || "sms_c_prod",
  pool_size: String.to_integer(System.get_env("DB_POOL_SIZE") ||
"20")
```

تكوين PostgreSQL

```
# config/runtime.exs
config :sms_c, SmsC.Repo,
  adapter: Ecto.Adapters.Postgres,
  username: System.get_env("DB_USERNAME") || "sms_user",
  password: System.get_env("DB_PASSWORD") || "secure_password",
  hostname: System.get_env("DB_HOSTNAME") || "localhost",
  port: String.to_integer(System.get_env("DB_PORT") || "5432"),
  database: System.get_env("DB_NAME") || "sms_c_prod",
  pool_size: String.to_integer(System.get_env("DB_POOL_SIZE") ||
"20")
```

اختيار قاعدة بيانات SQL

MySQL/MariaDB - موصى به لمعظم عمليات النشر:

- أداء ممتاز لكتابات CDR
- موثوقية مثبتة في بيئات الاتصالات
- دعم واسع للأدوات لأنظمة الفوترة
- إعداد نسخ متماثل سهل

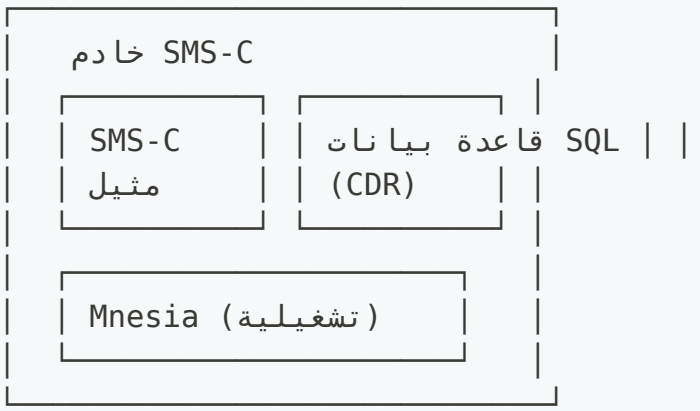
PostgreSQL - اعتبره إذا كنت بحاجة إلى:

- ميزات JSON/JSONB المتقدمة للتحليلات
- استعلامات معقدة على بيانات CDR
- بنية تحتية موجودة لـ PostgreSQL
- PostGIS للتحليل الجغرافي

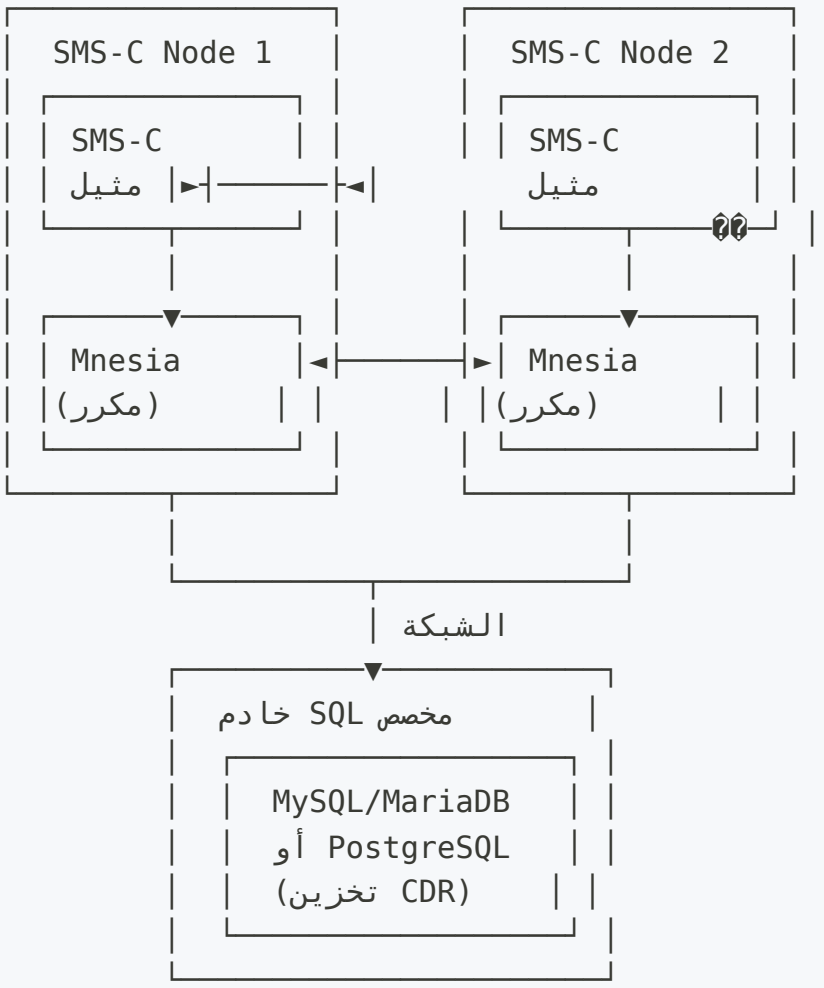
طوبولوجيات النشر

مهم: يمكن أن تعمل قاعدة بيانات SQL CDR عن مثيلات SMS-C الخاصة على **خادم منفصل** بك. هذه هي الطريقة الموصى بها لعمليات النشر في الإنتاج.

نشر خاد�ة واحد (تطوير/اختبار):



نشر موزع (إنتاج - موصى به):



فوائد خادم SQL منفصل:

- **عزل الأداء**: لا تؤثر كتابات CDR على معالجة الرسائل

- **قابلية التوسع**: توسيع قاعدة البيانات ومعالجة الرسائل بشكل مستقل
- **موثوقية** SMS-C: صيانة قاعدة البيانات لا تؤثر على وقت تشغيل
- **إدارة البيانات**: تخزين CDR مركزي لعدة مثيلات SMS-C
- **مرونة النسخ الاحتياطي**: جداول نسخ احتياطي مستقلة وسياسات الاحتفاظ

إرشادات حجم المجموعة

الوصف	حجم المجموعة	عبء العمل
تزامن ضئيل	5-10	تطوير
عمليات نشر صغيرة	10-15	حجم منخفض (< 100 رسالة/ثانية)
إنتاج نموذجي	20-30	حجم متوسط (100-1000 رسالة/ثانية)
سيناريوهات عالية الإنتاجية	40-100	حجم مرتفع (> 1000 رسالة/ثانية)

الحساب: `pool_size = (عدد العمليات المتزامنة المتوقعة في قاعدة البيانات) * 1.5`

أمثلة على اتصال قاعدة البيانات

استخدام متغيرات البيئة (موصى به للإنتاج):

```
# تعيين متغيرات البيئة
export DB_USERNAME=sms_prod_user
export DB_PASSWORD=strong_password_here
export DB_HOSTNAME=db-primary.internal.example.com
export DB_PORT=3306
export DB_NAME=sms_c_production
export DB_POOL_SIZE=30
```

تكوين مباشر (للتطوير فقط):

```
config :sms_c, SmsC.Repo,
  username: "dev_user",
  password: "dev_password",
  hostname: "localhost",
  database: "sms_c_dev",
  pool_size: 5
```

مراقبة مجموعة الاتصال

راقب استخدام المجموعة عبر مقاييس Prometheus:

- `ecto_pools_queue_time` - الوقت المستغرق في انتظار الاتصال
- `ecto_pools_query_time` - وقت تنفيذ الاستعلام
- `ecto_pools_connected_count` - الاتصالات النشطة

قم بتنبيه إذا تجاوز وقت الانتظار 100 مللي ثانية باستمرار - مما يشير إلى الحاجة إلى مجموعة أكبر.

تكوين API

يوفر واجهة برمجة التطبيقات REST قدرات تقديم وإدارة الرسائل.

تكوين API الأساسي

```
# config/runtime.exs
config :api_ex,
  port: String.to_integer(System.get_env("API_PORT") || "8443"),
  listen_ip: System.get_env("API_LISTEN_IP") || "0.0.0.0",
  enable_tls: System.get_env("API_ENABLE_TLS") != "false"
```

تكوين TLS/SSL

إعداد الإنتاج مع TLS (موصى به):

```
config :api_ex,
  port: 8443,
  listen_ip: "0.0.0.0",
  enable_tls: true,
  tls_cert_path: "/etc/sms_c/certs/server.crt",
  tls_key_path: "/etc/sms_c/certs/server.key"
```

إعداد التطوير بدون TLS:

```
config :api_ex,
  port: 8080,
  listen_ip: "127.0.0.1",
  enable_tls: false
```

إعداد شهادة API

قم بإنشاء شهادة موقعة ذاتيًا للاختبار:

```
# إنشاء دليل الشهادة
mkdir -p priv/cert

# توليد مفتاح خاص
openssl genrsa -out priv/cert/server.key 2048

# توليد طلب توقيع الشهادة
openssl req -new -key priv/cert/server.key -out
priv/cert/server.csr \
  -subj "/C=US/ST=State/L=City/O=Organization/CN=sms-
api.example.com"

# توليد شهادة موقعة ذاتيًا (صالحة لمدة 365 يومًا)
openssl x509 -req -days 365 -in priv/cert/server.csr \
  -signkey priv/cert/server.key -out priv/cert/server.crt

# تعيين الأذونات
chmod 600 priv/cert/server.key
chmod 644 priv/cert/server.crt
```

لإنتاج، استخدم الشهادات من CA موثوق (Let's Encrypt، CA تجارية، إلخ).

التحكم في وصول API

قائمة بيضاء IP (جدار حماية التطبيق):

```
# باستخدام iptables (Linux)
iptables -A INPUT -p tcp --dport 8443 -s 10.0.0.0/8 -j ACCEPT
iptables -A INPUT -p tcp --dport 8443 -j DROP

# باستخدام firewalld (Red Hat/CentOS)
firewall-cmd --permanent --add-rich-rule='rule family="ipv4"
source address="10.0.0.0/8" port protocol="tcp" port="8443"
accept'
firewall-cmd --reload
```

مصادقة مفتاح API (على مستوى التطبيق):

قم بتكوين عبر مكون إضافي مخصص في الموجه - راجع دليل العمليات لتفاصيل التنفيذ.

تكوين واجهة الويب

توفر الواجهة الويب إدارة المسارات، وتصفح الرسائل، والمراقبة.

تكوين واجهة الويب الأساسية

```
# config/runtime.exs
config :control_panel,
  port: String.to_integer(System.get_env("WEB_PORT") || "80"),
  hostname: System.get_env("WEB_HOSTNAME") || "localhost",
  enable_tls: System.get_env("WEB_ENABLE_TLS") == "true"
```

إعداد واجهة الويب للإنتاج

```
config :control_panel,
  port: 443,
  hostname: "sms-admin.example.com",
  enable_tls: true,
  tls_cert_path: "/etc/sms_c/certs/web.crt",
  tls_key_path: "/etc/sms_c/certs/web.key"
```

إعداد الوكيل العكسي (موصى به)

استخدم Nginx أو Apache كوكيل عكسي لمزيد من الأمان والميزات:

مثال تكوين Nginx:

```
upstream sms_web {
  server 127.0.0.1:4000;
  keepalive 32;
}

server {
  listen 80;
  server_name sms-admin.example.com;
  return 301 https://$server_name$request_uri;
}

server {
  listen 443 ssl http2;
  server_name sms-admin.example.com;

  ssl_certificate /etc/letsencrypt/live/sms-
admin.example.com/fullchain.pem;
  ssl_certificate_key /etc/letsencrypt/live/sms-
admin.example.com/privkey.pem;
  ssl_protocols TLSv1.2 TLSv1.3;
  ssl_ciphers HIGH:!aNULL:!MD5;

  # مصادقة أساسية لمزيد من الأمان
  auth_basic "SMS-C Admin";
  auth_basic_user_file /etc/nginx/.htpasswd;

  location / {
    proxy_pass http://sms_web;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
  }

  # دعم WebSocket لـــ LiveView
  location /live {
    proxy_pass http://sms_web;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
```

```
    proxy_read_timeout 86400;
  }
}
```

تكوين الاتحاد

يستخدم OmniMessage الاتحاد القائم على HTTP للنشر متعدد وحدات التحكم. تكتشف الوحدات بعضها البعض عبر سجلات DNS SRV أو قوائم الأقران الثابتة، وتبادل الصحة وسجلات الواجهة الأمامية عبر HTTPS، وإعادة توجيه الرسائل إلى وحدات التحكم البعيدة عند الحاجة.

راجع **دليل الاتحاد الجغرافي** للحصول على الهندسة المعمارية الكاملة، وأمثلة النشر، واستكشاف الأخطاء وإصلاحها.

بدء سريع

```
# config/runtime.exs
config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "_smsc._tcp.smsc.example.com"
```

مرجع كامل للمعلمات

المعلمة	النوع	الافتراضي	الوصف
`enabled`	Boolean	`false`	المفتاح الرئيسي للاتحاد.
`dns_srv_domain`	String	`""`	مجال DNS SRV لاكتشاف الأقران.
`dns_poll_interval_ms`	Integer	`30000`	فترة إعادة حل DNS.
`health_check_interval_ms`	Integer	`10000`	فترة فحص صحة الأقران.
`registry_sync_interval_ms`	Integer	`15000`	فترة تبادل سجل الواجهة الأمامية.
`forward_retry_interval_ms`	Integer	`5000`	فترة إعادة محاولة الرسائل المعلقة.
`forward_max_retries`	Integer	`50`	الحد الأقصى لمحاولات التوجيه قبل وضع علامة على الفشل.
`http_timeout_ms`	Integer	`5000`	مهلة لجميع مكالمات HTTP للأقران.
`api_port`	Integer	`8443`	منفذ API لبناء عناوين URL للأقران.
`static_peers`	قائمة	`[]`	قائمة أقران ثابتة (بديل لسجل DNS SRV).

متطلبات الشبكة

بين المواقع، مقارنةً بالمنفذين 4369 + 9200-9100 (HTTPS) يتطلب الاتحاد فقط المنفذ **8443** لتجميع Erlang.

```
# السماح بحركة المرور الخاصة بالاتحاد من المواقع الشريكة
iptables -A INPUT -p tcp -s 10.0.1.0/24 --dport 8443 -j ACCEPT
iptables -A INPUT -p tcp -s 10.0.2.0/24 --dport 8443 -j ACCEPT
```

تكوين قائمة الانتظار للرسائل

يتحكم في سلوك الاحتفاظ بالرسائل وانتهاء صلاحيتها.

انتهاء صلاحية الرسائل

```
# config/runtime.exs
config :sms_c,
  dead_letter_time_minutes: 1440  # 24 ساعة
```

القيم الشائعة:

- **60** - 1 ساعة (اختبار/تطوير)
- **1440** - 24 ساعة (إنتاج نموذجي)
- **4320** - 3 أيام (احتفاظ ممتد)
- **10080** - 7 أيام (أقصى احتفاظ)

تصبح الرسائل التي تتجاوز هذه القيمة غير قابلة للتسليم ويتم وضع علامة عليها للتنظيف.

تكوين إعادة محاولة التسليم

يستخدم سلوك إعادة المحاولة التراجع الأسي:

```
تأخير إعادة المحاولة = 2^(عدد المحاولات) دقائق
```

المحاولة	التأخير
1	2 دقائق
2	4 دقائق
3	8 دقائق
4	16 دقيقة
5	32 دقيقة
6	64 دقيقة
7	128 دقيقة
8	256 دقيقة

الحد الأقصى للمحاولات قبل الرسالة الميتة: محدود بواسطة `dead_letter_time_minutes`.

تكوين التنظيف

```
# config/config.exs
config :sms_c,
  cleanup_interval_minutes: 10,
  fingerprint_ttl_minutes: 5,
  event_ttl_days: 7
```

فترات التنظيف:

- **cleanup_interval_minutes**: مدى تكرار تشغيل عامل التنظيف (الافتراضي: 10)
- **fingerprint_ttl_minutes**: نافذة اكتشاف التكرار (الافتراضي: 5)
- **event_ttl_days**: احتفاظ سجل الأحداث (الافتراضي: 7)

تكوين الشحن

التكامل مع OCS للشحن والفوترة عبر الإنترنت.

تمكين الشحن

```
# config/runtime.exs
config :sms_c,
  default_charging_enabled: true,
  ocs_url: "http://ocs.internal.example.com:2080/jsonrpc",
  ocs_tenant: "sms.example.com",
  ocs_destination: "default",
  ocs_source: "sms_platform",
  ocs_subject: "sms_user",
  ocs_account: "default_account"
```

تعطيل الشحن

```
# config/runtime.exs
config :sms_c,
  default_charging_enabled: false
```

عند التعطيل، تتم معالجة جميع الرسائل دون فحوصات الشحن.

تكوين الشحن لكل مستأجر

```
config :sms_c,
  ocs_url: System.get_env("OCS_URL") ||
"http://localhost:2080/jsonrpc",
  ocs_tenant: System.get_env("OCS_TENANT") ||
"tenant1.example.com",
  ocs_account: System.get_env("OCS_ACCOUNT") || "default"
```

متغيرات البيئة لكل مستأجر:

```
# 1 المستأجر
export OCS_TENANT=tenant1.example.com
export OCS_ACCOUNT=tenant1_account

# 2 المستأجر
export OCS_TENANT=tenant2.example.com
export OCS_ACCOUNT=tenant2_account
```

سلوك فشل الشحن

قم بتكوين ما يحدث عند فشل الشحن:

```
config :sms_c,
  charging_failure_action: :allow  # أو :deny
```

- **:allow** - معالجة الرسالة حتى إذا فشل الشحن (تسجيل الخطأ)
- **:deny** - رفض الرسالة إذا فشل الشحن

مثال على اتصال OCS

اختبار اتصال OCS:

```
# اختبار واجهة برمجة تطبيقات OCS
curl -X POST http://ocs.internal.example.com:2080/jsonrpc \
  -H "Content-Type: application/json" \
  -d '{
    "method": "SessionSv1.AuthorizeEvent",
    "params": [{
      "Tenant": "sms.example.com",
      "Account": "test_account",
      "Destination": "1234567890",
      "Usage": 100
    }],
    "id": 1
  }'
```

الاستجابة المتوقعة:

```
{
  "id": 1,
  "result": {
    "Attributes": {},
    "MaxUsage": 100,
    ...
  }
}
```


تكوين Diameter Sh (HSS)

يمكن لـ OmniMessage استعلام HSS عبر واجهة Diameter Sh لاكتشاف المشتركين على الشبكة الذين هم مؤقتًا غير متصلين، مما يمنع توجيه الرسائل إلى البوابة الافتراضية. راجع دليل البحث عن المشتركين في HSS للحصول على الوثائق التشغيلية الكاملة.

```
# تمكين HSS dip في مسار التوجيه
config :sms_c,
  diameter_enabled: true

# تكوين كومة Diameter
config :diameter_ex,
  diameter: %{
    service_name: :omnimessage,
    listen_ip: "0.0.0.0",
    listen_port: 3868,
    decode_format: :map,
    host: "smsc01",
    realm: "epc.mnc005.mcc547.3gppnetwork.org",
    product_name: "OmniMessage",
    request_timeout: 5000,
    control_module: SmsC.Diameter.Control,
    processor_module: DiameterEx.Processor,
    vendor_id: 10415,
    supported_vendor_ids: [10415],
    applications: [
      %{
        application_name: :sh,
        application_dictionary: :diameter_gen_3gpp_sh,
        vendor_specific_application_ids: [
          %{vendor_id: 10415, auth_application_id: 16_777_217,
acct_application_id: nil}
        ]
      }
    ],
    peers: [
      %{
        host: "dra01.epc.mnc005.mcc547.3gppnetwork.org",
        ip: "10.0.0.1",
        port: 3868,
        realm: "epc.mnc005.mcc547.3gppnetwork.org",
        tls: false,
        transport: :diameter_tcp,
        initiate_connection: true
      }
    ]
  }
```

المعلمة	النوع	مطلوب	الافتراضي	الوصف
`diameter_enabled`	Boolean	لا	`false`	تمكين أو تعطيل البحث عن المشتركين في HSS في مسار التوجيه
`mock_sh`	Boolean	لا	`false`	استخدام استجابات HSS وهمية للاختبار (لم يتم بدء كومة Diameter)
`mock_sh_on_net_numbers`	قائمة	لا	`[]`	MSISDNs التي تُعتبر على الشبكة في وضع المحاكاة

للحصول على مرجع كامل لمعلمات كومة Diameter (المضيف، النطاق، الأقران، التطبيقات)، راجع دليل البحث عن المشتركين في HSS.

تكوين ENUM

استعلامات رقم E.164 القائمة على DNS للتوجيه الذكي.

تعطيل ENUM (افتراضي)

```
# config/runtime.exs
config :sms_c,
  enum_enabled: false
```

تمكين ENUM مع DNS الافتراضي

```
config :sms_c,
  enum_enabled: true,
  enum_domains: ["e164.arpa", "e164.org"],
  enum_dns_servers: [],  # استخدم DNS الافتراضي للنظام
  enum_timeout: 5000  # 5 ثوانٍ
```

تمكين ENUM مع خوادم DNS مخصصة

```
config :sms_c,
  enum_enabled: true,
  enum_domains: ["e164.internal.example.com", "e164.arpa"],
  enum_dns_servers: [
    {"10.0.1.53", 53},  # خادم DNS داخلي
    {"8.8.8.8", 53},    # Google Public DNS (احتياطي)
    {"1.1.1.1", 53}     # Cloudflare DNS (احتياطي)
  ],
  enum_timeout: 3000  # 3 ثوانٍ (فشل أسرع)
```

أولوية مجال ENUM

يتم استعلام المجالات بالترتيب حتى يتم العثور على استعلام ناجح:

```
config :sms_c,
  enum_domains: [
    "e164.internal.example.com",  # حاول الداخلية أولاً
    "e164.carrier.net",           # ثم الناقل
    "e164.arpa"                   # ثم السجل العام
  ]
```

تحسين أداء ENUM

لشبكات ذات زمن انتقال منخفض:

```
enum_timeout: 2000  # 2 ثانية
```

لروابط ذات زمن انتقال مرتفع/فضائية:

```
enum_timeout: 10000  # 10 ثوانٍ
```

مثال على إعداد ENUM لـ DNS

تكوين منطقة ENUM خاصة (تنسيق BIND9):

```
; ملف المنطقة لـ e164.internal.example.com
$ORIGIN e164.internal.example.com.
$TTL 300

; الرقم: +1-555-0100 يصبح
0.0.1.0.5.5.5.1.e164.internal.example.com
0.0.1.0.5.5.5.1.e164.internal.example.com. IN NAPTR 100 10 "u"
"E2U+sip" "!^.*$!sip:15550100@voip-gateway.example.com!" .
0.0.1.0.5.5.5.1.e164.internal.example.com. IN NAPTR 100 20 "u"
"E2U+pstn" "!^.*$!pstn:gateway-a.example.com!" .

; الرقم: +1-555-0200
0.0.2.0.5.5.5.1.e164.internal.example.com. IN NAPTR 100 10 "u"
"E2U+sip" "!^.*$!sip:15550200@voip-gateway.example.com!" .
```

اختبار حل ENUM:

```
# استعلام مجال ENUM
dig @10.0.1.53 NAPTR 0.0.1.0.5.5.5.1.e164.internal.example.com

# المخرجات المتوقعة تشمل سجلات NAPTR:
# 0.0.1.0.5.5.5.1.e164.internal.example.com. 300 IN NAPTR 100 10
"u" "E2U+sip" "!^.*$!sip:15550100@voip-gateway.example.com!" .
```

تكوين ترجمة الأرقام

تطبيق تطبيع الأرقام المستند إلى Regex قبل التوجيه.

تعطيل ترجمة الأرقام

```
# config/runtime.exs
config :sms_c,
  translation_rules: []
```

أمثلة أساسية لترجمة الأرقام

إضافة رمز الدولة إلى الأرقام المحلية:

```
config :sms_c,
  translation_rules: [
    %{
      calling_prefix: nil,
      called_prefix: "",
      source_smsc: nil,
      calling_match: "^(\d{10})$",          # مطابقة الأرقام المكونة من 10 أرقام
      calling_replace: "+1\1",               # إضافة +1
      called_match: "^(\d{10})$",
      called_replace: "+1\1",
      priority: 100,
      description: "إضافة +1 إلى الأرقام المكونة من 10 أرقام في أمريكا الشمالية",
      enabled: true
    }
  ]
```

تطبيع التنسيق الدولي:

```
%{
  calling_prefix: nil,
  called_prefix: nil,
  source_smsc: nil,
  calling_match: "^00(\d+)$",                مطابقة بادئة 00 #
  calling_replace: "+\1",                    + الاستبدال بـ #
  called_match: "^00(\d+)$",
  called_replace: "+\1",
  priority: 10,
  description: "تحويل بادئة 00 الدولية إلى +",
  enabled: true
}
```

إزالة أحرف التنسيق:

```
%{
  calling_prefix: nil,
  called_prefix: nil,
  source_smsc: nil,
  calling_match: "^\+?1?[\s\-\.\(\)]*(\d{3})[\s\-\.\)\(]*(\d{3})
[\s\-\.\(\)]*(\d{4})$",
  calling_replace: "+1\1\2\3",
  called_match: "^\+?1?[\s\-\.\(\)]*(\d{3})[\s\-\.\)\(]*(\d{3})
[\s\-\.\(\)]*(\d{4})$",
  called_replace: "+1\1\2\3",
  priority: 50,
  description: "تطبيع تنسيق رقم الهاتف الأمريكي",
  enabled: true
}
```

ترجمة محددة للناقل

إزا��ة رمز التوجيه:

```
%{
  calling_prefix: nil,
  called_prefix: "101",                          # فقط لبادئة 101
  source_smsc: "carrier_a",                      # فقط من هذا الناقل
  calling_match: nil,                            # لا تغيير في الاتصال
  calling_replace: nil,
  called_match: "^101(\d+)$",                   # إزالة رمز التوجيه
101
  called_replace: "\1",
  priority: 5,
  description: "إزالة رمز التوجيه للناقل من الرقم المتصل به",
  enabled: true
}
```

ترجمة متعددة القواعد

يتم تقييم القواعد بترتيب الأولوية (رقم أقل = أولوية أعلى):

```
config :sms_c,
  translation_rules: [
    # الأولوية 1: القواعد الأكثر تحديدًا أولاً
    %{
      calling_prefix: "1555",
      called_prefix: nil,
      source_smsc: nil,
      calling_match: "^(1555\d{7})$",
      calling_replace: "+\1",
      called_match: nil,
      called_replace: nil,
      priority: 1,
      description: "تطبيع الرقم المتميز",
      enabled: true
    },

    # الأولوية 50: القواعد العامة
    %{
      calling_prefix: nil,
      called_prefix: nil,
      source_smsc: nil,
      calling_match: "^(\d{10})$",
      calling_replace: "+1\1",
      called_match: "^(\d{10})$",
      called_replace: "+1\1",
      priority: 50,
      description: "تطبيع عام للأرقام المكونة من 10 أرقام",
      enabled: true
    }
  ]
```

تكوين التوجيه

للحصول على SMS يتم تحميل قواعد التوجيه الأولية عند بدء التشغيل الأول. راجع دليل توجيه الوثائق الكاملة حول التوجيه.

تحميل المسارات من التكوين

```
# config/runtime.exs
config :sms_c,
  sms_routes: [
    # مثال على التوجيه الجغرافي
    %{
      calling_regex: nil,
      called_regex: ~r/^\+1/,
      source_smsc: nil,
      dest_smsc: "north_america_gateway",
      source_type: nil,
      enum_domain: nil,
      auto_reply: false,
      auto_reply_message: nil,
      drop: false,
      charged: :default,
      on_net_only: false,
      weight: 100,
      priority: 50,
      description: "توجيه أمريكا الشمالية",
      enabled: true
    },

    # مثال على التوجيه المتوازن
    %{
      calling_regex: nil,
      called_regex: ~r/^\+44/,
      source_smsc: nil,
      dest_smsc: "uk_gateway_1",
      source_type: nil,
      enum_domain: nil,
      auto_reply: false,
      auto_reply_message: nil,
      drop: false,
      charged: :default,
      on_net_only: false,
      weight: 70,
      priority: 50,
      description: "(%70) البوابة الرئيسية في المملكة المتحدة",
      enabled: true
    },
    %{
```

```
      calling_regex: nil,
      called_regex: ~r/^\+44/,
      source_smsc: nil,
      dest_smsc: "uk_gateway_2",
      source_type: nil,
      enum_domain: nil,
      auto_reply: false,
      auto_reply_message: nil,
      drop: false,
      charged: :default,
      on_net_only: false,
      weight: 30,
      priority: 50,
      description: "(%30) البوابة الاحتياطية في المملكة المتحدة",
      enabled: true
    },

    # تقييد الربط - فقط على الشبكة فقط SMPP إلى الوجهات على الشبكة فقط
    %{
      calling_regex: nil,
      called_regex: nil,
      source_smsc: "carrier_smpp_bind",
      dest_smsc: "local_msc",
      source_type: :smpp,
      enum_domain: nil,
      auto_reply: false,
      auto_reply_message: nil,
      drop: false,
      charged: :default,
      on_net_only: true,
      weight: 100,
      priority: 50,
      description: "إنهاء على الشبكة فقط - X الناقل",
      enabled: true
    }
  ]
```

تخطي تحميل المسارات الأولية

```
# لا تحميل المسارات من التكوين (إدارة عبر واجهة الويب فقط)
config :sms_c,
  sms_routes: []
```

يتم تحميل المسارات المعرفة في التكوين فقط إذا كانت جدول التوجيه فارغًا (عند بدء التشغيل الأول).

تكوين تحسين الأداء

راجع دليل تحسين الأداء لاستراتيجيات التحسين التفصيلية.

عامل إدراج الدفعات

```
# config/config.exs
config :sms_c,
  batch_insert_batch_size: 100,          # الرسائل لكل دفعة
  batch_insert_flush_interval_ms: 100    # الحد الأق��ى لوقت الانتظار بالمللي ثانية
```

ملفات تعريف الأداء:

الملف الشخصي	حجم الدفعة	الفترة	الإنتاجية	زمن الانتقال
حجم مرتفع	200	200 مللي ثانية	~5,000 رسالة/ثانية	حتى 200 مللي ثانية
متوازن	100	100 مللي ثانية	~4,500 رسالة/ثانية	حتى 100 مللي ثانية
زمن انتقال منخفض	50	20 مللي ثانية	~3,000 رسالة/ثانية	حتى 20 مللي ثانية
في الوقت الحقيقي	10	10 مللي ثانية	~1,500 رسالة/ثانية	حتى 10 مللي ثانية

تكوين السجلات

مستويات السجل

```
# config/config.exs
config :logger, :console,
  level: :info,  # :debug, :info, :warning, :error
  format: "$time $metadata[$level] $message\n",
  metadata: [:request_id, :message_id, :route_id]
```

موصى به للتطوير: `:debug` **موصى به للإنتاج**: `:info` أو `:warning`

وجهات إخراج السجل

وحدة التحكم فقط (تطوير):

```
config :logger,
  backends: [:console]
```

(إنتاج) **مسجل الملفات**:

```
config :logger,
  backends: [:console, {LoggerFileBackend, :file_log}]

config :logger, :file_log,
  path: "/var/log/sms_c/application.log",
  level: :info,
  format: "$time $metadata[$level] $message\n",
  metadata: [:request_id, :message_id]
```

تدوير السجلات

logrotate باستخدام (Linux):

```
# /etc/logrotate.d/sms_c
/var/log/sms_c/*.log {
    daily
    rotate 30
    compress
    delaycompress
    notifempty
    create 0644 sms_user sms_group
    sharedscripts
    postrotate
        # إشارة التطبيق لإعادة فتح ملف السجل
        systemctl reload sms_c
    endscript
}
```

سيناريوهات التكوين الشائعة

مجمع عالي الحجم

تحسين لأقصى إنتاجية (5,000+ رسالة/ثانية):

```
# قاعدة البيانات
config :sms_c, SmsC.Repo,
  pool_size: 50

# عامل الدفعة
config :sms_c,
  batch_insert_batch_size: 200,
  batch_insert_flush_interval_ms: 200

# احتفاظ الرسائل
config :sms_c,
  dead_letter_time_minutes: 1440  # 24 ساعة

# الشحن (معطل من أجل الأداء)
config :sms_c,
  default_charging_enabled: false

# التنظيف (فترات ممتدة)
config :sms_c,
  cleanup_interval_minutes: 30
```

خدمات الرسائل في الوقت الحقيقي للمؤسسات

تحسين لزمن انتقال منخفض (< 20 مللي ثانية):

```
# قاعدة البيانات
config :sms_c, SmsC.Repo,
  pool_size: 20

# عامل الدفعة (زمن انتقال منخفض)
config :sms_c,
  batch_insert_batch_size: 20,
  batch_insert_flush_interval_ms: 10

# احتفاظ الرسائل
config :sms_c,
  dead_letter_time_minutes: 4320  # 3 أيام

# الشحن (مفعل)
config :sms_c,
  default_charging_enabled: true,
  ocs_url: "http://ocs.local:2080/jsonrpc"
```

تطوير/اختبار

تحسين للتصحيح والرؤية:

```
# قاعدة البيانات
config :sms_c, SmsC.Repo,
  pool_size: 5

# عامل الدفعة (فوري)
config :sms_c,
  batch_insert_batch_size: 1,
  batch_insert_flush_interval_ms: 10

# السجلات (مفصلة)
config :logger, :console,
  level: :debug

# احتفاظ الرسائل (قصير)
config :sms_c,
  dead_letter_time_minutes: 60  # 1 ساعة

# الشحن (معطل)
config :sms_c,
  default_charging_enabled: false
```

مزود خدمة متعدد المستأجرين

تكوين منفصل لكل مستأجر:

```
# بيئة المستأجر 1
export DB_NAME=sms_c_tenant1
export OCS_TENANT=tenant1.example.com
export OCS_ACCOUNT=tenant1_account
export NODE_NAME=sms_tenant1@node1.example.com

# بيئة المستأجر 2
export DB_NAME=sms_c_tenant2
export OCS_TENANT=tenant2.example.com
export OCS_ACCOUNT=tenant2_account
export NODE_NAME=sms_tenant2@node1.example.com
```

تكرار جغرافي

تجميع عبر المناطق:

```
# مجموعة شرق الولايات المتحدة
config :sms_c,
  cluster_nodes: [
    :"sms@us-east-1a.example.com",
    :"sms@us-east-1b.example.com",
    :"sms@us-west-1a.example.com"  # عبر المنطقة للتعافي من
الكوارث
  ],
  smsc_node_name: "us-east-1a"
```

التحقق من التكوين

اختبر التكوين قبل النشر:

```
# تحقق من بناء جملة التكوين
mix compile

# تحقق من اتصال قاعدة البيانات
mix ecto.create
mix ecto.migrate

# اختبار اتصال OCS (إذا كان مفعلًا)
curl -X POST http://localhost:2080/jsonrpc -H "Content-Type:
application/json" \
  -d '{"method":"SessionSv1.Ping","params":[],"id":1}'

# بدء التطبيق في وضع تفاعلي
iex -S mix phx.server
```

مرجع متغيرات البيئة

متغيرات البيئة الشائعة المستخدمة في التكوين:

المتغير	الغرض	المثال
`DB_USERNAME`	اسم مستخدم قاعدة البيانات	`sms_prod_user`
`DB_PASSWORD`	كلمة مرور قاعدة البيانات	`strong_password`
`DB_HOSTNAME`	مضيف قاعدة البيانات	`db.internal.example.com`
`DB_PORT`	منفذ قاعدة البيانات	`3306`
`DB_NAME`	اسم قاعدة البيانات	`sms_c_production`
`DB_POOL_SIZE`	حجم مجموعة الاتصال	`30`
`API_PORT`	منفذ الاستماع لـ API	`8443`
`API_LISTEN_IP`	عنوان IP للاستماع لـ API	`0.0.0.0`
`WEB_PORT`	منفذ واجهة الويب	`443`
`NODE_NAME`	اسم عقدة Erlang	`sms@node1.example.com`
`ERLANG_COOKIE`	سر التجميع	`shared_cookie_value`
`OCS_URL`	عنوان واجهة برمجة تطبيقات OCS	`http://ocs.local:2080/jsonrpc`
`OCS_TENANT`	مستأجر OCS	`sms.example.com`

أفضل ممارسات التكوين

1. **استخدم متغيرات البيئة** للقيم الحساسة (كلمات المرور، مفاتيح API)
2. **اختبر تغييرات التكوين** في بيئة staging قبل الإنتاج
3. **وثق الإعدادات المخصصة** في ملاحظات النشر

4. **تحكم في ملفات ال��كوين** (باستثناء الأسرار)
5. **راقب بعد التغييرات** لاكتشاف تدهور الأداء
6. **احتفظ بنسخ احتياطية** من التكوينات العاملة
7. **تحقق قبل إعادة التشغيل** لتجنب فشل بدء التشغيل
8. **استخدم أسماء متسقة** عبر البيئات
9. **حدد حدود الموارد** المناسبة للأجهزة
10. **راجع بشكل دوري** لإزالة الميزات غير المستخدمة

استكشاف مشكلات التكوين

العرض	السبب المحتمل	الحل
التطبيق لا يبدأ	خطأ في بناء الجملة في التكوين	تحقق من السجلات، تحقق من بناء الجملة
فشل اتصال قاعدة البيانات	بيانات اعتماد/مضيف خاطئ	تحقق من متغيرات البيئة DB_*
غير متاحة API	ربط خاطئ بالمنفذ/IP	تحقق من listen_ip و API_PORT
لا تتصل عقد التجميع	عدم تطابق في الكوكي، جدار ناري	تحقق من ERLANG_COOKIE، تحقق من المنافذ 4369، 9100-9200
فشل الشحن	غير متاح OCS	اختبار الاتصال بـ ocs_url
فشل استعلامات ENUM	غير متاح DNS خادم	اختبار الاتصال بـ DNS، تحقق من المهلة
أداء ضعيف	إعدادات دفعة خاطئة	مراجعة دليل تحسين الأداء
الرسائل لا تتجه	المسارات لم يتم تحميلها	تحقق من تكوين sms_routes أو واجهة الويب

للحصول على مساعدة إضافية، راجع دليل استكشاف الأخطاء وإصلاحها.

تكوين تخزين الرسائل (Mnesia)

احتفاظ الرسائل

يتم تخزين الرسائل في Mnesia للتكوين قابل تلقائي تنظيف مع السريع للوصول.

```
config :sms_c,
  # المدة التي يجب الاحتفاظ بالرسائل في Mnesia (ساعات)
  message_retention_hours: 24,

  # مدى تكرار التحقق من الرسائل القديمة (دقائق)
  retention_check_interval_minutes: 60
```

التوصيات:

- **الإنتاج:** 24-72 ساعة (توازن بين الاحتياجات التشغيلية مقابل الذاكرة)
- **التطوير:** 4-8 ساعات (تنظيف أسرع للاختبار)
- **حجم مرتفع:** 12-24 ساعة (الحفاظ على الذاكرة)

تأثير الذاكرة:

- متوسط الرسالة: ~1KB
- 10,000 رسالة: ~10MB
- 100,000 رسالة: ~100MB

تصدير CDR (سجل تفاصيل المكالمات)

الخاصة بك Ecto تلقائيًا إلى قاعدة بيانات CDRs عند تسليم الرسائل أو انتهاء صلاحيتها، يمكن كتابة للتخزين طويل الأمد وتحليلات الفوترة.

```
config :sms_c,
  # تمكين/تعطيل كتابة CDR
  cdr_enabled: true
```

تتضمن سجلات CDR:

- معرف الرسالة، الأرقام المتصلة/المتصل بها

- مصدر/وجهة SMSC
- العقدة الأصلية/الوجهة (للتجميعات)
- طوابع زمنية للتقديم، التسليم، انتهاء الصلاحية
- الحالة، محاولات التسليم
- جسم الرسالة الاختياري (انظر ضوابط الخصوصية)

متى يتم التعطيل:

- بيئات الاختبار حيث لا حاجة لـ CDRs
- استكشاف الأخطاء مؤقتًا لتقليل الحمل على قاعدة البيانات

ضوابط الخصوصية

تكوين رؤية جسم الرسالة واحتفاظها للامتثال للخصوصية.

```
config :sms_c,
  # حذف جسم الرسالة من Mnesia بعد التسليم الناجح
  delete_message_body_after_delivery: false,

  # إخفاء جسم الرسالة في واجهة الويب
  hide_message_body_in_ui: false,

  # إخفاء جسم الرسالة في تصديرات CSV
  hide_message_body_in_export: false
```

حالات الاستخدام:

حالة الاستخدام	التكوين
حفظ مساحة Mnesia، الامتثال للخصوصية	`delete_message_body_after_delivery: true`
منع المشغل من رؤية محتوى الرسالة	`hide_message_body_in_ui: true`
الامتثال لتصدير البيانات، تقارير معقمة	`hide_message_body_in_export: true`

تكوينات المثال:

أقصى خصوصية (امتثال)

```
config :sms_c,
  delete_message_body_after_delivery: true,
  hide_message_body_in_ui: true,
  hide_message_body_in_export: true,
  cdr_enabled: true  # الاحتفاظ بـ CDRs بدون أجسام
```

التطوير (رؤية كاملة)

```
config :sms_c,
  delete_message_body_after_delivery: false,
  hide_message_body_in_ui: false,
  hide_message_body_in_export: false,
  cdr_enabled: true
```

تسجيل بدء التشغيل

عند بدء تشغيل التطبيق، يتم تسجيل حالة التكوين:

```
[info] تخزين الرسائل: Mnesia (الاحتفاظ: 24 ساعة)
[info] تصدير CDR: مفعل
[info] حذف بعد التسليم: معطل
[info] شحن OCS: مفعل (url: http://..., tenant: ...)
```

يوفر هذا رؤية فورية للميزات النشطة.

الفيدرالية الجغرافية

العودة إلى فهرس الوثائق | مرجع التكوين | دليل العمليات ←

نظرة عامة

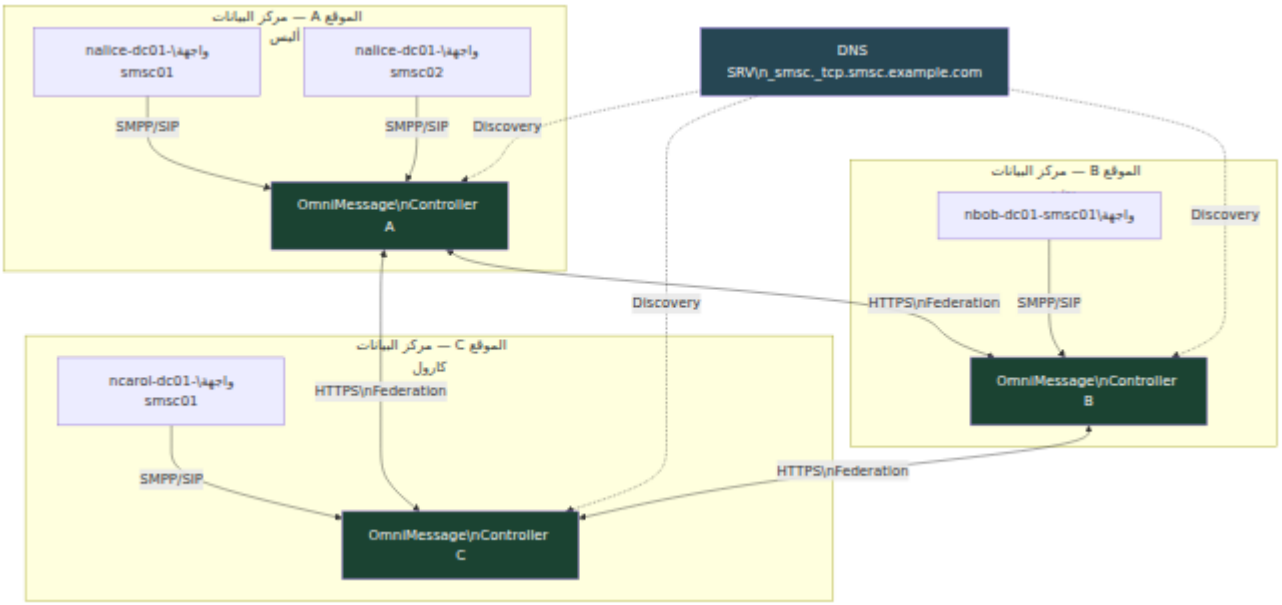
تستخدم OmniMessage **نموذج السحب** لنشر متعدد **الفيدرالية HTTP المعتمدة على** مع المتحكمين عبر مراكز البيانات أو المناطق. يعمل كل متحكم بشكل مستقل — يدير قائمة الرسائل الخاصة به، وجدول التوجيه، ومجموعة من الواجهات المتصلة. يكتشف المتحكمون بعضهم البعض ويتبادلون معلومات الصحة وسجل الواجهات عبر DNS SRV (أو التكوين الثابت)، عبر سجلات HTTPS.

الخاصة Mnesia عندما يحدد التوجيه أن الرسالة تنتمي إلى موقع بعيد، **تبقى الرسالة على بالمتحكم الأصلي**. يتم إرسال إشعار خ��يف إلى المتحكم الوجهة، مما يؤدي إلى إجراء استعلام فوري FederationPoller (وعند الإشعار) الخاص بالمتحكم الوجهة باستعلام دوري. يقوم عن الرسائل الموجهة إلى واجهاته المحلية، ويخزنها، ويجعلها متاحة للواجهات المحلية. بعد التسليم، يقوم المتحكم الوجهة بالإبلاغ عن حالة التسليم إلى الأصل.

الخصائص الرئيسية

الجانب	التفاصيل
الشبكة	يعمل عبر أي WAN، بما في ذلك الروابط غير الموثوقة
الحالة المشتركة	لا شيء — كل متحكم مستقل
ملكية الرسالة	تبقى الرسائل على المتحكم الأصلي حتى يتم تسليمها
الأمان	TLS مع HTTPS
المنافذ	واحد HTTPS (8443) منفذ
هدف النطاق	5-20 متحكم
معالجة التقسيم	لا انقسام للعقل — Poller إعادة المحاولة من قبل

الهيكلية



قرارات التصميم الرئيسية

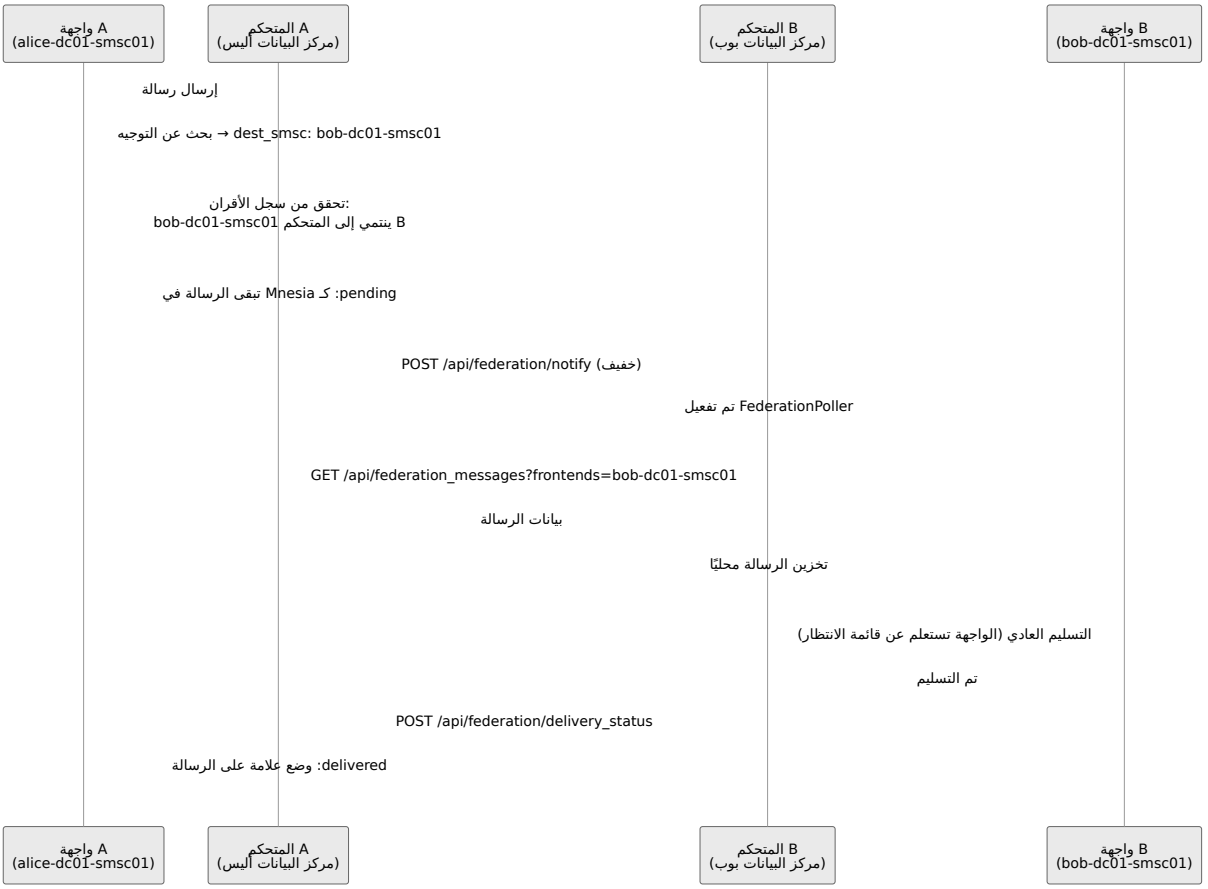
- **تدفق الرسائل (نموذج السحب)**: تبقى الرسائل على المتحكم الأصلي. يقوم المتحكم الوجهة بسحبها عبر الاستعلام الدوري والإشعار عند الطلب. تتواصل الواجهات فقط مع المتحكم المنزلي الخاص بها.
- **معالجة التقسيم**: تعيد Poller المحاولة تلقائيًا. تبقى الرسائل آمنة على المتحكم الأصلي حتى يتم تسليمها بنجاح.
- **مزامنة التكوين**: لا شيء. يدير كل متحكم جدول التوجيه الخاص به بشكل مستقل.
- **شبكة الأقران**: شبكة كاملة. يتبادل جميع المتحكمين معلومات الصحة وسجلات الواجهات مع جميع المتحكمين الآخرين.
- **الاكتشاف**: سجلات DNS SRV (موصى بها) أو قائمة أقران ثابتة (مطلوبة لبيئات DNS بدون ديناميكي).

تدفق الرسائل (نموذج السحب)

عندما تصل رسالة إلى متحكم ويحدد التوجيه أن الواجهة الوجهة موجودة على متحكم بعيد:

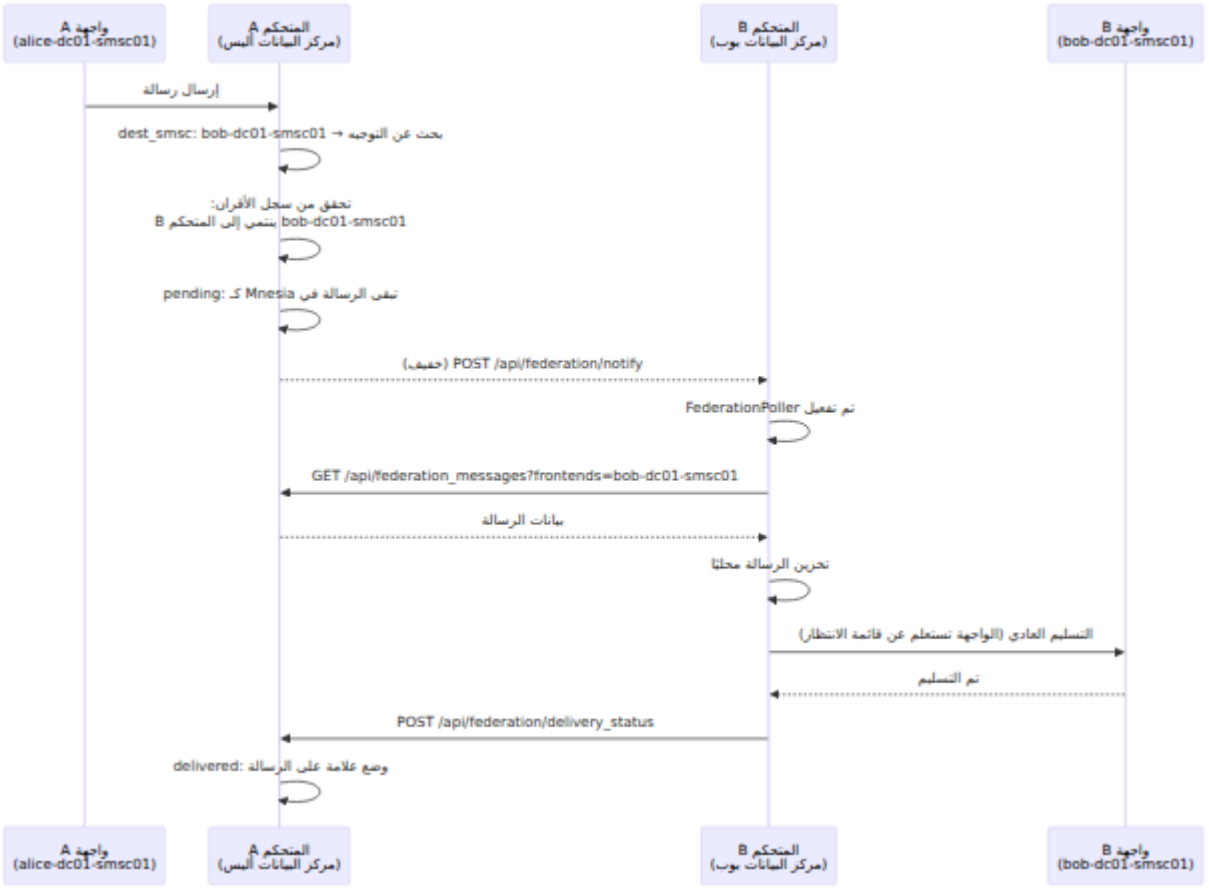
1. تبقى الرسالة في Mnesia الخاصة بالمتحكم الأصلي كـ `:pending`
2. يتم إرسال إشعار خفيف إلى المتحكم الوجهة (إشعار وإهمال)

3. يقوم FederationPoller الخاص بالمتحكم الوجهة بجلب الرسالة عبر GET /api/federation_messages
4. يتم تخزين الرسال�ً محليًا وجعلها متاحة للواجهات المحلية
5. بعد التسليم، يقوم المتحكم الوجهة بالإبلاغ عن الحالة إلى الأصل



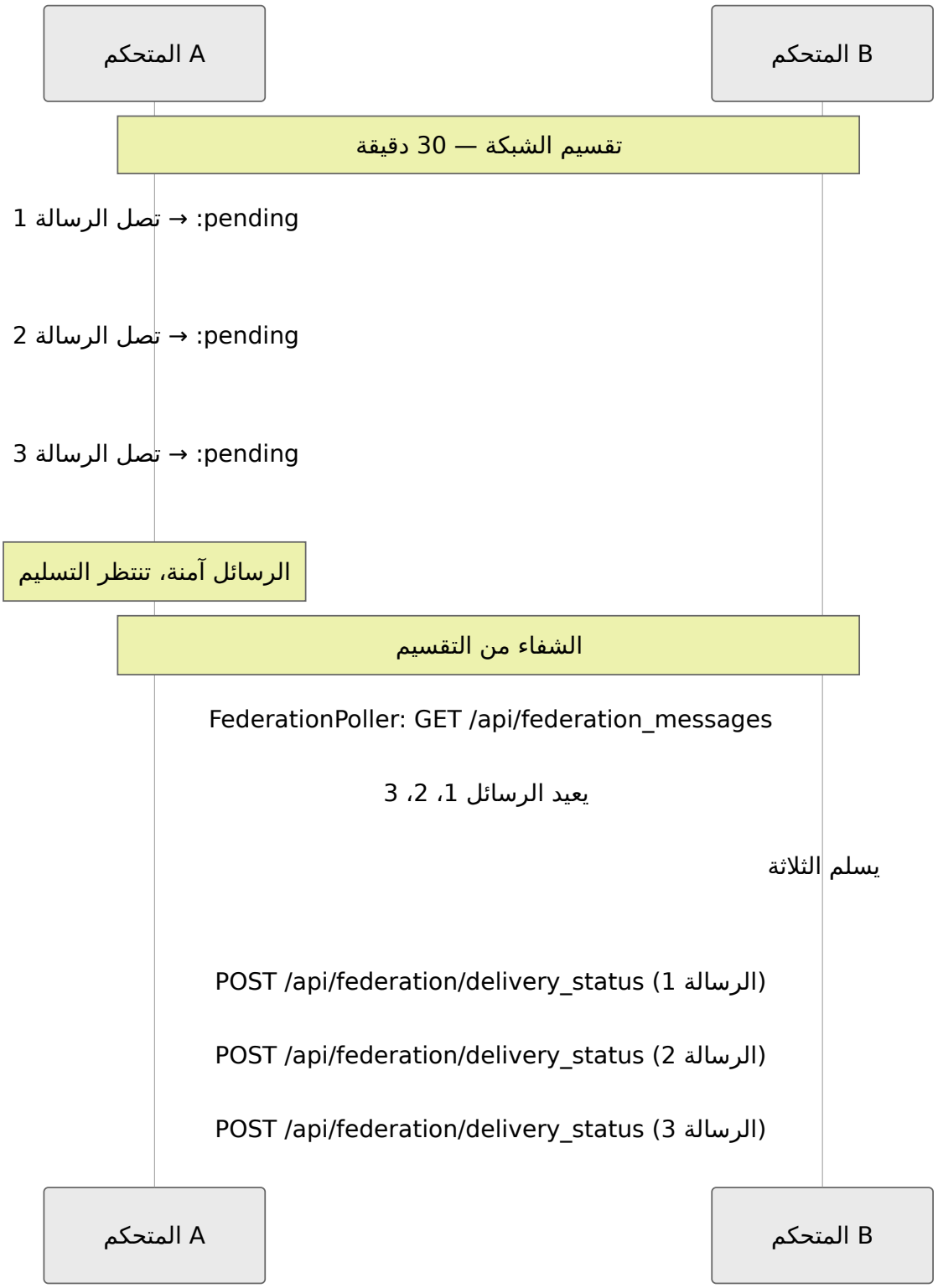
عدم الوصول إلى القرين — فشل الإشعار

إذا كان المتحكم الوجهة معطلاً عند إرسال الإشعار، فإن الإشعار يفشل بصمت. تبقى الرسالة الخاص به FederationPoller على الأصل. عندما يتعافى المتحكم الوجهة، يستأنف `:pending` الاستعلام الدوري ويكتشف الرسالة خلال الدورة التالية (الافتراضي: كل 5 ثوانٍ). لا تفقد أي رسائل.



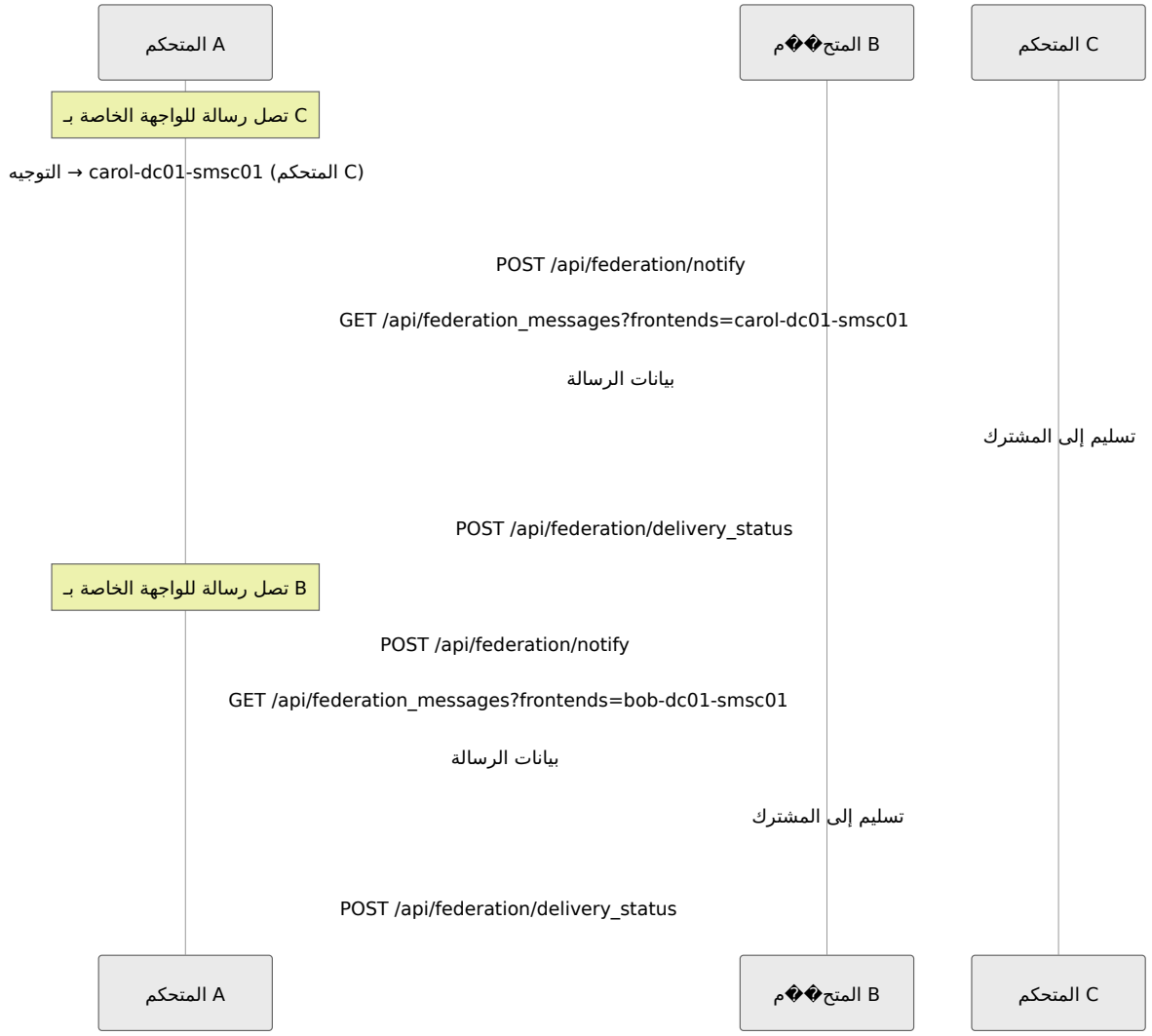
استعادة التقسيم — رسائل متعددة في قائمة الانتظار

خلال تقسيم طويل، تتراكم ��لرسائل على المتحكم الأصلي. عندما يتعافى الرابط، يلحق FederationPoller تلقائيًا — يتم إرجاع جميع الرسائل المعلقة في استعلام واحد.



تدفق الرسائل عبر ثلاثة مواقع

في شبكة متعددة المواقع، يستعلم كل متحكم فقط عن الرسائل الموجهة إلى واجهاته الخاصة.



صحة الأقران ومزامنة السجل

تحافظ المتحكمات الفيدرالية على الوعي ببعضها البعض من خلال دورتين دوريتيين:

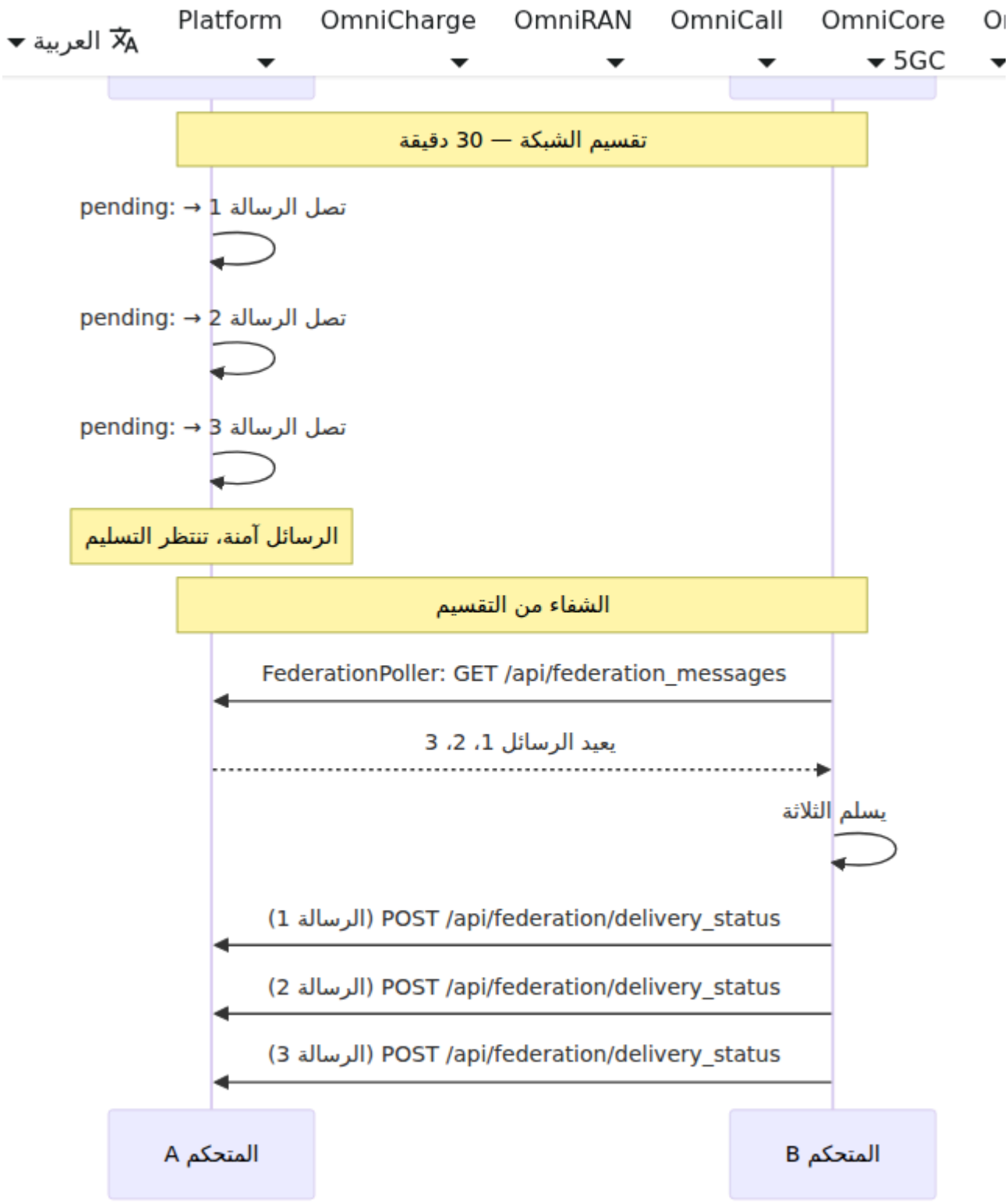
دورة فحص الصحة

كل 10 ثوانٍ (قابلة للتكوين)، يقوم كل متحكم باستدعاء `POST /api/federation/health` على كل قرين معروف. يكون الاتصال ثنائي الاتجاه — يرسل المتصل حالته الخاصة ويتلقى حالة القرين في الاستجابة. يتم وضع علامة على القرين ��أنه **غير صحي** بعد 3 فشل متتالي.

دورة مزامنة السجل

كل 15 ثانية (قابلة للتكوين)، يقوم كل متحكم باستدعاء `POST /api/federation/registry` على كل قرين صحي. يتبادل الاتصال قوائم الواجهات — يرسل المتصل واجهاته النشطة ويتلقى

واجهات القرين النشطة. هذه هي الطريقة التي يتعلم بها المتحكمون من يملك أي واجهة.



حالات الأقران وتأثيرها على استعلام الفيدرالية:

الحالة	فحوصات الصحة	FederationPoller	مزامنة السجل
غير معروف	قيد التقدم	لم يتم الاستعلام	لم يتم المحاولة
صحي	ناجح	استعلام نشط	نشط
متدهور	فشل 1-2	لم يتم الاستعلام	لم يتم المحاولة
غير صحي	فشل 3+	لم يتم الاستعلام	لم يتم المحاولة

التكوين

يتم تكوين الفيدرالية في `config/runtime.exs` تحت المفتاح `:federation`. يتم **تعطيلها** **افتراضيًا** ويجب تمكينها صراحة.

التكوين الأدنى (اكتشاف DNS SRV)

```
# config/runtime.exs
config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "_smsc._tcp.smsc.example.com"
```

مرجع التكوين الكامل

```
# config/runtime.exs
config :sms_c, :federation,
  # مفتاح رئيسي – الفيدرالية غير نشطة تمامًا عندما تكون false
  enabled: true,

  # مجال DNS SRV لاكتشاف الأقران (اتركه فارغًا لاستخدام static_peers
بدلاً من ذلك)
  dns_srv_domain: "_smsc._tcp.smsc.example.com",

  # كم مرة يجب إعادة حل سجلات DNS SRV (بالملي ثانية)
  dns_poll_interval_ms: 30_000,

  # كم مرة يجب تبادل حالة الصحة مع جميع الأقران المعروفين (بالملي
ثانية)
  health_check_interval_ms: 10_000,

  # كم مرة يجب تبادل سجلات الواجهة مع الأقران الصحيين (بالملي ثانية)
  registry_sync_interval_ms: 15_000,

  # كم مرة يستعلم FederationPoller على عن الأقران الصحيين للحصول
الرسائل (بالملي ثانية)
  poll_interval_ms: 5_000,

  # مهلة HTTP لجميع استدعاءات واجهة برمجة التطبيقات بين الأقران
(بالملي ثانية)
  http_timeout_ms: 5_000,

  # منفذ واجهة برمجة التطبيقات المستخدم عند بناء عناوين الأقران من
سجلات DNS SRV
  api_port: 8443,

  # قائمة الأقران الثابتة – تستخدم عندما يكون dns_srv_domain فارغًا
  static_peers: []
```

معلمات الفيدرالية

المعلمة	النوع	مطلوب	الافتراضي	الوصف
`enabled`	Boolean	نعم	`false`	مفتاح رئيسي. عندما تكون `false`، تبدأ مات الفيدرالية ن لا تكتشف أو تتصل بالأقران
`dns_srv_domain`	String	لا	`""`	مجال DNS SI تشاف الأقران ائيً��. عند ركه فارغًا، يتم استخدام `static_pee` بدلاً من ذلك.
`dns_poll_interval_ms`	Integer	لا	`30000`	ل الزمني بين لات إعادة حل بم DNS SRV. نخفضة تكشف عن أقران جدد ل أسرع ولكن زيد من تحميل DNS.
`health_check_interval_ms`	Integer	لا	`10000`	ل الزمني بين جولات فحص صحة. كل جولة صل بكل قرين معروف.
`registry_sync_interval_ms`	Integer	لا	`15000`	ل الزمني بين جولات مزامنة ل الواجهة. كل

المعلمة	النوع	مطلوب	الافتراضي	الوصف
				ولة تتصل بكل قرين صحي.
`poll_interval_ms`	Integer	لا	`5000`	م مرة يستعلم FederationP عن الأقران حيين للحصول لرسائل. تؤدي الإشعارات إلى تعلامات فورية
`http_timeout_ms`	Integer	لا	`5000`	ة لكل استدعاء ي إلى HTTPS ن. ينطبق على وصات الصحة، زامنة السجل، وجلب الرسائل
`api_port`	Integer	لا	`8443`	ذ واجهة برمجة التطبيقات ستخدم عند بناء بن الأقران من سجلات DNS يجب ن تط بق مع ذ واجهة برمجة بيقات المكون في `config :api_ex`.
`static_peers`	List	لا	`[]`	قائمة بتكوينات ن للبيئات التي لا تحتوي على دخال هو .SRV لة تحتوي على

الوصف	الافتراضي	مطلوب	النوع	المعلمة
مفاتيح `host` `port`.				

تكوين القرين الثابت

قم بتكوين الأقران بشكل صريح، DNS SRV لبيئات حيث لا يتوفر:

```
config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "",
  static_peers: [
    %{host: "10.0.1.2", port: 8443},
    %{host: "10.0.2.2", port: 8443},
    %{host: "10.0.3.2", port: 8443}
  ]
```

المعلمة	النوع	مطلوب	الافتراضي	الوصف
`host`	String	نعم	-	عنوان IP أو اسم المضيف للمتحكم البعيد.
`port`	Integer	لا	`8443`	منفذ واجهة برمجة التطبيقات HTTPS للمتحكم البعيد.

إعداد سجل DNS SRV

تسمح سجلات DNS SRV للمتحكمات باكتشاف بعضها البعض تلقائيًا. يقوم كل متحكم بحل المجال المكون ويستخدم أزواج المضيف/المنفذ الناتجة كأقران.

تكوين منطقة DNS:

```
; الأولوية الوزن المنفذ الهدف
_smsc._tcp.smsc.example.com. 86400 IN SRV 10 100 8443 smsc-
alice.smsc.example.com.
_smsc._tcp.smsc.example.com. 86400 IN SRV 10 100 8443 smsc-
bob.smsc.example.com.
_smsc._tcp.smsc.example.com. 86400 IN SRV 10 100 8443 smsc-
carol.smsc.example.com.

; سجلات A للأهداف
smsc-alice.smsc.example.com.    86400 IN A 10.0.1.2
smsc-bob.smsc.example.com.    86400 IN A 10.0.2.2
smsc-carol.smsc.example.com.  86400 IN A 10.0.3.2
```

ستكتشف جميع المتحكمات الموجودة .DNS جديد إلى SRV **إضافة موقع جديد**: أضف سجل القرين الجديد خلال دورة واحدة من `dns_poll_interval_ms` (الافتراضي 30 ثانية).

ستقوم المتحكمات الموجودة بإزالة القرين المغادر .DNS من SRV **إزالة موقع**: قم بإزالة سجل خلال دورة استعلام واحدة.

متطلبات الشبكة

تتطلب الفيدرالية منفذ HTTPS واحد فقط بين المواقع.

المنفذ	البروتوكول	الاتجاه	الغرض
8443	TCP/TLS	ثنائي الاتجاه	جميع حركة الفيدرالية (الصحة، السجل، الإشعار، الرسائل، حالة التسليم)
53	UDP/TCP	صادرة	حل DNS SRV (إذا كنت تستخدم اكتشاف DNS)

قواعد جدار الحماية (لكل موقع):

```
# السماح بحركة الفيدرالية من عناوين IP للمتحكمين الأقران
iptables -A INPUT -p tcp -s 10.0.1.0/24 --dport 8443 -j ACCEPT
iptables -A INPUT -p tcp -s 10.0.2.0/24 --dport 8443 -j ACCEPT
iptables -A INPUT -p tcp -s 10.0.3.0/24 --dport 8443 -j ACCEPT
```

نقاط نهاية واجهة برمجة التطبيقات للفيدرالية

تُقدم جميع نقاط نهاية الفيدرالية على منفذ واجهة برمجة التطبيقات القياسي (8443) تحت `/api/federation/`. يتم استدعاء هذه النقاط من قبل المتحكمات الأقران، وليس من قبل العملاء الخارجيين أو الواجهات.

الغرض	الطريقة	نقطة النهاية
يعيد اسم عقدة هذا المتحكم وإصدار البروتوكول	GET	`/api/federation/identity`
تبادل صحة ثنائي الاتجاه — يرسل المتصل الحالة، ويستجيب المتصل الآخر بحالته الخاصة	POST	`/api/federation/health`
تبادل سجل الواجهة ثنائي الاتجاه	POST	`/api/federation/registry`
استلام إشعار خفيف بأن رسالة متاحة لواجهة محلية	POST	`/api/federation/notify`
استلام تأكيد التسليم من المتحكم الوجهة	POST	`/api/federation/delivery_status`
إعادة الرسائل المعلقة للواجهات المحددة (تستخدم من قبل FederationPoller)	GET	`/api/federation_messages`

تستخدم جميع الطلبات والاستجابات JSON. يتم نقل تعريف القرين في رأس `X-Federation-Node`.

أمثلة النشر

المثال 1: نشر عبر ثلاثة مواقع مع DNS SRV

ثلاث مراكز بيانات، كل منها يحتوي على واجهات محلية لقاعدة مشتركيه. يوفر DNS SRV الاكتشاف.

```
# الموقع A — مركز البيانات أليس (config/runtime.exs)
config :sms_c,
  smsc_node_name: "alice-dc01-smsc01"

config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "_smsc._tcp.smsc.example.com"
```

```
# الموقع B — مركز البيانات بوب (config/runtime.exs)
config :sms_c,
  smsc_node_name: "bob-dc01-smsc01"

config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "_smsc._tcp.smsc.example.com"
```

```
# الم��قع C — مركز البيانات كارول (config/runtime.exs)
config :sms_c,
  smsc_node_name: "carol-dc01-smsc01"

config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "_smsc._tcp.smsc.example.com"
```

كيف يعمل: تقوم جميع المتحكمات الثلاثة بحل نفس مجال SRV وتكتشف بعضها تلقائيًا. يسجل كل متحكم واجهاته المحلية. عندما تصل رسالة إلى مركز البيانات أليس موجهة إلى مشترك يخدمه مركز البيانات بوب، يختار جدول التوجيه كوجهة. تكتشف طبقة الفيدرالية أن `bob-dc01-smsc01` الخاص بالمتحكم B هذه الواجهة تنتمي إلى المتحكم FederationPoller وترسل إشعارًا. يستعلم B عن المتحكم A ويخزن الرسالة للتسليم المحلي.

المثال 2: نشر نشط-نشط عبر موقعين مع أقران ثابتين

مركزان للبيانات متصلان عبر رابط مخصص. لا يتوفر DNS SRV.

```
# مركز البيانات الرئيسي (config/runtime.exs)
config :sms_c,
  smsc_node_name: "primary-smsc01"

config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "",
  static_peers: [
    %{host: "10.200.1.5", port: 8443}
  ]
```

```
# مركز البيانات الثانوي (config/runtime.exs)
config :sms_c,
  smsc_node_name: "secondary-smsc01"

config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "",
  static_peers: [
    %{host: "10.100.1.5", port: 8443}
  ]
```

كيف يعمل: يتم تكوين كل متحكم بعنوان IP الخاص بالآخر. تستمر فحوصات الصحة ومزامنات FederationPoller السجل عبر الرابط المخصص. تبقى الرسائل على أي متحكم استقبلها. يستعلم الخاص بكل متحكم عن القرين للحصول على الرسائل الموجهة إلى الواجهات المحلية.

المثال 3: استعلام عدواني للروابط ذات الكمون المنخفض

للمواقع المتصلة بروابط موثوقة وذات كمون منخفض حيث تكون الفيدرالية السريعة مهمة:

```
config :sms_c, :federation,
  enabled: true,
  dns_srv_domain: "_smsc._tcp.cluster.internal",
  health_check_interval_ms: 5_000,
  registry_sync_interval_ms: 5_000,
  poll_interval_ms: 1_000,
  http_timeout_ms: 2_000
```

حالة الاستخدام: نشرات الحرم الجامعي أو المناطق الحضرية حيث تكون المواقع مت��لة وتكون الفشل السريع أكثر أهمية من تقليل حركة التحكم RTT بروابط أقل من 10 مللي ثانية.

المقاييس

تظهر الفيدرالية الأحداث التالية التي يمكن ملاحظتها عبر Prometheus 9568 على المنفذ.

المقياس: `sms_c_federation_message_received_count` عدد **الوصف**: عداد **النوع**: إشعارات الفيدرالية المستلمة من الأقران **التسميات**:

- `origin_node` - القرين الذي أرسل الإشعار

المقياس: `sms_c_federation_health_check_count` نتائج فحص **الوصف**: عداد **النوع**: الصحة لكل قرين **التسميات**:

- `peer` - معرف القرين
- `result` - `ok` أو `failed`

استعلامات المثال:

```
# معدل الإشعارات في الدقيقة
rate(sms_c_federation_message_received_count[5m]) * 60

# فشل فحوصات الصحة حسب القرين
sum by (peer)
(rate(sms_c_federation_health_check_count{result="failed"}[5m]))
```

المؤشرات الرئيسية للمراقبة

عتبة التنبيه	أين	ما يجب مراقبته
عدد متزايد (القرين لا يسلم)	`GET /api/federation/status` أو واجهة الويب	الرسائل المخبأة الفيدرالية
أي قرين غير صحي > 5 دقائق	`GET /api/federation/identity` على كل قرين	حالة صحة القرين
ثوانٍ (تدهور 2 > الرابط)	سجلات التطبيق	زمن استجابة فحص الصحة

استكشاف الأخطاء وإصلاحها

القرين غير مكتشف

الأعراض: سجلات المتحكم تظهر عدم اكتشاف أي أقران؛ `cluster_status` تعيد فارغة.

الأسباب المحتملة:

- مجال DNS SRV غير صحيح أو غير قابل للحل
- خادم DNS غير قابل للوصول من مضيف المتحكم
- سجلات SRV لم تنتشر بعد
- الفيدرالية غير مفعلة (`enabled: false`)

الحل:

1. تحقق من حل DNS SRV من مضيف المتحكم: `dig SRV _smsc._tcp.smsc.example.com`
2. تحقق من أن `enabled: true` تم تعيينه في تكوين الفيدرالية
3. تحقق من سجلات التطبيق بحثًا عن رسائل "اكتشاف الفيدرالي" عند بدء التشغيل
4. إذا كنت تستخدم أقران ثابتة، تحقق من أن قائمة `static_peers` تحتوي على إدخالات مضيف/منفذ صح��حة

القرين مُعَلّم بأنه غير صحي

الأعراض: الرسائل إلى موقع محدد لا يتم استعلامها. تظهر السجلات رسائل "فحص الصحة فشل".

الأسباب المحتملة:

- المتحكم البعيد معطل
- جدار الحماية يحظر المنفذ 8443 بين المواقع
- مشاكل في شهادة TLS
- تقسيم الشبكة بين المواقع

الحل:

1. تحقق من أن المتحكم البعيد يعمل: `curl -k https://<peer-host>:8443/api/federation/identity`
2. تحقق من أن قواعد جدار الحماية تسمح بحركة المرور على المنفذ 8443 من عنوان IP للمتحكم المحلي
3. راجع سجلات التطبيق على كلا الجانبين بحثًا عن أخطاء TLS أو الاتصال
4. تحقق من الاتصال الشبكي بين المواقع

الرسائل لا تُسَلَّم إلى الموقع البعيد

الأعراض: تبقى الرسائل على المتحكم الأصلي. عدد الرسائل المخبأة الفيدرالية هو 0 `:pending` على الوجهة.

الأسباب المحتملة:

- المتحكم الوجهة غير صحي (يستعلم فقط عن الأقران الصحيين FederationPoller)
- لا يعمل FederationPoller
- سجل الواجهة غير متزامن (الوجهة لا تعرف أي الواجهات محلية)

الحل:

1. تحقق من حالة القرين على المتحكم الوجهة
2. تحقق من أن FederationPoller موجود في شجرة الإشراف (تحقق من سجلات التطبيق بحثًا عن "تم بدء تشغيل الفيدرالي Poller")

3. تحقق من أن سجلات الواجهات متزامنة (`GET /api/federation/status` — تحقق من قوائم الواجهات)
4. اختبر يدويًا نقطة نهاية federation_messages: `curl -k 'https://<origin>:8443/api/federation_messages?frontends=<frontend_name>&include_unrouted=false'`

بحث المشترك HSS (Diameter Sh)

العودة إلى فهرس الوثائق | دليل توجيه الرسائل القصيرة ←

نظرة عامة

يمكن لـ OmniMessage استعلام خادم المشتركين المنزليين (HSS) عبر واجهة Diameter Sh لتحديد ما إذا كان المشترك المدعو ينتمي إلى الشبكة قبل اتخاذ قرار التوجيه. يمنع ذلك توجيه الرسائل الموجهة إلى المشتركين على الشبكة بشكل غير صحيح عبر بوابة افتراضية عندما يكون المشترك غير متصل مؤقتًا أو غير مسجل.

بدون تمكين بحث HSS، ستسقط الرسالة الموجهة إلى مشترك على الشبكة غير المسجل حاليًا إلى جدول التوجيه القياسي. إذا كان هناك مسار شامل أو افتراضي، فسيتم إرسال الرسالة إلى بوابة خارجية - وهو أمر غير صحيح لمشترك ينتمي إلى الشبكة.

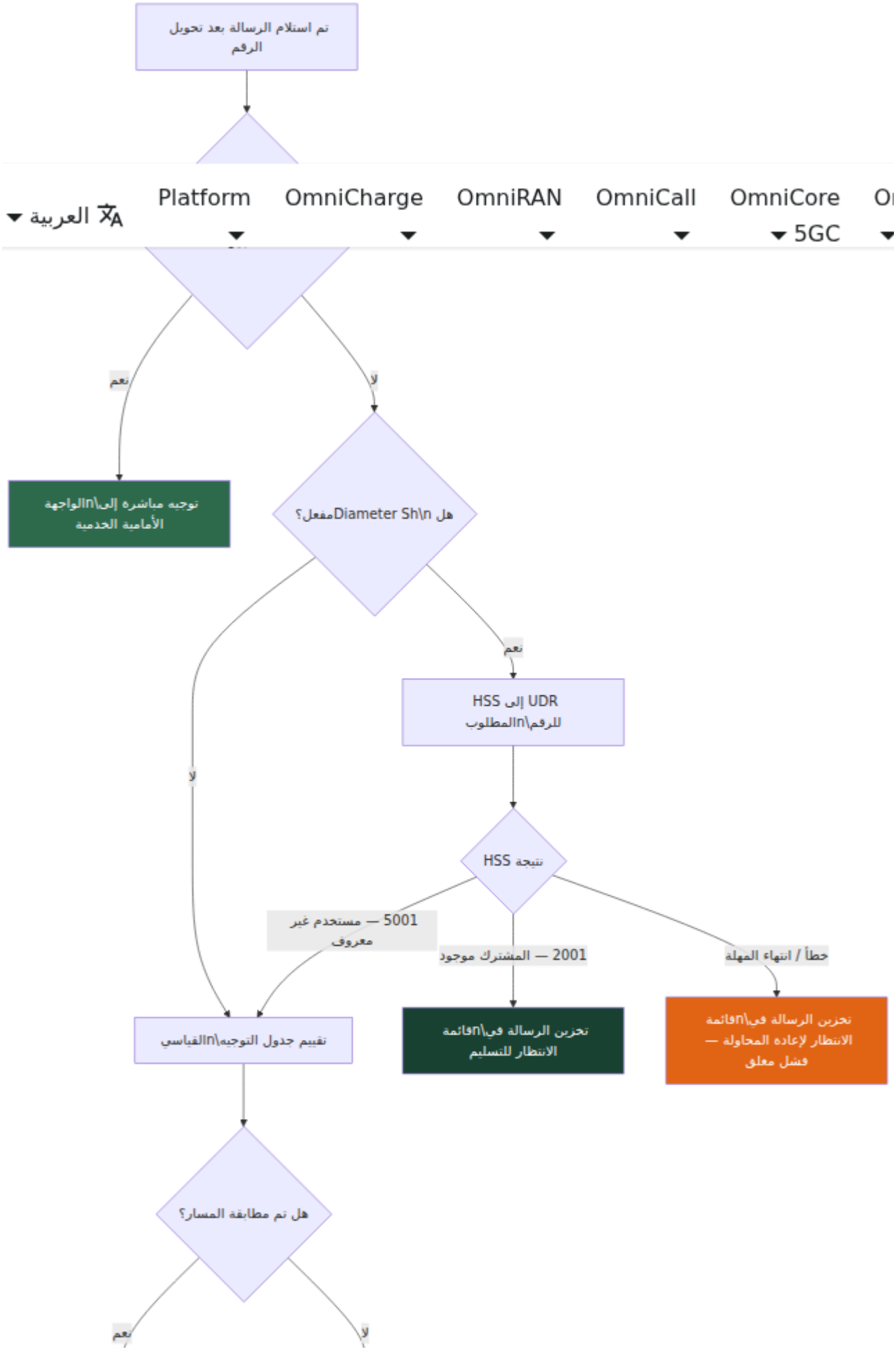
مع تمكين بحث HSS، ترسل OmniMessage **طلب بيانات المستخدم (UDR)** إلى HSS للرقم المدعو. إذا أكد وجود المشترك (رمز النتيجة 2001)، يتم الاحتفاظ بالرسالة في قائمة انتظار الرسائل للتسليم عند التسجيل التالي للمشترك. إذا أعاد HSS "رمز" (رمز النتيجة 5001)، فإن المشترك خارج الشبكة ويستمر التوجيه بشكل طبيعي.

المواصفات ذات الصلة

العنوان	المواصفة
واجهة Sh المعتمدة على Diameter	3GPP TS 29.329
واجهة IMS Sh — تدفقات الإشارات ومحتويات الرسائل	3GPP TS 29.328
بروتوكول Diameter الأساسي	RFC 6733

سلسلة أولوية التوجيه

كـ **الأولوية 2**، بين فحص الموقع/ OmniMessage مع قرار التوجيه في HSS يتناسب بحث التسجيل وجدول التوجيه المعتمد على التعبيرات العادي��.


تم استلام الرسالة بعد تحويل الرقم
نعم
لا
توجيه مباشرة إلى\nالواجهة الأمامية الحذمية
هل \nDiameter Shمفعل؟
نعم
لا
UDR إلى HSS
للرقم\nالمطلوب
نتيجة HSS
5001 — مستخدم غير معروف
2001 — المشترك موجود
خطأ / انتهاء المهلة
تقييم جدول التوجيه\nالقياسي
تخزين الرسالة في\nقائمة الانتظار للتسليم
تخزين الرسالة في\nقائمة الانتظار لإعاده المحاولة — فشل معلق
هل تم مطابقة المسار؟
نعم
لا

الأولوية 1 — التوجيه المعتمد على الموقع

إذا تم العثور على MSISDN الوجهة في جدول التسجيلات/المواقع، ي��م توجيه الرسالة مباشرة إلى الواجهة الأمامية التي تخدم ذلك المشترك. لا يحدث بحث أو تقييم جدول التوجيه HSS.

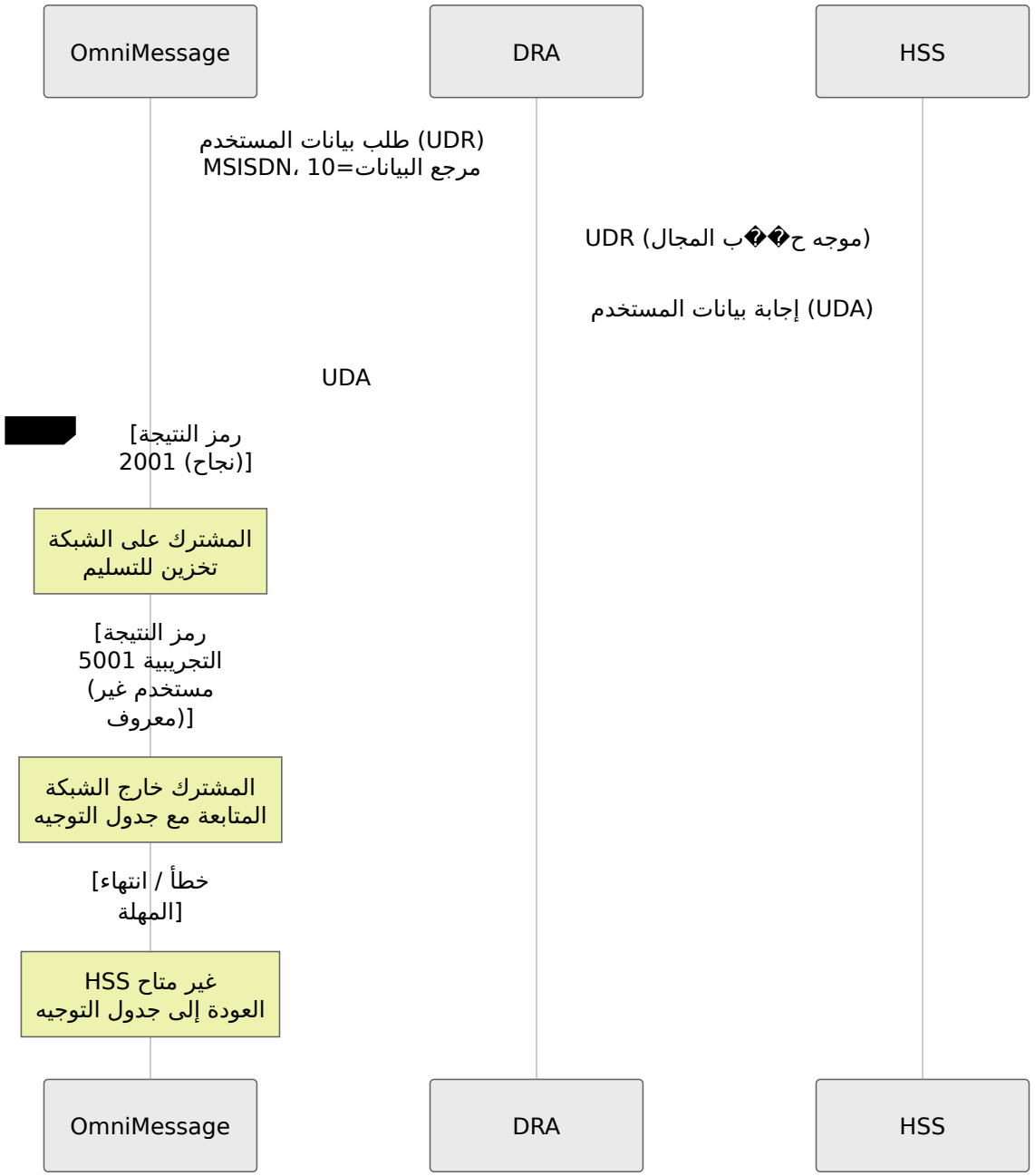
الأولوية 2 — بحث مشترك HSS

إذا لم يتم العثور على تسجيل وكان HSS عن OmniMessage مفعلًا، تستعلم Diameter Sh لتحديد ما إذا كان المشترك على الشبكة. سيتم تخزين رسالة مشترك على الشبكة غير المتصل ببساطة في قائمة الانتظار — سيتم تسليمها عندما يسجل المشترك وتصل تحديثات الموقع.

الأولوية 3 — جدول التوجيه

إذا أكد HSS أن المشترك غير معروف (خارج الشبكة)، أو إذا لم يكن Diameter مفعلًا، يتم تقييم جدول التوجيه القياسي المعتمد على التعبيرات العادية. راجع دليل توجيه الرسائل القصيرة للحصول على التفاصيل.

Diameter Sh تدفق رسالة



معلمات UDR

AVP	القيمة	الوصف
Session-Id	تم إنشاؤه تلقائيًا	معرف الجلسة الفريد
Auth-Session-State	1 (NO_STATE_MAINTAINED)	لا حاجة لحالة الجلسة للاستعلامات
Data-Reference	10	طلب جميع بيانات المستخدم
Origin-Host	المضيف المكون + المجال	هوية Diameter الخاصة بـ OmniMessage
Origin-Realm	المجال المكون	مجال Diameter الخاص بـ OmniMessage
Destination-Realm	المجال المكون	مجال HSS (موجه عبر DRA)
User-Identity	MSISDN مشفر بتنسيق TBCD	رقم المشترك المدعو
Server-Name	"OmniMessage"	هوية خادم التطبيق
Vendor-Specific-Application-Id	Auth: 16777217، Vendor: 10415	معرف تطبيق Sh وفقًا لـ 3GPP TS 29.329

رموز نتائج UDA

الرمز	النوع	المعنى	إجراء OmniMessage
2001	رمز النتيجة	نجاح — تم العثور على المشترك	تخزين الرسالة للتسليم
5001	رمز النتيجة التجريبية	مستخدم غير معروف — غير موجود في HSS	المتابعة مع جدول التوجيه
آخر	رمز النتيجة التجريبية	خطأ غير متوقع	تسجيل تحذير، تخزين لإعادة المحاولة (فشل مغلق)

التكوين

تمكين بحث HSS

يتم التحكم في بحث HSS بواسطة كتلتين تكوين في `config/runtime.exs`:

1. علامة `diameter_enabled` تحت `:sms_c` — تتحكم فيما إذا كان يتم إجراء استعلام HSS في مسار التوجيه
2. تكوين `:diameter_ex` — يقوم بتكوين كومة Diameter نفسها (الهوية، الأقران، التطبيقات)

التكوين الأدنى

```
# تمكين استعلام HSS في مسار التوجيه
config :sms_c,
  diameter_enabled: true

# تكوين كومة Diameter
config :diameter_ex,
  diameter: %{
    service_name: :omnimessage,
    listen_ip: "0.0.0.0",
    listen_port: 3868,
    decode_format: :map,
    host: "smsc01",
    realm: "epc.mnc005.mcc547.3gppnetwork.org",
    product_name: "OmniMessage",
    request_timeout: 5000,
    peer_selection_algorithm: :random,
    allow_undefined_peers_to_connect: true,
    log_unauthorized_peer_connection_attempts: true,
    control_module: SmsC.Diameter.Control,
    processor_module: DiameterEx.Processor,
    auth_application_ids: [],
    acct_application_ids: [],
    vendor_id: 10415,
    supported_vendor_ids: [10415],
    applications: [
      %{
        application_name: :sh,
        application_dictionary: :diameter_gen_3gpp_sh,
        vendor_specific_application_ids: [
          %{
            vendor_id: 10415,
            auth_application_id: 16_777_217,
            acct_application_id: nil
          }
        ]
      }
    ],
    peers: [
      %{
        port: 3868,
        host: "dra01.epc.mnc005.mcc547.3gppnetwork.org",
```

```
        ip: "10.0.0.1",
        realm: "epc.mnc005.mcc547.3gppnetwork.org",
        tls: false,
        transport: :diameter_tcp,
        initiate_connection: true
      }
    ]
  }
```

معلمات OmniMessage

المعلمة	النوع	مطلوب	الافتراضي	الوصف
`diameter_enabled`	Boolean	لا	`false`	تمكين أو تعطيل استعلام مشترك HSS في مسار التوجيه. عندما تكون `false`، تتجاوز جميع الرسائل فحص HSS وتذهب مباشرة إلى جدول التوجيه.
`mock_sh`	Boolean	لا	`false`	استخدام استجابات HSS وهمية بدلاً من استعلامات Diameter الحقيقية. للاختبار فقط.
`mock_sh_on_net_numbers`	قائمة	لا	`[]`	قائمة من MSISDNs التي يتم اعتبارها على الشبكة عندما تكون `mock_sh` هي `true`.

معلمات كومة Diameter

المعلمة	النوع	مطلوب	الافتراضي
`service_name`	Atom	نعم	-
`listen_ip`	String	لا	`"0.0.0`
`listen_port`	Integer	لا	3868
`decode_format`	Atom	نعم	-
`host`	String	نعم	-
`realm`	String	نعم	-
`product_name`	String	لا	-
`request_timeout`	Integer	لا	5000
`peer_selection_algorithm`	Atom	لا	`:rando`
`allow_undefined_peers_to_connect`	Boolean	لا	`true`

المعلمة	النوع	مطلوب	تراضي
`log_unauthorized_peer_connection_attempts`	Boolean	لا	`true`
`control_module`	Module	نعم	-
`processor_module`	Module	نعم	-
`auth_application_ids`	قائمة	لا	`[]`
`acct_application_ids`	قائمة	لا	`[]`
`vendor_id`	Integer	نعم	-
`supported_vendor_ids`	قائمة	نعم	-

معلمات التطبيق

تقوم قائمة `applications` بتكوين التطبيقات التي تم تمكينها في Diameter. بالنسبة لبحث Sh، يتطلب الأمر فقط تطبيق HSS.

المعلمة	النوع	مطلوب	الافتراضي	
`application_name`	Atom	نعم	-	أن يكون `:sh` ◆◆
`application_dictionary`	Atom	نعم	-	ة قاموس Erlang. `:diame`
`vendor_specific_application_ids`	قائمة	نعم	-	المحددة نظر أدناه

معرف التطبيق المحدد للبائع (Sh):

المعلمة	النوع	القيمة	الوصف
`vendor_id`	Integer	`10415`	معرف بائع 3GPP
`auth_application_id`	Integer	`16_777_217`	معرف تطبيق Sh 3 وفقًا لـ 3GPP TS 29.329
`acct_application_id`	-	`nil`	لا تستخدم لـ Sh

معلمات الأقران

تحدد قائمة `peers` أقران Diameter (عادةً DRAs أو HSS مباشرة) التي تتصل بها OmniMessage.

المعلمة	النوع	مطلوب	الافتراضي	الوصف
`host`	String	نعم	-	هوية مضيف اصة Diameter بالند (FQDN). تتطابق تمامًا مع وية المكونة للند
`ip`	String	نعم	-	لاتصال IP عنوان TCP.
`port`	Integer	لا	3868	منفذ Diamete للند.
`realm`	String	نعم	-	مجال Diamete للند.
`tls`	Boolean	لا	`false`	لهذا TLS تمكين الاتصال بالند.
`transport`	Atom	لا	`:diameter_tcp`	بروتوكول النقل: `:diameter_t` أو `:diameter_s`
`initiate_connection`	Boolean	لا	`true`	إذا كانت `true`، تقوم OmniMessag ببدء اتصال TCP بالند. إذا كانت `false`، تنتظر OmniMessag اتصال الند.

أقران متعددة

قم بتكوين أقران متعددة من أجل التكرار. تختار OmniMessage من بين الأقران المتصلة باستخدام `peer_selection_algorithm`:

```
peers: [
  %{
    port: 3868,
    host: "dra01.epc.mnc005.mcc547.3gppnetwork.org",
    ip: "10.0.0.1",
    realm: "epc.mnc005.mcc547.3gppnetwork.org",
    tls: false,
    transport: :diameter_tcp,
    initiate_connection: true
  },
  %{
    port: 3868,
    host: "dra02.epc.mnc005.mcc547.3gppnetwork.org",
    ip: "10.0.0.2",
    realm: "epc.mnc005.mcc547.3gppnetwork.org",
    tls: false,
    transport: :diameter_tcp,
    initiate_connection: true
  }
]
```

وضع المحاكاة

لبيئات الاختبار بدون HSS مباشر، قم بتمكين وضع المحاكاة:

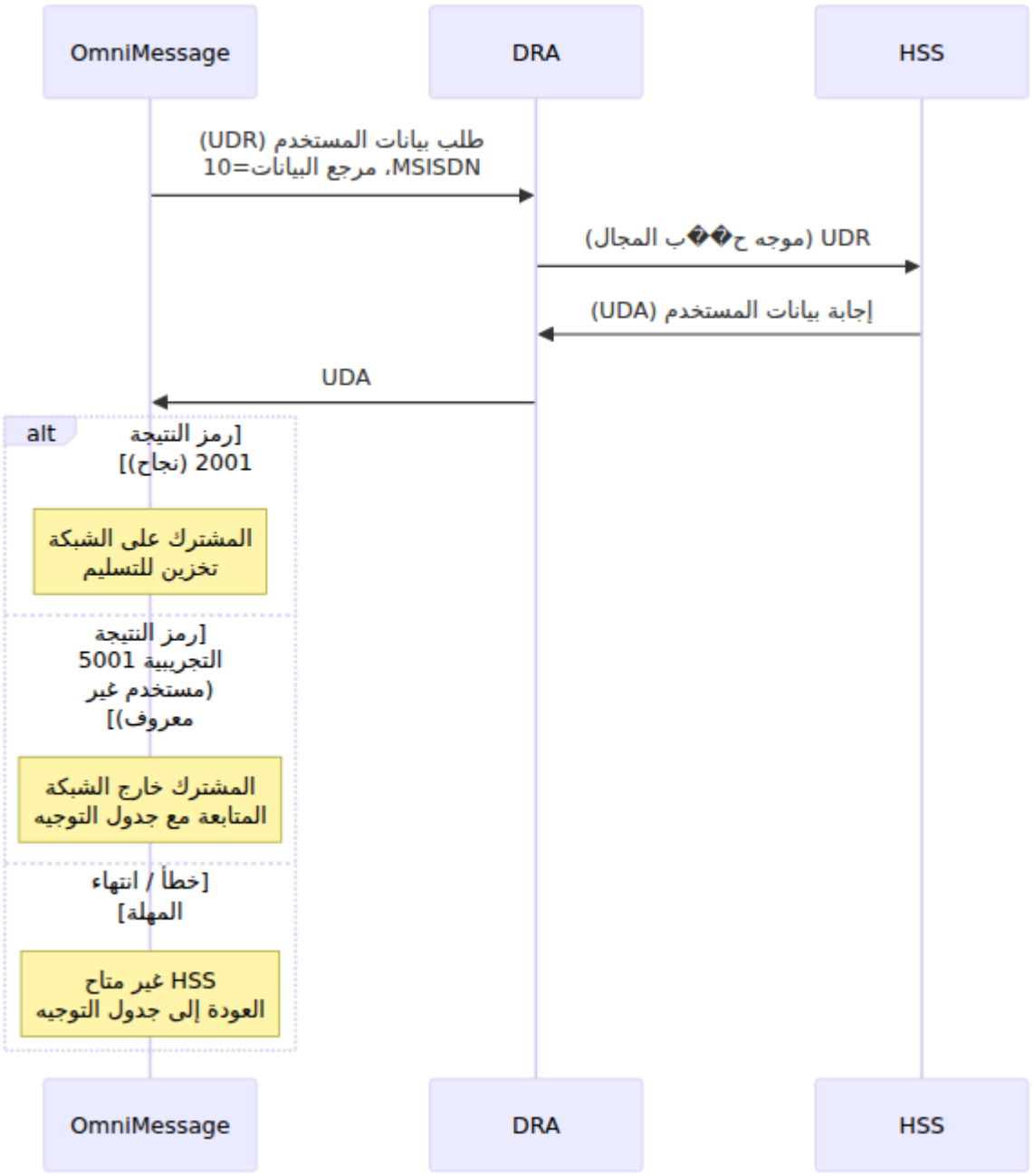
```
config :sms_c,
  diameter_enabled: true,
  mock_sh: true,
  mock_sh_on_net_numbers: [
    "68987654321",
    "68912345678"
  ]
```

كمشترك `mock_sh_on_net_numbers` في قائمة MSISDN في وضع المحاكاة، يتم اعتبار أي MSISDNs الأخرى كمجهولة. يتم اعتبار جميع HSS 2001) معادل لرمز نتيجة) على الشبكة ولا يتم إجراء اتصالات الشبكة Diameter (معادلة لرمز النتيجة 5001). لا يتم بدء كومة.

تسليم التخزين والتسليم

عندما يتم تخزين رسالة لمشترك على الشبكة غير المسجل، تبقى الرسالة في قائمة انتظار رسائل Mnesia مع تعيين `dest_smsc` إلى `nil`. عندما يسجل المشترك وتصل تحديثات الموقع، تلتقط MSISDN حلقة التسليم الرسائل المعلقة لذلك وتوجهها إلى الواجهة الأمامية الخدمية.

تكون الرسائل في قائمة الانتظار خاضعة لسياسة الاحتفاظ القياسية. إذا لم يسجل المشترك قبل انتهاء صلاحية الرسالة، يتم نقل الرسالة إلى قائمة الرسائل الميتة وفقًا لـ `dead_letter_time_minutes` المكونة.



ملاحظات تشغيلية

تسجيل بدء التشغيل

عند بدء التشغيل Diameter Sh حالة OmniMessage تسجل:

- **مفعل**: `Diameter Sh (HSS): ENABLED (host: smsc01, realm: epc.mnc005.mcc547.3gppnetwork.org, peers: 2)`
- **معطل**: `Diameter Sh (HSS): DISABLED`

HSS غير متاح (فشل مغلق)

إذا فشل بحث HSS — لأي سبب — انتهاء المهلة، عدم وجود أقران متصلين، استجابة غير معالجة تسجل OmniMessage تحذيرًا و**تخزن الرسالة لإعادة المحاولة**. تبقى الرسالة مع تعيين `dest_smsc` إلى `nil` وسيتم إعادة تقييمها في دورة الاستطلاع التالية، حيث سيتم محاولة استعلام HSS مرة أخرى.

هذا تصميم **فشل مغلق**: بدون إجابة قاطعة من HSS، لن تقوم OmniMessage بتوجيه الرسالة إلى بوابة خارج الشبكة. يمنع ذلك المشتركين على الشبكة من إرسال رسائلهم بشكل غير صحيح إلى شركات النقل الخارجية خلال انقطاع HSS.

عندما **يتم تعطيل صراحة Diameter** (`diameter_enabled: false`)، يتم توجيه الرسائل عبر جدول التوجيه القياسي دون أي فحص HSS. سلوك الفشل المغلق — هذا خيار مشغل مقصود. ينطبق Diameter فقط عندما يكون مفعلًا ولكن الفحص يفشل.

إدخالات سجل الأحداث

تسجل HSS نشطًا أنواع الأحداث التالية في سجل أحداث الرسائل عندما يكون بحث:

نوع الحدث	الوصف
`hss_dip_on_net`	تم العثور على المشترك في HSS (رمز النتيجة 2001). تم تخزين الرسالة للتسليم.
`hss_dip_off_net`	لم يتم العثور على المشترك في HSS (رمز النتيجة 5001). المتابعة مع التوجيه القياسي.
`hss_dip_error`	فشل بحث HSS (انتهاء المهلة، عدم الاتصال، إلخ). تم تخزين الرسالة لإعادة المحاولة.

تتضمن جميع الأحداث بيانات وصفية منظمة: `result_code`، `hss` (اسم مضيف الند)، `duration_ms`، `session_id`، `destination_msisdn`، و`message_id`.

تستمر أحداث التوجيه القياسية (`sms_routing_started`، `sms_routing_selected`، إلخ) في التسجيل للرسائل التي تتابع إلى جدول التوجيه.

فرض التوجيه على الشبكة على مستوى المسار

ذي الأولوية 2 (الذي ينطبق عالميًا قبل التوجيه)، يمكن أن تفرض HSS بالإضافة إلى استعلام المسارات الفردية متطلبات على الشبكة باستخدام ثلاثة علامات. يتم التحقق من هذه العلامات **بعد** مطابقة المسار و**قبل** تعيين الرسالة إلى `dest_smsc` لذلك المسار.

علامات المسار

العلامة	الوصف	فحص HSS	سلوك الفشل
`on_net_only`	اسم مستعار قديم لـ `terminating_on_net_only`	يتحقق من الرقم المدعو	إسقاط الرسالة
`originating_on_net_only`	يتطلب أن يكون المرسل (الطرف المتصل) على الشبكة	يتحقق من الرقم المتصل	إسقاط الرسالة
`terminating_on_net_only`	يتطلب أن يكون الوجهة (الطرف المدعو) على الشبكة	يتحقق من الرقم المدعو	إسقاط الرسالة

تعمل جميع العلامات الثلاثة **بشكل مغلق عند الفشل**: إذا كان HSS غير متاح أو تم تعطيل Diameter، يتم إسقاط الرسالة بدلاً من تسليمها إلى وجهة غير موثوقة.

مثال على التكوين

```
config :sms_c, :sms_routes, [
  # بوابة SMPP على الشبكة – تقبل فقط الرسائل الموجهة إلى وجهات مؤكدة على الشبكة
  %{
    calling_regex: nil,
    called_regex: ~r/^1907/,
    source_smsc: nil,
    dest_smsc: "local-gateway",
    source_type: nil,
    auto_reply: false,
    drop: false,
    charged: :default,
    weight: 100,
    priority: 10,
    description: "بوابة محلية – وجهات على الشبكة فقط",
    on_net_only: false,
    originating_on_net_only: false,
    terminating_on_net_only: true,
    enabled: true
  },

  # تراجع خارج الشبكة – شامل لجميع المشتركين المؤكدين خارج الشبكة بواسطة HSS
  %{
    calling_regex: nil,
    called_regex: nil,
    source_smsc: nil,
    dest_smsc: "external-gateway",
    source_type: :ims,
    auto_reply: false,
    drop: false,
    charged: :default,
    weight: 100,
    priority: 255,
    description: "بوابة خارجية – تراجع خارج الشبكة",
    on_net_only: false,
    originating_on_net_only: false,
    terminating_on_net_only: false,
    enabled: true
```

```
  }
]
```

كيف يعمل: عندما تصل رسالة لمشترك غير مسجل حاليًا:

1. **فحص الموقع** — لم يتم العثور على تسجيل نشط
2. **استعلام HSS** — تم تأكيد المشترك على الشبكة → تم تخزين الرسالة للتسليم (لا توجيه)
3. **استعلام HSS** — تم تأكيد المشترك خارج الشبكة → يتم تقييم جدول المسار
4. يتطابق مسار "local-gateway" ولكن لديه `terminating_on_net_only: true` — تم تأكيد HSS بالفعل خارج الشبكة، لذا يتم تخطي هذا ا��مسار (يتم إسقاط الرسالة في هذا المسار)
5. يتطابق مسار "external-gateway" بدون متطلبات على الشبكة → يتم توجيه الرسالة إلى بوابة خارجية

حالة الاستخدام: منع حركة المرور خارج الشبكة من إرسالها إلى البنية التحتية المحلية التي يمكنها فقط خدمة المشتركين على الشبكة، مع ضمان وصول حركة المرور خارج الشبكة إلى شركة نقل خارجية.

التفاعل مع استعلام HSS ذي الأولوية 2

يخدم استعلام HSS ذي الأولوية 2 وعلامات مستوى المسار على الشبكة أغراضًا مختلفة:

الآلية	متى	الغرض	سلوك الفشل
استعلام HSS ذي الأولوية 2	قبل التوجيه	تخزين المشتركين على الشبكة للتسليم	فشل مغلق (تخزين لإعادة المحاولة)
`terminating_on_net_only`	بعد مطابقة المسار	فرض متطلبات على الشبكة على مسارات محددة	فشل مغلق (إسقاط الرسالة)
`originating_on_net_only`	بعد مطابقة المسار	فرض متطلبات على الشبكة على المنشئ	فشل مغلق (إسقاط الرسالة)

انتهاء صلاحية الموقع

على تسجيلات `expires` يتحقق التوجيه المعتمد على الموقع (الأولوية 1) من الطابع الزمني المشتركين. إذا انتهت صلاحية تسجيل المشترك — على سبيل المثال، دخلوا وضع الطيران — يتم والأولوية 3 (جدول HSS استعلام) اعتبار الموقع غير نشط وتستمر عملية التوجيه إلى الأولوية 2 (التوجيه.

هذا يمنع إرسال الرسائل إلى واجهة أمامية لم تعد قادرة على الوصول إلى المشترك، مما يتجنب فشل التسليم غير الضروري ودورات إعادة المحاولة.

استكشاف الأخطاء وإصلاحها

الرسائل لا تزال تتجه إلى بوابة افتراضية للمشتركين على الشبكة

الأعراض: يتم إرسال رسائل إلى مشتركين على الشبكة غير المتصلين عبر البوابة الافتراضية بدلاً من تخزينها.

الأسباب المحتملة:

- تم تعيين `diameter_enabled` إلى `false`
- لا توجد أقران Diameter (سجلات التطبيق لحالة اتصال الند متصلة (تحقق م
- HSS تحقق من السجلات عن تحذيرات "فشل استعلام) يعيد أخطاء أو ينتهي الوقت HSS")
- HSS إنتاج تحويل الرقم تنسيق رقم مختلف عما يتوقعه

الحل:

1. تحقق من `diameter_enabled: true` في `runtime.exs`
2. عند بدء التشغيل Diameter تحقق من السجلات لحالة اتصال الند
3. في سجل التطبيق "HSS ابحث عن تحذيرات "فشل استعلام
4. تأكد من أن تنسيق الرقم المدعو بعد التحويل يتطابق مع ما يتوقعه HSS (E.164 بدون +)

الرسائل مخزنة ولكن لم يتم تسليمها أبدًا

الأعراض: يتم تخزين الرسائل بشكل صحيح لمشتركين على الشبكة ولكن لم يتم تسليمها، حتى بعد تسجيل المشترك.

الأسباب المحتملة:

- تحديثات الموقع لا تصل إلى OmniMessage
- `destination_msisdn` في تسجيل الموقع لا يتطابق مع MSISDN للرسالة المخزنة
- انتهت صلاحية الرسائل وتم نقلها إلى قائمة الرسائل الميتة

الحل:

1. تحقق من أن تحديثات الموقع/التسجيل ��تم استلامها (تحقق من صفحة التسجيلات في لوحة التحكم)
2. تأكد من أن تنسيق MSISDN متسق بين قائمة انتظار الرسائل وجدول التسجيل
3. تحقق من `dead_letter_time_minutes` — قم بزيادته إذا كان المشتركون قد يكونون غير متصلين لفترات طويلة

عدم اتصال الند Diameter

الأعراض: تظهر السجلات عدم وجود اتصالات أقران Diameter عند بدء التشغيل.

الأسباب المحتملة:

- عنوان IP أو منفذ الند غير صحيح في التكوين
- جدار الحماية يمنع المنفذ 3868
- عدم تطابق هوية مضيف الند (يجب أن تتطابق القيمة المكونة لـ `host` تمامًا مع هوية الند Diameter)
- عدم تطابق المجال بين OmniMessage والند

الحل:

1. تحقق من أن عنوان IP والمنفذ صحيحان للند
2. تأكد من أن قواعد جدار الحماية تسمح بحركة مرور TCP على المنفذ المكون
3. تحقق من أن قيمة `host` للند تتطابق تمامًا مع ما يعلنه الند في تبادل قدراته
4. تأكد من أن `realm` يتطابق بين تكوينات OmniMessage والند

وثائق مقاييس Prometheus لنظام SMS-C

العودة إلى فهرس الوثائق | **الوثيقة الرئيسية ←**

نظرة عامة

تصف هذه الوثيقة جميع مقاييس Prometheus التي يكشف عنها نظام SMS-C. تم تصميم هذه المقاييس لتمكين موظفي العمليات من مراقبة صحة النظام وأدائه وحل المشكلات.

الوصول إلى المقاييس

نقطة نهاية مقاييس Prometheus متاحة على:

```
http://localhost:9568/metrics
```

تقوم هذه النقطة بكشف المقاييس بتنسيق نص Prometheus الذي يمكن أن يقوم خادم Prometheus بجمعه. يتم تحديث المقاييس في الوقت الحقيقي أثناء معالجة النظام للرسائل.

اتفاقية تسمية المقاييس

جميع المقاييس النمط: `sms_c.<category>.<metric_name>.<type>`

الفئات:

- `license` - مقاييس حالة الترخيص
- `message` - مقاييس معالجة الرسائل
- `routing` - مقاييس قرارات التوجيه
- `enum` - مقاييس بحث ENUM/NAPTR

- `delivery` - مقاييس تسليم الرسائل
- `queue` - مقاييس إدارة الطوابير
- `charging` - مقاييس الفوترة/التحصيل
- `mnesia` - مقاييس قاعدة البيانات
- `frontend` - مقاييس اتصال الواجهة الأمامية
- `location` - مقاييس الموقع/التسجيل
- `phoenix.endpoint` - مقاييس طلبات واجهة برمجة التطبيقات HTTP
- `vm` - مقاييس نظام Erlang VM

مقاييس الترخيص

sms_c_license_status

النوع: Gauge

الوصف: حالة الترخيص الحالية لنظام OmniMessage SMS-C.

القيم:

- `1` - ترخيص صالح
- `0` - ترخيص غير صالح/منتهي

التسميات: لا شيء

اسم المنتج: `omnimessage`

حالة الاستخدام: مراقبة صلاحية الترخيص لضمان عمل النظام بترخيص صالح. عند عدم الصلاحية، يتم استقبال الرسائل ولكن يتم توجيهها إلى الوجهة "NOLICENCE" بدلاً من التوجيه العادي.

السلوك عند عدم صلاحية الترخيص:

- يتم **قبول** الرسائل الواردة وتخزينها
- يتم **تعيين** وجهة الرسالة (`dest_smsc`) تلقائيًا إلى "NOLICENCE"
- يتم **تجاوز** التوجيه العادي
- ظل واجهة المستخدم والمراقبة **متاحة**

- تظل قاعدة البيانات وجميع الخدمات **قيد التشغيل**

التنبيه:

```
- alert: SMS_C_License_Invalid
  expr: sms_c_license_status == 0
  for: 1m
  labels:
    severity: critical
  annotations:
    summary: "ترخيص SMS-C غير صالح أو منتهي"
    description: "حالة الترخيص غير صالحة - يتم توجيه الرسائل إلى NOLICENCE"
```

أمثلة استعلامات Prometheus:

```
# تحقق مما إذا كان الترخيص صالحًا
sms_c_license_status == 1

# تنبيه عند الترخيص غير صالح
sms_c_license_status == 0

# عد الرسائل الموجهة إلى NOLICENCE (تشير إلى مشكلة في الترخيص)
sms_c_routing_route_matched_count{dest_smsc="NOLICENCE"}
```

مقاييس معالجة الرسائل

sms_c_message_received_count

النوع: Counter

الوصف: إجمالي عدد الرسائل المستلمة من قبل SMS-C من جميع المصادر.

التسميات:

- `source_smsc`: اسم SMSC المصدر الذي أرسل الرسالة
- `source_type`: نوع اتصال المصدر (ims، circuit_switched، smpp)

- `message_type`: نوع الرسالة (sms، mms)

حالة الاستخدام: مراقبة حجم الرسائل الواردة حسب المصدر والنوع. استخدمها لاكتشاف أنماط الحركة، وتحديد الفترات المزدحمة، ورصد الشذوذ في تدفق الرسائل.

التنبيه: قم بإعداد التنبيهات للانخفاضات المفاجئة (مشكلات محتملة في الاتصال بالمصدر) أو الارتفاعات (هجوم محتمل/رسائل غير مرغوب فيها).

---

sms_c_message_validated_count

النوع: Counter

الوصف: إجمالي عدد عمليات التحقق من الرسائل التي تم تنفيذها.

التسميات:

- `valid`: ما إذا كانت عملية التحقق قد نجحت (صحيح أو خطأ)

حالة الاستخدام: تتبع معدلات نجاح/فشل التحقق. قد تشير معدلات الفشل العالية إلى رسائل مشوهة أو مشكلات في التكامل.

التنبيه: تنبيه عند تجاوز معدل فشل التحقق العتبة (على سبيل المثال، > 5% فشل).

---

sms_c_message_processing_stop_duration

النوع: Histogram

الوصف: الوقت المستغرق لمعالجة رسالة من الاستلام إلى الاكتمال (يشمل التحقق، والتوجيه، والطوابير).

الوحدة: مللي ثانية

الأقسام: 10، 50، 100، 250، 500، 1000، 2500، 5000 مللي ثانية

التسميات:

- `success`: ما إذا كانت المعالجة قد نجحت (صحيح أو خطأ)

حالة الاستخدام: مراقبة أداء معالجة الرسائل من النهاية إلى النهاية. تحديد التباطؤ في خط معالجة الرسائل.

التنبيه: تنبيه عند تجاوز زمن الاستجابة p99 أو p95 عتبات SLA.

مقاييس التوجيه

sms_c_routing_route_matched_count

النوع: Counter

الوصف: إجمالي عدد مرات تطابق مسار معين وتم اختياره لتوجيه الرسائل.

التسميات:

- `route_id`: معرف فريد للمسار المتطابق
- `dest_smsc`: الوجهة المختارة بواسطة المسار SMSC
- `priority`: قيمة الأولوية للمسار المتطابق

حالة الاستخدام: فهم أي المسارات تُستخدم بشكل متكرر. تحديد المسارات غير المستغلة أو المحملة بشكل زائد. مفيد في التخطيط للطاقة وتحسين المسارات.

التنبيه: تنبيه إذا كانت المسارات عالية الأولوية نادرًا ما تتطابق (قد تشير إلى تكوين خاطئ في التوجيه).

---

sms_c_routing_failed_count

النوع: Counter

الوصف: إجمالي عدد حالات فشل التوجيه حيث لم يتم العثور على مسار مناسب.

التسميات:

- `reason`: سبب الفشل (no_route_found، validation_failed، إلخ.)

حالة الاستخدام: تتبع حالات فشل التوجيه لتحديد الفجوات في التكوين أو أنماط الحركة غير المتوقعة.

التنبيه: تنبيه عند أي حالات فشل في التوجيه لأنها تشير إلى أن الرسائل لا يمكن تسليمها.

---

sms_c_routing_action_count

النوع: Counter

الوصف: إجمالي عدد الإجراءات الخاصة بالتوجيه التي تم اتخاذها.

التسميات:

- `action`: نوع الإجراء (drop، auto_reply، forward)
- `route_id`: المسار الذي أدى إلى الإجراء

حالة الاستخدام: مراقبة قواعد الإزالة (مكافحة الرسائل غير المرغوب فيها)، واستخدام الرد التلقائي، وأنماط التوجيه.

التنبيه: تنبيه عند الارتفاعات غير المتوقعة في إجراءات الإزالة (قد تشير إلى هجوم رسائل غير مرغوب فيها).

---

sms_c_routing_stop_duration

النوع: Histogram

الوصف: الوقت المستغرق لتقييم جميع المسارات واختيار أفضل تطابق.

الوحدة: مللي ثانية

الأقسام: 1، 5، 10، 25، 50، 100، 250، 500 مللي ثانية

التسميات:

- `dest_smsc`: الوجهة المختارة SMSC

حالة الاستخدام: مراقبة أداء محرك التوجيه. يشير التوجيه البطيء إلى وجود عدد كبير جدًا من المسارات أو منطق تطابق معقد.

التنبيه: تنبيه عندما يستغرق التوجيه وقتًا أطول من المتوقع باستمرار (على سبيل المثال، p95 > 50 مللي ثانية).

مقاييس بحث ENUM/NAPTR

sms_c_enum_cache_hit_count

النوع: Counter

الوصف: إجمالي عدد عمليات البحث عن ENUM التي تم تقديمها من الذاكرة المؤقتة (لم تتطلب استعلام DNS).

التسميات:

- `domain`: نطاق ENUM الذي تم ال��ستعلام عنه

حالة الاستخدام: مراقبة فعالية الذاكرة المؤقتة. تؤدي معدلات الضرب العالية إلى تقليل الحمل على DNS وتحسين الأداء.

التنبيه: تنبيه إذا انخفض معدل الضرب تحت العتبة (قد تشير إلى مشكلات في الذاكرة المؤقتة أو حركة غير عادية).

---

sms_c_enum_cache_miss_count

النوع: Counter

الوصف: إجمالي عدد عمليات البحث عن ENUM التي تطلبت استعلام DNS (ليست في الذاكرة المؤقتة).

التسميات:

- `domain`: نطاق ENUM الذي تم الاستعلام عنه

حالة الاستخدام: تتبع حالات الفشل في الذاكرة المؤقتة لفهم فعالية الذاكرة المؤقتة. استخدمها مع عدد الضربات لحساب معدل الضرب.

الحساب: `cache_hit_rate = hits / (hits + misses)`

---

sms_c_enum_cache_size_size

النوع: Gauge

الوصف: العدد الحالي للإدخالات في ذاكرة ENUM المؤقتة.

حالة الاستخدام: مراقبة حجم الذاكرة المؤقتة لضمان عدم نموها بشكل غير محدود. يساعد في ضبط إعدادات TTL للذاكرة المؤقتة.

التنبيه: تنبيه إذا تجاوز حجم الذاكرة المؤقتة الحدود المتوقعة (قد تشير إلى تسرب في الذاكرة).

---

sms_c_enum_lookup_stop_duration

النوع: Histogram

الوصف: الوقت المستغرق لإكمال بحث ENUM (بما في ذلك استعلام DNS إذا لم يكن مخزنًا في الذاكرة المؤقتة).

الوحدة: مللي ثانية

الأقسام: 10، 50، 100، 250، 500، 1000، 2500، 5000 مللي ثانية

التسميات:

- `domain`: نطاق ENUM الذي تم الاستعلام عنه
- `success`: (صحيح أو خطأ) ما إذا كانت عملية البحث قد نجحت
- `cache_hit`: (صحيح أو خطأ) ما إذا كانت النتيجة قد تم تقديمها من الذاكرة المؤقتة

حالة الاستخدام: مراقبة أداء بحث ENUM. تحديد خوادم DNS البطيئة أو مشكلات الشبكة.

التنبيه: تنبيه عندما يتجاوز وقت البحث p95 عتبة المهلة.

---

sms_c_enum_naptr_records_record_count

النوع: Histogram

الوصف: عدد سجلات NAPTR التي تم إرجاعها بواسطة بحث ENUM ناجح.

الأقسام: 0، 1، 2، 3، 5، 10

التسميات:

- `domain`: نطاق ENUM الذي تم الاستعلام عنه

حالة الاستخدام: فهم توزيع سجلات ENUM. يجب أن تعيد معظم عمليات البحث 1-3 سجلات.

التنبيه: تنبيه إذا تم إرجاع 0 سجلات بشكل متكرر (مشكلة في تكوين DNS).

مقاييس التسليم

sms_c_delivery_queued_count

النوع: Counter

الوصف: إجمالي عدد الرسائل المعلقة للتسليم إلى الوجهة SMSC.

التسميات:

- `dest_smsc`: اسم الوجهة SMSC

حالة الاستخدام: مراقبة تدفق الرسائل إلى كل وجهة. مفيد في التخطيط للطاقة.

التنبيه: قارن مع عدد نجاح/فشل التسليم لاكتشاف التراكم.

---

sms_c_delivery_attempted_count

النوع: Counter

الوصف: إجمالي عدد محاولات التسليم التي تمت (تشمل المحاولات المتكررة).

التسميات:

- `dest_smsc`: اسم SMSC الوجهة

حالة الاستخدام: تتبع حجم محاولات التسليم. يشير عدد المحاولات المرتفع بالنسبة لعدد الرسائل المعلقة إلى سلوك إعادة المحاولة.

---

sms_c_delivery_succeeded_count

النوع: Counter

الوصف: إجمالي عدد الرسائل التي تم تسليمها بنجاح إلى SMSC الوجهة.

التسميات:

- `dest_smsc`: اسم SMSC الوجهة

حالة الاستخدام: تتبع التسليم الناجح لكل وجهة. المقياس الأساسي للنجاح.

التنبيه: تنبيه إذا انخفض معدل النجاح تحت عتبة SLA.

الحساب: `success_rate = succeeded / queued`

---

sms_c_delivery_failed_count

النوع: Counter

الوصف: إجمالي عدد الرسائل التي فشلت في التسليم بعد جميع محاولات إعادة المحاولة.

التسميات:

- `dest_smsc`: اسم SMSC الوجهة
- `reason`: سبب الفشل

حالة الاستخدام: تتبع حالات فشل التسليم لتحديد الوجهات أو أنماط الفشل المقلقة.

التنبيه: تنبيه على معدلات الفشل المرتفعة أو أسباب الفشل المحددة.

---

sms_c_delivery_dead_letter_count

النوع: Counter

الوصف: إجمالي عدد الرسائل التي تم نقلها إلى قائمة الرسائل الميتة (غير قابلة للتسليم).

التسميات:

- `reason`: سبب الرسالة الميتة (مثل `max_retries_exceeded`، `expired`)

حالة الاستخدام: مراقبة الرسائل غير القابلة للتسليم التي تتطلب تدخلًا يدويًا.

التنبيه: تنبيه على أي أحداث رسائل ميتة لأنها تمثل فشل التسليم الكامل.

---

sms_c_delivery_succeeded_attempt_count

النوع: Histogram

الوصف: عدد محاولات التسليم المطلوبة قبل التسليم الناجح.

الأقسام: 1، 2، 3، 5، 10

التسميات:

- `dest_smsc`: اسم SMSC الوجهة

حالة الاستخدام: فهم سلوك إعادة المحاولة. يجب أن تنجح معظم التسليمات في المحاولة الأولى.

التنبيه: تنبيه إذا تجاوز متوسط عدد المحاولات 2 (يشير إلى مشكلات موثوقية الوجهة).

---

sms_c_delivery_failed_attempt_count

النوع: Histogram

الوصف: عدد محاولات التسليم التي تمت قبل الفشل النهائي.

الأقسام: 1، 2، 3، 5، 10

التسميات:

- `dest_smsc`: اسم SMSC الوجهة

حالة الاستخدام: فهم عدد مرات إعادة المحاولة التي تحدث قبل الاستسلام.

---

sms_c_delivery_time_delta_delta_ms

النوع: Histogram

الوصف: الوقت من تقديم الرسالة إلى تأكيد التسليم. تلتقط هذه المقياس زمن الاستجابة الكامل من لحظة دخول الرسالة إلى SMS-C إلى لحظة تأكيد التسليم، مع تسميات مفصلة للتحليل حسب المصدر، والوجهة، وسلوك إعادة المحاولة.

الوحدة: مللي ثانية

الأقسام: 50، 100، 250، 500، 1000، 2000، 5000، 10000، 20000، 60000، 600000 مللي ثانية (من 50 مللي ثانية إلى 10 دقائق)

التسميات:

- `source_smsc`: المصدر الذي قدم الرسالة SMSC
- `dest_smsc`: الوجهة التي سلمت الرسالة SMSC
- `delivery_attempts`: عدد محاولات التسليم المطلوبة (0 = نجاح المحاولة الأولى)

حالة الاستخدام: تحليل أداء التسليم من النهاية إلى النهاية حسب زوج المصدر-الوجهة. تحديد المسارات البطيئة، وتركيبات SMSC المقلقة، ومطابقة زمن الاستجابة مع سلوك إعادة المحاولة. ضروري لمراقبة SLA والتخطيط للطاقة.

التنبيه:

- تنبيه عندما يتجاوز p95 عتبة SLA لمجموعات المصدر/الوجهة المحددة
- تنبيه عندما تتطلب الرسائل بشكل مستمر محاولات متعددة

أمثلة استعلامات Prometheus:

```
# زمن التسليم p95 حسب SMSC المصدر والوجهة
histogram_quantile(0.95,
  sum by (source_smsc, dest_smsc, le) (
    rate(sms_c_delivery_time_delta_delta_ms_bucket[5m])
  )
)

# زمن التسليم p99 بشكل عام
histogram_quantile(0.99,
  sum by (le) (
    rate(sms_c_delivery_time_delta_delta_ms_bucket[5m])
  )
)

# متوسط زمن التسليم حسب SMSC المصدر
sum by (source_smsc) (rate(sms_c_delivery_time_delta_delta_ms_sum[5m]
/
sum by (source_smsc)
(rate(sms_c_delivery_time_delta_delta_ms_count[5m]))

# زمن التسليم للرسائل التي تتطلب إعادة المحاولة مقابل نجاح المحاولة
الأولى
histogram_quantile(0.95,
  sum by (le)
(rate(sms_c_delivery_time_delta_delta_ms_bucket{delivery_attempts="0"
[5m]))
)
histogram_quantile(0.95,
  sum by (le)
(rate(sms_c_delivery_time_delta_delta_ms_bucket{delivery_attempts!="0
[5m]))
)

# أبطأ أز��اج المصدر-الوجهة (حسب p95)
topk(10,
  histogram_quantile(0.95,
    sum by (source_smsc, dest_smsc, le) (
      rate(sms_c_delivery_time_delta_delta_ms_bucket[1h])
    )
  )
)

# نسبة الرسائل التي تم تسليمها في غضون ثانيتين
```

```
sum(rate(sms_c_delivery_time_delta_delta_ms_bucket{le="2000"}[5m])) /
sum(rate(sms_c_delivery_time_delta_delta_ms_count[5m])) * 100

# الرسائل التي تم تسليمها في غضون ثانيتين حسب الوجهة
sum by (dest_smsc)
(rate(sms_c_delivery_time_delta_delta_ms_bucket{le="2000"}[5m])) /
sum by (dest_smsc) (rate(sms_c_delivery_time_delta_delta_ms_count[5m]
* 100
```

مقاييس الطابور

sms_c_queue_size_size

النوع: Gauge

الوصف: العدد الإجمالي الحالي للرسائل في الطابور (جميع الحالات مجتمعة).

التسميات:

- `queue_type`: نوع الطابور (message_queue، dead_letter)

حالة الاستخدام: مراقبة عمق الطابور لاكتشاف التراكمات أو مشكلات المعالجة.

التنبيه: تنبيه عندما يتجاوز حجم الطابور الحدود السعوية.

---

sms_c_queue_size_pending

النوع: Gauge

الوصف: العدد الحالي للرسائل المعلقة للتسليم (لم يتم المحاولة بعد).

التسميات:

- `queue_type`: نوع الطابور

حالة الاستخدام: مراقبة عدد الرسائل المعلقة. تشير الأعداد المرتفعة المعلقة إلى تأخيرات في المعالجة.

التنبيه: تنبيه عندما يتجاوز العدد المعلق العتبة لفترة ممتدة.

---

sms_c_queue_size_failed

النوع: Gauge

الوصف: العدد الحالي للرسائل في حالة الفشل (في انتظار إعادة المحاولة).

التسميات:

- `queue_type`: نوع الطابور

حالة الاستخدام: مراقبة تراكم الرسائل الفاشلة. تشير إلى مشكلات التسليم.

التنبيه: تنبيه على العدد المرتفع من الفشل لأنه يؤثر على معدلات التسليم.

---

sms_c_queue_size_delivered

النوع: Gauge

الوصف: العدد الحالي للرسائل التي تم تسليمها والتي في انتظار التنظيف/الإزالة من الطابور.

التسميات:

- `queue_type`: نوع الطابور

حالة الاستخدام: مراقبة تأخر التنظيف. تشير الأعداد المرتفعة إلى أن عملية التنظيف تتأخر.

التنبيه: تنبيه إذا تراكمت الرسائل المرسلة بشكل كبير.

---

sms_c_queue_oldest_message_age_seconds

النوع: Gauge

الوصف: عمر (بالثواني) لأقدم رسالة حاليًا في حالة الانتظار.

التسميات:

- `queue_type`: نوع الطابور

حالة الاستخدام: اكتشاف تقدم الرسائل وتوقف المعالجة. حرج لمراقبة SLA.

التنبيه: تنبيه عندما يتجاوز عمر الرسالة الأقدم عتبة SLA (على سبيل المثال، > 300 ثانية).

مقاييس التحصيل

sms_c_charging_requested_count

النوع: Counter

الوصف: إجمالي عدد طلبات التحصيل/الفوترة المقدمة إلى OCS أو نظام الفوترة.

التسميات:

- `account`: معرف الحساب الذي يتم تحصيله

حالة الاستخدام: تتبع حجم التحصيل لكل حساب. مفيد لمطابقة الفواتير.

---

sms_c_charging_succeeded_count

النوع: Counter

الوصف: إجمالي عدد عمليات التحصيل الناجحة.

التسميات:

- `account`: معرف الحساب الذي تم تحصيله

حالة الاستخدام: مراقبة معدل نجاح التحصيل لكل حساب.

الحساب: `success_rate = succeeded / requested`

---

sms_c_charging_failed_count

النوع: Counter

الوصف: إجمالي عدد عمليات التحصيل الفاشلة.

التسميات:

- `account`: معرف الحساب
- `reason`: سبب الفشل

حالة الاستخدام: تحديد حالات فشل التحصيل التي قد تؤثر على الإيرادات أو تتطلب تدخل الحساب.

التنبيه: تنبيه على معدلات فشل التحصيل المرتفعة.

---

sms_c_charging_succeeded_duration

النوع: Histogram

الوصف: الوقت المستغرق لإكمال طلب تحصيل ناجح.

الوحدة: مللي ثانية

الأقسام: 10، 50، 100، 250، 500، 1000، 2500، 5000 مللي ثانية

التسميات:

- `account`: معرف الحساب

حالة الاستخدام: مراقبة أداء نظام الفوترة. يمكن أن يؤدي التحصيل البطيء إلى تأخير تسليم الرسائل.

التنبيه: تنبيه عندما يتجاوز زمن التحصيل p95 العتبة.

مقاييس صحة النظام

sms_c_mnesia_table_size_record_count

النوع: Gauge

الوصف: العدد الحالي للسجلات في كل جدول قاعدة بيانات Mnesia. يتم الاستطلاع كل 10 ثوانٍ.

التسميات:

- `table`: اسم الجدول

الجداول المتعقبة:

- `sms_route` - قواعد توجيه SMS
- `message_store` - طابور الرسائل (الرسائل المعلقة، المرسلة، الفاشلة)
- `location_store` - بيانات موقع/تسجيل المشتركين
- `frontend_store` - تسجيلات الواجهة الأمامية/SMSC
- `translation_rule` - عدد قواعد ترجمة الأرقام
- `cell_tower_store` - بيانات موقع برج الخلايا
- `message_events` - سجلات أحداث الرسائل

حالة الاستخدام: مراقبة نمو قاعدة البيانات. اكتشاف تراكم البيانات غير المتوقع. التخطيط للطاقة.

التنبيه: تنبيه على معدلات نمو الجدول غير المتوقعة أو عند الاقتراب من حدود السعة.

أمثلة استعلامات Prometheus:

```
# جميع أحجام الجداول
sms_c_mnesia_table_size_record_count

# حجم طابور الرسائل
sms_c_mnesia_table_size_record_count{table="message_store"}

# تسجيلات المشتركين النشطة
sms_c_mnesia_table_size_record_count{table="location_store"}

# عدد قواعد التوجيه
sms_c_mnesia_table_size_record_count{table="sms_route"}
```

---

sms_c_frontend_status_count

النوع: Gauge

الوصف: عدد الواجهات الأمامية في كل حالة اتصال.

التسميات:

- `frontend_name`: معرف الواجهة الأمامية
- `status`: حالة الاتصال (متصل، غير متصل)

حالة الاستخدام: مراقبة اتصال الواجهة الأمامية. اكتشاف حالات فشل الاتصال.

التنبيه: تنبيه عندما تنفصل الواجهات الأمامية المتوقعة.

---

sms_c_location_registered_count

النوع: Counter

الوصف: إجمالي عدد تسجيلات الموقع/المشتركين التي استقبلها النظام.

التسميات:

- `location`: اسم الواجهة الأمامية/SMSC حيث تم تسجيل المشترك
- `ims_capable`: ما إذا كان المشترك يدعم IMS (صحيح/خطأ)

مقابل غير IMS **حالة الاستخدام**: مراقبة نشاط تسجيل المشتركين. تتبع المشتركين القابلين لـ القابلين. اكتشاف عواصف التسجيل أو الفشل.

التنبيه: إعداد التنبيهات لـ:

- انخفاض معدلات التسجيل (قد تشير إلى مشكلات في الشبكة)
- ارتفاعات غير عادية في التسجيلات
- نسبة عالية من التسجيلات غير القابلة لـ IMS (تدفق الأجهزة القديمة)

استعلام مثال:

```
# معدل ا��تسجيل في الدقيقة
rate(sms_c_location_registered_count[1m])

# نسبة تسجيل IMS مقابل غير IMS
sum(rate(sms_c_location_registered_count{ims_capable="true"}[5m]))
/
sum(rate(sms_c_location_registered_count[5m]))
```

---

sms_c_location_active_registrations_count

النوع: Gauge

الوصف: العدد الحالي لتسجيلات المشتركين النشطة، مجمعة حسب الموقع (واجهة أمامية/SMSC) وقدرة IMS. يتم استطلاع هذه المقياس كل 10 ثوانٍ وتعكس الحالة الحالية لمخزن الموقع.

التسميات:

- `location`: اسم الواجهة الأمامية/SMSC حيث تم تسجيل المشتركين
- `ims_capable`: ما إذا كانت التسجيلات قابلة لـ IMS (صحيح/خطأ)

حالة الاستخدام: مراقبة توزيع المشتركين الحالي عبر الواجهات الأمامية. تتبع اعتماد IMS. تحديد عدم التوازن في الحمل بين الواجهات الأمامية. التخطيط للطاقة لكل مثيل SMSC.

التنبيه: إعداد التنبيهات لـ:

- الانخفاضات المفاجئة في التسجيلات لموقع (قد تشير إلى مشكلات في الواجهة الأمامية)
- ت��زيع غير متوازن عبر الواجهات الأمامية
- اقتراب إجمالي التسجيلات من حدود السعة

أمثلة استعلامات Prometheus:

```
# إجمالي التسجيلات النشطة
sum(sms_c_location_active_registrations_count)

# التسجيلات النشطة حسب الموقع
sum by (location) (sms_c_location_active_registrations_count)

# حسب الموقع IMS التسجيلات النشطة القابلة لـ
sum by (location)
(sms_c_location_active_registrations_count{ims_capable="true"})

# معدل اعتماد IMS لكل موقع
sum by (location)
(sms_c_location_active_registrations_count{ims_capable="true"}) /
sum by (location) (sms_c_location_active_registrations_count) *
100

# إجمالي تسجيلات IMS مقابل غير IMS
sum by (ims_capable) (sms_c_location_active_registrations_count)

# المواقع ذات أكبر عدد من التسجيلات
topk(10, sum by (location)
(sms_c_location_active_registrations_count))

# توزيع التسجيلات كنسبة من الإجمالي
sum by (location) (sms_c_location_active_registrations_count) /
sum(sms_c_location_active_registrations_count) * 100
```

مقاييس طلبات واجهة برمجة التطبيقات HTTP

phoenix_endpoint_stop_duration

النوع: Distribution (Histogram)

الوصف: مدة معالجة طلب HTTP بالمللي ثانية، من بدء الطلب إلى اكتمال الاستجابة.

التسميات:

- `route`: مسار نقطة نهاية واجهة برمجة التطبيقات (مثل `/api/messages`، `/api/frontends`)

الأقسام: 10 مللي ثانية، 50 مللي ثانية، 100 مللي ثانية، 250 مللي ثانية، 500 مللي ثانية، 1 ثانية، 2.5 ثانية، 5 ثوانٍ

حالة الاستخدام: مراقبة أداء واجهة برمجة التطبيقات. تحديد نقاط النهاية البطيئة. تتبع زمن الاستجابة SLAs.

التنبيه: إعداد التنبيهات لـ:

- زمن الاستجابة P95 > 500 مللي ثانية لنقاط النهاية الحرجة
- زمن الاستجابة P99 > 1 ثانية لأي نقطة نهاية
- اتجاهات زيادة زمن الاستجابة

استعلام مثال:

```
# زمن الاستجابة P95 حسب نقطة النهاية
histogram_quantile(0.95,
  rate(phoenix_endpoint_stop_duration_bucket[5m]))

# الطلبات التي تستغرق أكثر من ثانية واحدة
sum(rate(phoenix_endpoint_stop_duration_bucket{le="1000"}[5m]))
```

---

phoenix_endpoint_stop_count

النوع: Counter

الوصف: إجمالي عدد طلبات HTTP المكتملة، مصنفة حسب المسار ورمز الحالة HTTP.

التسميات:

- `route`: مسار نقطة نهاية واجهة برمجة التطبيقات
- `status`: رمز الحالة HTTP (200، 201، 400، 404، 500، إلخ.)

حالة الاستخدام: مراقبة حجم طلبات واجهة برمجة التطبيقات ومعدلات النجاح. تتبع معدلات الأخطاء حسب نقطة النهاية.

التنبيه: إعداد التنبيهات لـ:

- معدل الأخطاء > 5% لأي نقطة نهاية
- أخطاء 5xx على نقاط النهاية الحرجة
- انخفاضات مفاجئة في حجم الطلبات

استعلام مثال:

```
# معدل الطلبات لكل نقطة نهاية
sum by (route) (rate(phoenix_endpoint_stop_count[5m]))

# معدل الأخطاء حسب نقطة النهاية
sum by (route) (rate(phoenix_endpoint_stop_count{status=~"5.."}
[5m])) /
sum by (route) (rate(phoenix_endpoint_stop_count[5m]))

# معدل النجاح
sum(rate(phoenix_endpoint_stop_count{status=~"2.."}[5m])) /
sum(rate(phoenix_endpoint_stop_count[5m]))
```

---

phoenix_router_dispatch_exception_count

النوع: Counter

الوصف: إجمالي عدد الاستثناءات/الأخطاء التي تم رفعها أثناء معالجة طلب HTTP.

التسميات:

- `route`: مسار نقطة النهاية API حيث حدث الاستثناء
- `kind`: نوع الاستثناء (خطأ، خروج، رمي)

حالة الاستخدام: تتبع أخطاء التطبيق. تحديد نقاط النهاية المقلقة. مراقبة استقرار النظام.

التنبيه: إعداد التنبيهات لأي قيمة غير صفرية على نقاط النهاية الحرجة.

استعلام مثال:

```
# معدل الاستثناءات حسب نقطة النهاية
rate(phoenix_router_dispatch_exception_count[5m])

# إجمالي الاستثناءات في الساعة الماضية
increase(phoenix_router_dispatch_exception_count[1h])
```

مقاييس Erlang VM

vm_memory_total

النوع: Gauge

الوصف: إجمالي الذاكرة المخصصة بواسطة Erlang VM بالبايت.

حالة الاستخدام: مراقبة الاستخدام العام للذاكرة. اكتشاف تسرب الذاكرة. التخطيط للطاقة.

التنبيه: تنبيه عندما يتجاوز استخدام الذاكرة > 80% من الذاكرة ��لمتاحة للنظام.

---

vm_memory_processes

النوع: Gauge

الوصف: الذاكرة المستخدمة بواسطة عمليات Erlang بالبايت.

حالة الاستخدام: تتبع استهلاك الذاكرة للعمليات. المصدر الأكثر شيوعًا لنمو الذاكرة.

التنبيه: تنبيه على معدل النمو المرتفع المستمر.

---

vm_total_run_queue_lengths_total

النوع: Gauge

الوصف: العدد الإجمالي للعمليات التي تنتظر أن يتم جدولتها عبر جميع جدولة CPU.

حالة الاستخدام: قياس حمل النظام. تشير القيم العالية إلى تشبع وحدة المعالجة المركزية.

التنبيه: تنبيه عندما تكون القيمة باستمرار > 10 * عدد نوى CPU.

---

vm_system_counts_process_count

النوع: Gauge

الوصف: العدد الحالي للعمليات التي تعمل في VM.

حالة الاستخدام: مراقبة أنماط إنشاء العمليات. اكتشاف تسرب العمليات.

التنبيه: تنبيه عند الاقتراب من حد العمليات (الافتراضي 262,144).

جمع المقاييس والاستطلاع

يجمع النظام تلقائيًا المقاييس التالية كل 10 ثوانٍ:

- أحجام وأعمار الطوابير
- أحجام جداول Mnesia
- إحصائيات ذاكرة ENUM

جميع المقاييس الأخرى مدفوعة بالأحداث وتصدر عندما تحدث الإجراء المقابل.

أنماط المراقبة الشائعة

معدل نجاح التسليم حسب الوجهة

تتبع معدل نجاح تسليم الرسائل لكل SMSC وجهة:

الصيغة: `(sms_c_delivery_succeeded_count) / (sms_c_delivery_queued_count)`

التفسير: يجب أن تكون > 95% للوجهات الصحية. تشير المعدلات المنخفضة إلى مشكلات في التسليم.

زمن الاستجابة للرسائل من النهاية إلى النهاية

راقب الوقت الإجمالي من استلام الرسالة إلى التسليم:

المقاييس:

- `sms_c_message_processing_stop_duration` (المعالجة)
- `sms_c_delivery_time_delta_delta_ms` (التسليم)

التفسير: يمثل المجموع زمن الاستجابة الكلي الذي يواجهه المستخدم.

فعالية ذاكرة ENUM

قياس مدى أدا�ة ذاكرة ENUM:

الصيغة: `(sms_c_enum_cache_hit_count) / (sms_c_enum_cache_hit_count + sms_c_enum_cache_miss_count)`

التفسير: يجب أن تكون > 80% بعد التسخين. قد تشير المعدلات المنخفضة إلى TTL قصير أو تباين حركة عالي.

استخدام المسار

تحديد أي المسارات تتعامل مع أكبر قدر من الحركة:

المقياس: `sms_c_routing_route_matched_count` مجمعة حسب `route_id`

التفسير: استخدم لتحديد المسارات الساخنة للتحسين والتخطيط للطاقة.

اتجاه تراكم الطابور

راقب ما إذا كان طابور الرسائل ينمو (تراكم) أو يتقلص (يتعقب):

المقاييس:

- `sms_c_queue_size_pending` (الحالي المعلق)
- `sms_c_queue_oldest_message_age_seconds` (اتجاه العمر)

التفسير: عدد المعلق المتزايد + العمر المتزايد = تشكيل تراكم.

معدل إعادة المحاولة

فهم مدى تكرار الحاجة إلى إعادة المحاولات للتسليم:

المقياس: `sms_c_delivery_succeeded_attempt_count` النسب المئوية

التفسير: إذا كان p95 > 1، فإن معظم الرسائل تتطلب إعادة المحاولة. يشير إلى مشكلات موثوقية الوجهة.

التنبيهات الموصى بها

التنبيه	الشرط	الشدة	وصف
معدل فشل التوجيه المرتفع	زيادة `routing_failed_count`	حرجة	لا يمكن توجيه لرسائل
تراكم الطابور	`queue_size_pending` > العتبة	تحذير	تراكم لرسائل
الرسائل القديمة في الطابور	`queue_oldest_message_age_seconds` > 300	حرجة	انتهاك SLA
ارتفاع فشل التسليم	ارتفاع `delivery_failed_count`	عالي	شكلات الوجهة
أحداث الرسائل الميتة	`delivery_dead_letter_count` > 0	عالي	رسائل غير قابلة لتسليم
مهلات بحث ENUM	`enum_lookup_stop_duration` p95 > 5000 مللي ثانية	تحذير	شكلات DNS
انخفاض معدل الضرب في الذاكرة المؤقتة	معدل الضرب في ذاكرة ENUM < 0.7	تحذير	الذاكرة المؤقتة غير فعالة

التنبيه	الشرط	الشدة	وصف
الواجهة الأمامية غير متصلة	`frontend_status_count{status="disconnected"}` > 0	عالي	فقدان الاتصال
فشل التحصيل	`charging_failed_count` > العتبة	عالي	شكلات الفوترة
معالجة الرسائل البطيئة	`message_processing_stop_duration` p95 > 1000 مللي ثانية	تحذير	تدهور الأداء

توصيات لوحة المعلومات

لوحة معلومات العمليات

الغرض: مراقبة صحة النظام في الوقت الحقيقي

الألواح:

1. تدفق الرسائل (المستلمة/المعالجة/المُرسلة في الدقيقة)
2. أحجام الطوابير (المعلقة، الفاشلة، المرسلة)
3. معدل نجاح التسليم حسب الوجهة
4. زمن الاستجابة p95 للمعالجة والتسليم
5. حالة الواجهات الأمامية النشطة
6. التنبيهات الحالية

لوحة معلومات الأداء

الغرض: تحليل أداء النظام

الألواح:

1. هيستوجرام مدة معالجة الرسائل
2. هيستوجرام مدة التوجيه
3. هيستوجرام مدة بحث ENUM
4. هيستوجرام مدة التحصيل
5. توزيع محاولات التسليم
6. معدلات الضرب في الذاكرة المؤقتة

لوحة معلومات الأعمال

الغ��ض: تحليل الحركة والاستخدام

الألواح:

1. المصدر SMSC الرسائل حسب
2. الوجهة SMSC الرسائل حسب
3. خريطة استخدام المسار
4. عدد الردود التلقائية وإجراءات الإزالة
5. إحصائيات استخدام ENUM
6. حجم التحصيل حسب الحساب

الاحتفاظ بالمقاييس

إعدادات الاحتفاظ الموصى بها لـ Prometheus:

- **المقاييس الخام**: 15 يومًا
- **التجميعات لمدة 5 دقائق**: 90 يومًا
- **التجميعات لمدة ساعة واحدة**: سنتان

يوفر هذا تاريخًا تفصيليًا حديثًا مع الحفاظ على الاتجاهات طويلة الأمد للتخطيط للطاقة.

استكشاف الأخطاء باستخدام المقاييس

السيناريو: عدم تسليم الرسائل

خطوات التحقيق:

1. هل يتم استلام الرسائل؟ - تحقق من `sms_c_message_received_count`
2. هل يتم توجيهها؟ - تحقق من `sms_c_routing_failed_count`
3. هل يتم وضعها في الطابور؟ - تحقق من `sms_c_delivery_queued_count`
4. هل تفشل محاولات التسليم؟ - تحقق من `sms_c_delivery_failed_count`
5. تحقق من تسميات `dest_smsc` لتحديد الوجهة المقلقة

السيناريو: معالجة الرسائل ببطء

خطوات التحقيق:

1. الوقت - تحقق من هيستوجرام `sms_c_message_processing_stop_duration` الإجمالي للمعالجة
2. هل التوجيه بطيء؟ - تحقق من `sms_c_routing_stop_duration`
3. هل عمليات بحث ENUM - تحقق من `sms_c_enum_lookup_stop_duration` بطيئة؟
4. هل التحصيل بطيء؟ - تحقق من `sms_c_charging_succeeded_duration`
5. حدد عنق الزجاجة واستقصاء المكون المحدد

السيناريو: نمو طابور الرسائل

خطوات التحقيق:

1. هل ينمو؟ - تحقق من اتجاه `sms_c_queue_size_pending`
2. هل تحدث محاولات التسليم؟ - تحقق من `sms_c_delivery_attempted_count`
3. هل تفشل؟ - تحقق من `sms_c_delivery_failed_count`
4. هل يستغرق التسليم وقتًا - تحقق من `sms_c_delivery_time_delta_delta_ms` طويلاً؟

5. تحقق من تسميات `dest_smsc` لتحديد الوجهات البطيئة

أمثلة استعلامات Prometheus

تدفق الرسائل

الرسائل المستلمة في الثانية (متوسط 5 دقائق):

```
rate(sms_c_message_received_count[5m])
```

الرسائل المستلمة في الدقيقة (متوسط ساعة واحدة):

```
rate(sms_c_message_received_count[1h]) * 60
```

إجمالي الرسائل اليوم:

```
increase(sms_c_message_received_count[24h])
```

الرسائل حسب نوع المصدر:

```
sum by (source_type) (rate(sms_c_message_received_count[5m]))
```

الرسائل حسب SMSC المصدر:

```
sum by (source_smsc) (rate(sms_c_message_received_count[5m]))
```

أداء التسليم

معدل نجاح التسليم (النسبة المئوية):

```
(rate(sms_c_delivery_succeeded_count[5m]) /
rate(sms_c_delivery_queued_count[5m])) * 100
```

معدل فشل التسليم (النسبة المئوية):

```
(rate(sms_c_delivery_failed_count[5m]) /
rate(sms_c_delivery_queued_count[5m])) * 100
```

متوسط محاولات التسليم (p95):

```
histogram_quantile(0.95,
sms_c_delivery_succeeded_attempt_count_bucket)
```

نجاح التسليم حسب الوجهة:

```
sum by (dest_smsc) (rate(sms_c_delivery_succeeded_count[5m]))
```

أسباب فشل التسليم:

```
sum by (reason) (rate(sms_c_delivery_failed_count[5m]))
```

زمن التسليم (p95):

```
histogram_quantile(0.95,
sms_c_delivery_time_delta_delta_ms_bucket)
```

زمن التسليم (p99):

```
histogram_quantile(0.99,
sms_c_delivery_time_delta_delta_ms_bucket)
```

مقاييس الطابور

الرسائل المعلقة الحالية:

```
sms_c_queue_size_pending
```

الرسائل الفاشلة التي تنتظر إعادة المحاولة:

```
sms_c_queue_size_failed
```

عمر أقدم رسالة (بالدقائق):

```
sms_c_queue_oldest_message_age_seconds / 60
```

معدل نمو الطابور (رسائل/ساعة):

```
rate(sms_c_queue_size_size[1h]) * 3600
```

الرسائل التي تدخل الطابور:

```
rate(sms_c_delivery_queued_count[5m])
```

الرسائل التي تخرج من الطابور:

```
rate(sms_c_delivery_succeeded_count[5m]) +
rate(sms_c_delivery_failed_count[5m])
```

تراكم الطابور (الدخول - الخروج):

```
rate(sms_c_delivery_queued_count[5m]) -
(rate(sms_c_delivery_succeeded_count[5m]) +
rate(sms_c_delivery_failed_count[5m]))
```

أداء التوجيه

معدل نجاح التوجيه:

```
(1 - (rate(sms_c_routing_failed_count[5m]) /
(rate(sms_c_routing_route_matched_count[5m]) +
rate(sms_c_routing_failed_count[5m])))) * 100
```

أكثر المسارات استخدامًا:

```
topk(10, sum by (route_id, dest_smsc)
(rate(sms_c_routing_route_matched_count[1h])))
```

زمن التوجيه (p50، p95، p99):

```
histogram_quantile(0.50, sms_c_routing_stop_duration_bucket)
histogram_quantile(0.95, sms_c_routing_stop_duration_bucket)
histogram_quantile(0.99, sms_c_routing_stop_duration_bucket)
```

حالات فشل التوجيه في الدقيقة:

```
rate(sms_c_routing_failed_count[5m]) * 60
```

إجراءات الإزالة في الساعة:

```
increase(sms_c_routing_action_count{action="drop"}[1h])
```

إجراءات الرد التلقائي في الساعة:

```
increase(sms_c_routing_action_count{action="auto_reply"}[1h])
```

أداء ENUM

معدل الضرب في ذاكرة ENUM:

```
rate(sms_c_enum_cache_hit_count[5m]) /
(rate(sms_c_enum_cache_hit_count[5m]) +
rate(sms_c_enum_cache_miss_count[5m]))
```

نسبة الضرب في ذاكرة ENUM:

```
(rate(sms_c_enum_cache_hit_count[5m]) /
(rate(sms_c_enum_cache_hit_count[5m]) +
rate(sms_c_enum_cache_miss_count[5m])))) * 100
```

زمن البحث عن ENUM (p95):

```
histogram_quantile(0.95, sms_c_enum_lookup_stop_duration_bucket)
```

عمليات بحث ENUM (مخزنة مقابل غير مخزنة) في الثانية:

```
# (سريعة) مخزنة
rate(sms_c_enum_cache_hit_count[5m])

# غير مخزنة (تتطلب استعلام DNS)
rate(sms_c_enum_cache_miss_count[5m])
```

متوسط عدد سجلات NAPTR المعادة:

```
rate(sms_c_enum_naptr_records_record_count_sum[5m]) /
rate(sms_c_enum_naptr_records_record_count_count[5m])
```

حجم ذاكرة ENUM:

```
sms_c_enum_cache_size_size
```

أداء المعالجة

زمن معالجة الرسائل (p95):

```
histogram_quantile(0.95,
sms_c_message_processing_stop_duration_bucket)
```

زمن معالجة الرسائل (p99):

```
histogram_quantile(0.99,
sms_c_message_processing_stop_duration_bucket)
```

حالات الفشل في المعالجة:

```
rate(sms_c_message_processing_stop_duration_count{success="false"}
[5m])
```

معدل فشل التحقق:

```
rate(sms_c_message_validated_count{valid="false"}[5m]) /
rate(sms_c_message_validated_count[5m])
```

مقاييس التحصيل

معدل نجاح التحصيل:

```
rate(sms_c_charging_succeeded_count[5m]) /
rate(sms_c_charging_requested_count[5m])
```

حالات فشل التحصيل في الدقيقة:

```
rate(sms_c_charging_failed_count[5m]) * 60
```

زمن التحصيل (p95):

```
histogram_quantile(0.95, sms_c_charging_succeeded_duration_bucket)
```

حجم التحصيل حسب الحساب:

```
sum by (account) (rate(sms_c_charging_requested_count[1h]))
```

صحة الواجهة الأمامية

الواجهات الأمامية النشطة:

```
sum(sms_c_frontend_status_count{status="connected"})
```

الواجهات الأمامية غير المتصلة:

```
sum(sms_c_frontend_status_count{status="disconnected"})
```

الواجهات الأمامية حسب الاسم:

```
sum by (frontend_name)
(sms_c_frontend_status_count{status="connected"})
```

صحة النظام

أحجام جداول Mnesia:

```
sms_c_mnesia_table_size_record_count
```

عدد المسارات:

```
sms_c_mnesia_table_size_record_count{table="sms_route"}
```

عدد قواعد الترجمة:

```
sms_c_mnesia_table_size_record_count{table="translation_rule"}
```

أمثلة لوحات المعلومات في Grafana

لوحة المعلومات 1: العمليات في الوقت الحقيقي

الغرض: مراقبة النشاط الصحي للنظام الحالي.

الألواح:

1. **تدفق الرسائل (الرسم البياني)**
   - الاستعلام: `rate(sms_c_message_received_count[5m])`
   - الاستعلام: `rate(sms_c_delivery_succeeded_count[5m])`
   - الوحدة: الرسائل/الثانية
   - الأسطورة: {{source_type}}
2. **معدل نجاح التسليم (Gauge)**
   - الاستعلام: `(rate(sms_c_delivery_succeeded_count[5m]) / rate(sms_c_delivery_queued_count[5m])) * 100`
   - الوحدة: النسبة المئوية (0-100)
   - العتبات:
     - الأحمر: < 90
     - الأصفر: 90-95
     - الأخضر: > 95
3. **عمق الطابور (الرسم البياني)**
   - الاستعلام: `sms_c_queue_size_pending`
   - الاستعلام: `sms_c_queue_size_failed`
   - الوحدة: الرسائل
   - الأسطورة: {{queue_type}}
4. **عمر أقدم رسالة (Stat)**
   - الاستعلام: `sms_c_queue_oldest_message_age_seconds / 60`
   - الوحدة: الدقائق
   - العتبات:

        - الأخضر: < 5
        - الأصفر: 5-10
        - الأحمر: > 10

5. **الواجهات الأمامية النشطة (Stat)**
    - الاستعلام: `sum(sms_c_frontend_status_count{status="connected"})`
    - الوحدة: العدد
    - اللون: الأزرق

6. **حال��ت فشل التوجيه (الرسم البياني)**
    - الاستعلام: `rate(sms_c_routing_failed_count[5m]) * 60`
    - الوحدة: الفشل/الدقيقة
    - عتبة التنبيه: > 0

لوحة المعلومات 2: تحليل الأداء

الغرض: تحليل أداء النظام وتحديد عنق الزجاجة.

الألواح:

1. **زمن الاستجابة من النهاية إلى النهاية (الرسم البياني)**
    - الاستعلام: `histogram_quantile(0.50, sms_c_message_processing_stop_duration_bucket)` (p50)
    - الاستعلام: `histogram_quantile(0.95, sms_c_message_processing_stop_duration_bucket)` (p95)
    - الاستعلام: `histogram_quantile(0.99, sms_c_message_processing_stop_duration_bucket)` (p99)
    - الوحدة: الملي ثانية
    - الأسطورة: النسبة المئوية

2. **زمن المكونات (Bar Gauge)**
    - التوجيه: `histogram_quantile(0.95, sms_c_routing_stop_duration_bucket)`

- ENUM: `histogram_quantile(0.95, sms_c_enum_lookup_stop_duration_bucket)`
- التحصيل: `histogram_quantile(0.95, sms_c_charging_succeeded_duration_bucket)`
- التسليم: `histogram_quantile(0.95, sms_c_delivery_time_delta_delta_ms_bucket)`
- الوحدة: المللي ثانية
- الأشرطة الأفق��ة

3. **توزيع محاولات التسليم (Heatmap)**

- الاستعلام: `sms_c_delivery_succeeded_attempt_count_bucket`
- يظهر عدد المحاولات التي يتم عادةً الحاجة إليها
- مقياس الألوان: الأزرق (محاولة واحدة) إلى الأحمر (العديد من المحاولات)

4. **أداء ذاكرة ENUM (الرسم البياني)**

- معدل الضرب: `rate(sms_c_enum_cache_hit_count[5m]) / (rate(sms_c_enum_cache_hit_count[5m]) + rate(sms_c_enum_cache_miss_count[5m]))`
- حجم الذاكرة المؤقتة: `sms_c_enum_cache_size_size`
- مزدوج (معدل مقابل الحجم) Y محور

5. **معدل نجاح المعالجة (Gauge)**

- الاستعلام: `(rate(sms_c_message_processing_stop_duration_count{success="true"}[5m]) / rate(sms_c_message_processing_stop_duration_count[5m])) * 100`
- الوحدة: النسبة المئوية
- العتبات:
  - الأحمر: < 95
  - الأصفر: 95-99
  - الأخضر: > 99

لوحة المعلومات 3: تحليل الحركة

الغرض: تحليل أنماط حركة الرسائل وتوزيع التوجيه.

الألواح:

1. **الرسائل حسب نوع المصدر (Pie Chart)**
   - الاستعلام: `sum by (source_type) (increase(sms_c_message_received_count[1h]))`
   - يظهر التوزيع: SMPP مقابل CS مقابل IMS
2. **الرسائل حسب SMSC المصدر (Bar Chart)**
   - الاستعلام: `sum by (source_smsc) (rate(sms_c_message_received_count[1h]))`
   - أعلى 10 مصادر
   - الأشرطة الأفقية
3. **استخدام المسار (Table)**
   - الأعمدة:
     - معرف المسار
     - الوجهة SMSC
     - الرسائل (1 ساعة): `sum by (route_id, dest_smsc) (increase(sms_c_routing_route_matched_count[1h]))`
     - الأولوية
     - معدل النجاح
   - مرتبة حسب عدد الرسائل
4. **التسليم حسب الوجهة (الرسم البياني)**
   - الاستعلام: `sum by (dest_smsc) (rate(sms_c_delivery_succeeded_count[5m]))`
   - الوحدة: الرسائل/الثانية
   - الرسم البياني المتراكم
   - الأسطورة: {{dest_smsc}}

5. **إجراءات الإزالة/الرد التلقائي (Stat)**
    - تم الإزالة: `increase(sms_c_routing_action_count{action="drop"}[1h])`
    - تم الرد التلقائي: `increase(sms_c_routing_action_count{action="auto_reply"}[1h])`
    - إحصائيات جنبًا إلى جنب
6. **نمط الحركة الساعي (الرسم البياني)**
    - الاستعلام: `rate(sms_c_message_received_count[1h]) * 3600`
    - نطاق الزمن: آخر 7 أيام
    - يظهر الأنماط اليومية

لوحة المعلومات 4: السعة والموارد

الغرض: مراقبة استخدام الموارد وحدود السعة.

الألواح:

1. **سعة الطابور (الرسم البياني)**
    - الحالي: `sms_c_queue_size_size`
    - خط السعة: قيمة ثابتة بناءً على حدود النظام
    - يظهر اتجاه الاستخدام
2. **نمو جدول قاعدة البيانات (الرسم البياني)**
    - الرسائل: `sms_c_mnesia_table_size_record_count{table="sms_route"}`
    - الترجمات: `sms_c_mnesia_table_size_record_count{table="translation_rule"}`
    - الاتجاه على مدى آخر 30 يومًا
3. **اتجاه تراكم الطابور (الرسم البياني)**

- الاستعلام: `rate(sms_c_delivery_queued_count[5m]) - (rate(sms_c_delivery_succeeded_count[5m]) + rate(sms_c_delivery_failed_count[5m]))`
- إيجابي = تراكم ينمو
- سلبي = يتعقب

4. **أقصى حركة (Stat)**

   - الاستعلام: `max_over_time(rate(sms_c_message_received_count[5m])[24h:])`
   - يظهر أعلى معدل 5 دقائق في آخر 24 ساعة
   - الوحدة: الرسائل/الثانية

5. **استخدام السعة (Gauge)**

   - الاستعلام: `(rate(sms_c_message_received_count[5m]) / MAX_CAPACITY) * 100`
   - استبدل MAX_CAPACITY بحدود النظام لديك
   - الوحدة: النسبة المئوية
   - العتبات:
     - الأخضر: < 70
     - الأصفر: 70-85
     - الأحمر: > 85

لوحة المعلومات 5: الامتثال لـ SLA

الغرض: تتبع مقاييس SLA والامتثال.

الألواح:

1. **امتثال SLA (Gauge)**

   - نجاح التسليم: `(rate(sms_c_delivery_succeeded_count[1h]) / rate(sms_c_delivery_queued_count[1h])) * 100`
   - خط الهدف عند 99%
   - العتبات:
     - الأحمر: < 95

- الأصفر: 95-99
- الأخضر: >= 99

2. **الرسائل التي تم تسليمها ضمن SLA (Stat)**

   - الاستعلام: `count(sms_c_delivery_time_delta_delta_ms_bucket{le="5000"}) / count(sms_c_delivery_time_delta_delta_ms_bucket)`
   - يظهر النسبة المئوية التي تم تسليمها في غضون 5 ثوانٍ
   - الوحدة: النسبة المئوية

3. **انتهاكات SLA (Counter)**

   - الرسائل التي تتجاوز 5 دقائق: `increase(sms_c_queue_oldest_message_age_seconds{} > 300)[24h:]`
   - يجب أن تكون 0

4. **الوقت الفعلي (Stat)**

   - الاستعلام: `up{job="sms-c"}`
   - ثنائي: 1 = متصل، 0 = غير متصل
   - يظهر الحالة الحالية

5. **اتجاه معدل النجاح اليومي (الرسم البياني)**

   - الاستعلام: `avg_over_time((rate(sms_c_delivery_succeeded_count[1h]) / rate(sms_c_delivery_queued_count[1h]))[24h:1h])`
   - نطاق الزمن: آخر 30 يومًا
   - خط SLA 99% عند

أمثلة قواعد التنبيه

التنبيهات الحرجة

حالات فشل التوجيه:

```
alert: RoutingFailuresDetected
expr: increase(sms_c_routing_failed_count[5m]) > 0
for: 2m
labels:
  severity: critical
annotations:
  summary: "{{ $value }} حالات فشل في التوجيه في آخر 5 دقائق"
  description: "لا يمكن توجيه الرسائل. تحقق من تكوين التوجيه."
```

تراكم الطابور:

```
alert: MessageQueueBacklog
expr: sms_c_queue_size_pending > 10000
for: 5m
labels:
  severity: critical
annotations:
  summary: "يوجد {{ $value }} رسالة معلقة في الطابور"
  description: "الطابور يتراكم. تحقق من أداء التسليم."
```

الرسائل القديمة في الطابور:

```
alert: OldMessagesInQueue
expr: sms_c_queue_oldest_message_age_seconds > 300
for: 2m
labels:
  severity: critical
annotations:
  summary: "أقدم رسالة عمرها {{ $value }} ثانية"
  description: "الرسائل لا يتم تسليمها. تحقق من الواجهات الأمامية."
```

جميع الواجهات الأمامية غير متصلة:

```
alert: NoActiveFrontends
expr: sum(sms_c_frontend_status_count{status="connected"}) == 0
for: 1m
labels:
  severity: critical
annotations:
  summary: "لا توجد واجهات أمامية متصلة"
  description: "لا يوجد مسار تسليم متاح. تحقق من اتصال الواجهة
الأمامية."
```

نمو قائمة الرسائل الميتة:

```
alert: DeadLetterMessagesIncreasing
expr: rate(sms_c_delivery_dead_letter_count[10m]) > 0
for: 5m
labels:
  severity: critical
annotations:
  summary: "{{ $value }} رسائل تم نقلها إلى قائمة الرسائل الميتة"
  description: "الرسائل تصبح غير قابلة للتسليم. تحقق من الفشل."
```

تنبيهات التحذير

معدل نجاح التسليم المنخفض:

```
alert: LowDeliverySuccessRate
expr: (rate(sms_c_delivery_succeeded_count[10m]) /
rate(sms_c_delivery_queued_count[10m])) < 0.95
for: 10m
labels:
  severity: warning
annotations:
  summary: "معدل نجاح التسليم هو {{ $value | humanizePercentage
}}"
  description: "معدل النجاح أقل من 95%. تحق من حالات فشل
التسليم."
```

معدل إعادة المحاولة المرتفع:

```
alert: HighDeliveryRetryRate
expr: histogram_quantile(0.95,
sms_c_delivery_succeeded_attempt_count_bucket) > 2
for: 15m
labels:
  severity: warning
annotations:
  summary: "95 محاولات التسليم في النسبة المئوية: {{ $value }}"
  description: "الرسائل تتطلب محاولات متعددة. تحقق من موثوقية
الوجهة."
```

معالجة الرسائل البطيئة:

```
alert: SlowMessageProcessing
expr: histogram_quantile(0.95,
sms_c_message_processing_stop_duration_bucket) > 1000
for: 10m
labels:
  severity: warning
annotations:
  summary: "95 زمن المعالجة في النسبة المئوية: {{ $value }} مللي
ثانية"
  description: "معالجة الرسائل بطيئة. تحقق من موارد النظام."
```

عمليات بحث ENUM الفاشلة:

```
alert: HighEnumFailureRate
expr: rate(sms_c_enum_lookup_stop_duration_count{success="false"}
[10m]) > 0.1
for: 10m
labels:
  severity: warning
annotations:
  summary: "معدل فشل بحث ENUM: {{ $value }}"
  description: "عمليات بحث DNS تفشل. تحقق من خوادم DNS."
```

معدل الضرب في ذاكرة ENUM المنخفض:

```
alert: LowEnumCacheHitRate
expr: rate(sms_c_enum_cache_hit_count[10m]) /
(rate(sms_c_enum_cache_hit_count[10m]) +
rate(sms_c_enum_cache_miss_count[10m])) < 0.70
for: 30m
labels:
  severity: warning
annotations:
  summary: "معدل الضرب في ذاكرة ENUM: {{ $value |
humanizePercentage }}"
  description: "كفاءة الذاكرة المؤقتة منخفضة. قد تشير إلى حركة
مرور الأرقام الفريدة."
```

فشل التحصيل:

```
alert: ChargingFailuresDetected
expr: rate(sms_c_charging_failed_count[10m]) > 0.05
for: 10m
labels:
  severity: warning
annotations:
  summary: "معدل فشل التحصيل: {{ $value }}"
  description: "أخطاء في نظام التحصيل. تحقق من اتصال OCS."
```

ملاحظات إضافية

- جميع مقاييس الزمن تستخدم دقة النانو ثانية داخليًا ولكن يتم تحويلها إلى مللي ثانية للإبلاغ
- مقاييس العداد تراكمية ويجب استخدامها مع دوال `rate()` أو `increase()` في استعلامات Prometheus
- مقاييس Gauge تمثل القيم الفورية في وقت الجمع
- توفر مقاييس Histogram حسابات النسب المئوية (p50، p95، p99) ويمكن استخدامها لإنشاء خرائط حرارية
- جميع المقاييس تشمل تسميات افتراضية تضيفها Prometheus (instance، job، إلخ.)
- عند إنشاء لوحات المعلومات، استخدم نطاقات زمنية مناسبة: 5 m للوقت الحقيقي، 1 h للاتجاهات، 24 h+ للتخطيط للطاقة

- لاستعلامات معقدة تُستخدم بشكل متكرر Prometheus قم بإعداد قواعد التسجيل في لتحسين أداء لوحة المعلومات
- استخدم قوالب المتغيرات في Grafana للوحة معلومات ديناميكية (اختر dest_smsc، source_smsc، إلخ.)

SMS-C دليل ترجمة أرقام

العودة إلى فهرس الوثائق README الرئيسي | الملف ←

نظرة عامة

يوفر نظام ترجمة أرقام SMS-C تحويلات مرنة تعتمد على regex لأرقام الهواتف قبل التوجيه. يمكن لقواعد الترجمة تطبيع الأرقام، إضافة بادئات دولية، تنسيق الأرقام لبوابات معينة، وسلسلة تحويلات متعددة معًا. يتم تخزين القواعد في Mnesia لضمان الاستمرارية ويمكن تعديلها في وقت التشغيل دون انقطاع الخدمة.

الميزات الرئيسية

- **مطابقة بناءً على البادئة**: مطابقة الأرقام حسب البادئة قبل تطبيق التحويلات
- **تحويل يعتمد على regex**: مطابقة أنماط قوية واستبدال مع مجموعات التقاط
- **تصفية SMSC المصدر**: تطبيق ترجمات مختلفة بناءً على مصدر الرسالة
- **تقييم بناءً على الأولوية**: التحكم في ترتيب القواعد مع أولويات قابلة للتكوين (-1 إلى 255)
- **سلسلة القواعد**: الاستمرار في المعالجة من خلال قواعد متعددة مع منع الحلقات
- **تحويلات منفصلة للاتصال/المتصل**: تحويل مستقل للأرقام الأصلية والوجهة
- **تحميل ملف التكوين**: تحميل القواعد الأولية من `runtime.exs` عند بدء التشغيل الأول
- **تكوين وقت التشغيل**: إضافة أو تعديل أو تعطيل القواعد دون إعادة التشغيل
- **واجهة ويب**: واجهة CRUD كاملة لإدارة القواعد
- **أداة المحاكاة**: اختبار منطق الترجمة مع تقييم خطوة بخطوة
- **نسخ احتياطي/استعادة**: تصدير واستيراد تكوينات الترجمة
- **تكامل ما قبل التوجيه**: تطبيق الترجمات قبل التوجيه لضمان تنسيقات الأرقام المتسقة

الهيكلية

نموذج البيانات

تحتوي كل قاعدة ترجمة على الحقول التالية:

الحقل	النوع	الوصف	مطلوب
`rule_id`	عدد صحيح	معرف فريد يتزايد تلقائيًا	نعم (تلقائي)
`calling_prefix`	سلسلة/لا شيء	مطابقة بادئة لرقم الاتصال (لا شيء = حرف بدل)	لا
`called_prefix`	سلسلة/لا شيء	مطابقة بادئة لرقم المتصل (لا شيء = حرف بدل)	لا
`source_smsc`	سلسلة/لا شيء	المصدر (لا شيء = SMSC اسم حرف بدل)	لا
`calling_match`	سلسلة/لا شيء	نمط regex لمطابقة رقم الاتصال	لا
`calling_replace`	سلسلة/لا شيء	نمط الاستبدال لرقم الاتصال	لا
`called_match`	سلسلة/لا شيء	نمط regex لمطابقة رقم المتصل	لا
`called_replace`	سلسلة/لا شيء	نمط الاستبدال لرقم المتصل	لا
`priority`	عدد صحيح	أولوية القاعدة (1-255، أقل = أولوية أعلى)	نعم
`description`	سلسلة	وصف قابل للقراءة البشرية	لا
`enabled`	منطقي	تمكين/تعطيل القاعدة	نعم
`continue`	منطقي	الاستمرار في تقييم القواعد بعد المطابقة (افتراضي: خطأ)	لا

ملاحظة: يتم تقييم القواعد بترتيب الأولوية (أقل رقم أولاً). يتم تقييم القواعد المفعلة فقط.

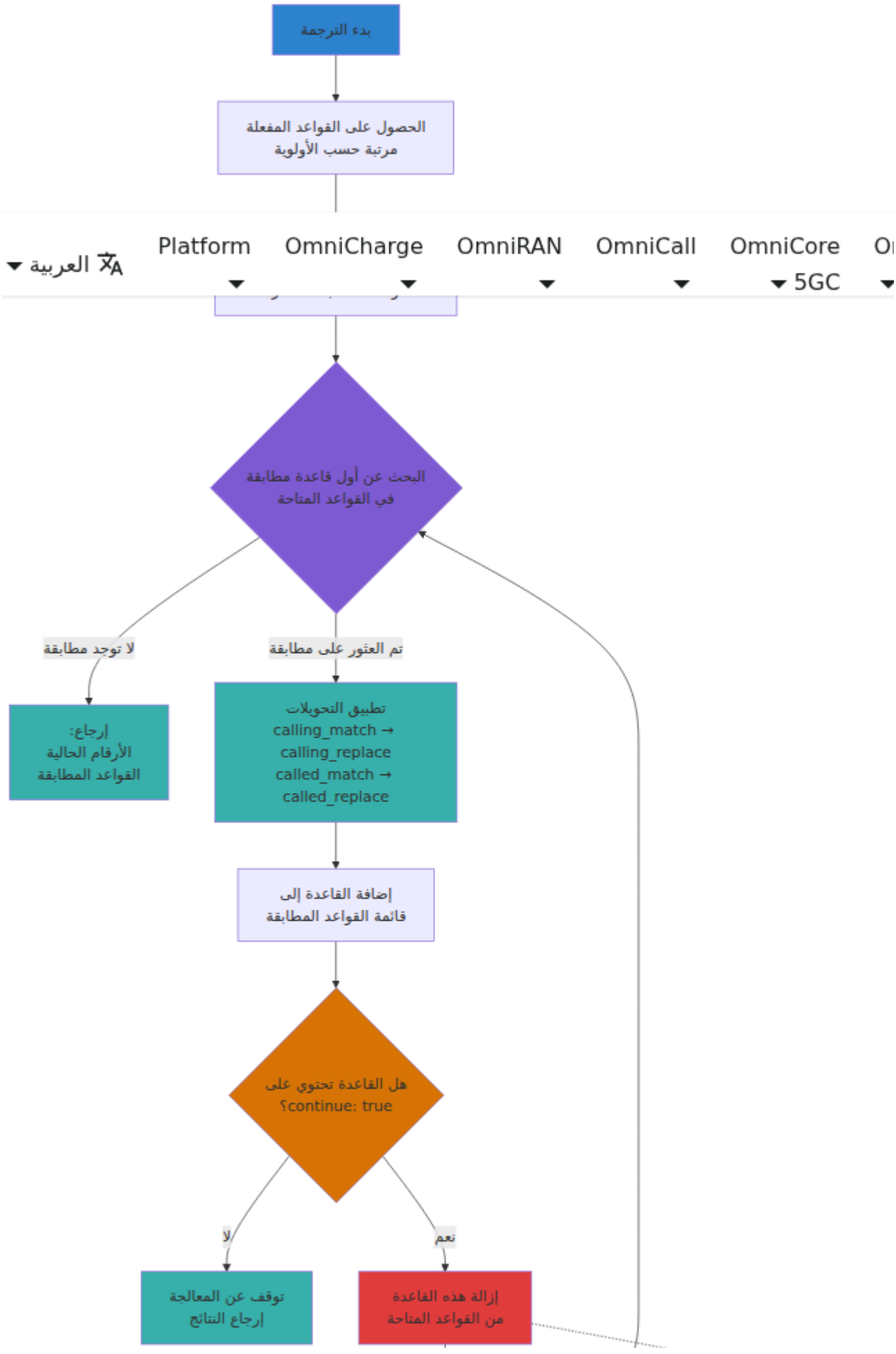
خوارزمية الترجمة

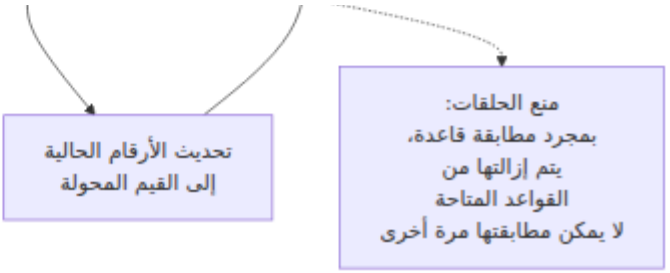
عند ترجمة الأرقام، يقوم النظام:

1. **استرجاع القواعد المفعلة** مرتبة حسب الأولوية (الأدنى أولاً)
2. **تقييم القواعد بالتتابع** ضد معلمات الرسالة:
   - مطابقة `calling_prefix` (إذا تم تحديده)
   - مطابقة `called_prefix` (إذا تم تحديده)
   - مطابقة `source_smsc` (إذا تم تحديده)
3. **تطبيق أول قاعدة مطابقة**:
   - تحويل رقم الاتصال باستخدام `calling_match` و `calling_replace`
   - تحويل رقم المتصل باستخدام `called_match` و `called_replace`
4. **التحقق من علامة الاستمرار**:
   - إذا كانت `continue: false` → توقف عن المعالجة، ارجع النتيجة
   - إذا كانت `continue: true` → إزالة القاعدة المطابقة من القواعد المتاحة، الاستمرار مع الخطوة 2 باستخدام **الأرقام المحولة**
5. **إرجاع الأرقام النهائية** وقائمة بجميع القواعد المطبقة

سلسلة القواعد مع منع الحلقات

سلسلة قوية من القواعد مع منع الحلقات `continue` تتيح ��لامة:


بدء الترجمة
الحصول على القواعد المفعلة
مرتبة حسب الأولوية
العربية
Platform
OmniCharge
OmniRAN
OmniCall
OmniCore
5GC
البحث عن أول قاعدة مطابقة
في القواعد المتاحة
لا توجد مطابقة
تم العثور على مطابقة
إرجاع:
الأرقام الحالية
القواعد المطابقة
تطبيق التحويلات
calling_match →
calling_replace
called_match →
called_replace
إضافة القاعدة إلى
قائمة القواعد المطابقة
هل القاعدة تحتوي على
continue: true؟
لا
نعم
توقف عن المعالجة
إرجاع النتائج
إزالة هذه القاعدة
من القواعد المتاحة



الأحرف البديلة

- `لا شيء` أو القيم الفارغة تعمل كأحرف بديلة تتطابق مع أي قيمة
- القاعدة التي لا تحتوي على معايير مطابقة هي قاعدة شاملة
- القاعدة التي لا تحتوي على أنماط تحويل (مطابقة/استبدال لا شيء) تمرر الأرقام دون تغيير

مثال: سيناريو سلسلة القواعد

Parse error on line 20: ...] style R1 fill:#38B2AC style R ----------^
Expecting 'SOLID_OPEN_ARROW', 'DOTTED_OPEN_ARROW', 'SOLID_ARROW', 'BIDIRECTIONAL_SOLID_ARROW', 'DOTTED_ARROW', 'BIDIRECTIONAL_DOTTED_ARROW', 'SOLID_CROSS', 'DOTTED_CROSS', 'SOLID_POINT', 'DOTTED_POINT', got 'TXT'

المحاولة مجددا

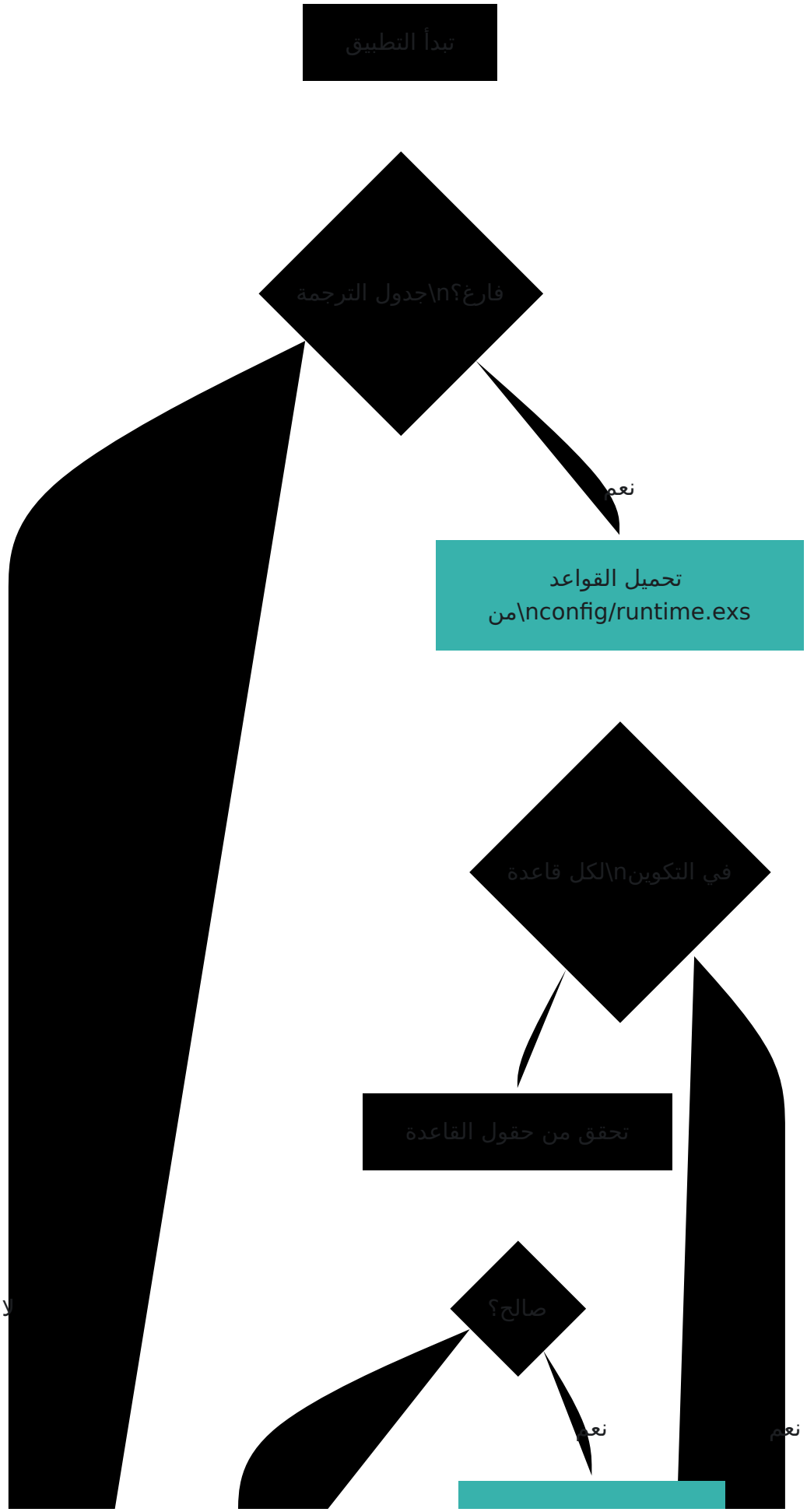
التكوين

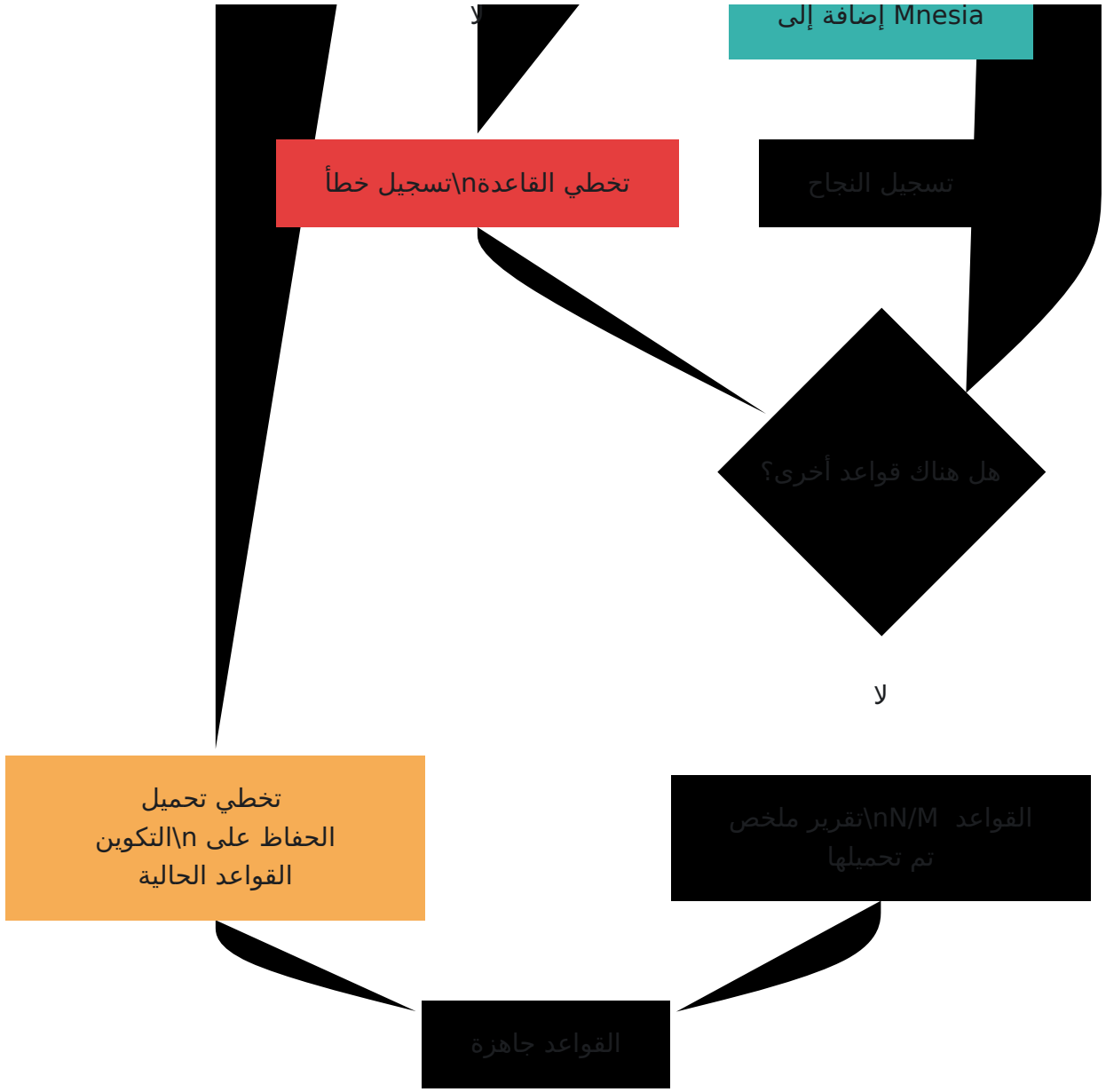
تحميل القواعد من ملف التكوين

يمكن تعريف قواعد الترجمة في `config/runtime.exs` وسيتم تحميلها تلقائيًا عند بدء التشغيل الأول.

مهم: يتم تحميل القواعد من التكوين فقط عندما تكون جدول الترجمة **فارغًا** (عند بدء التشغيل الأول). هذا يحافظ على القواعد المضافة عبر واجهة الويب أثناء وقت التشغيل ويمنع التكرارات عند إعادة التشغيل.

تدفق تحميل التكوين


تبدأ التطبيق
فارغ؟\nجدول الترجمة
نعم
تحميل القواعد
من\nconfig/runtime.exs
في التكوين\nلكل قاعدة
تحقق من حقول القاعدة
صالح؟
لا
نعم
نعم



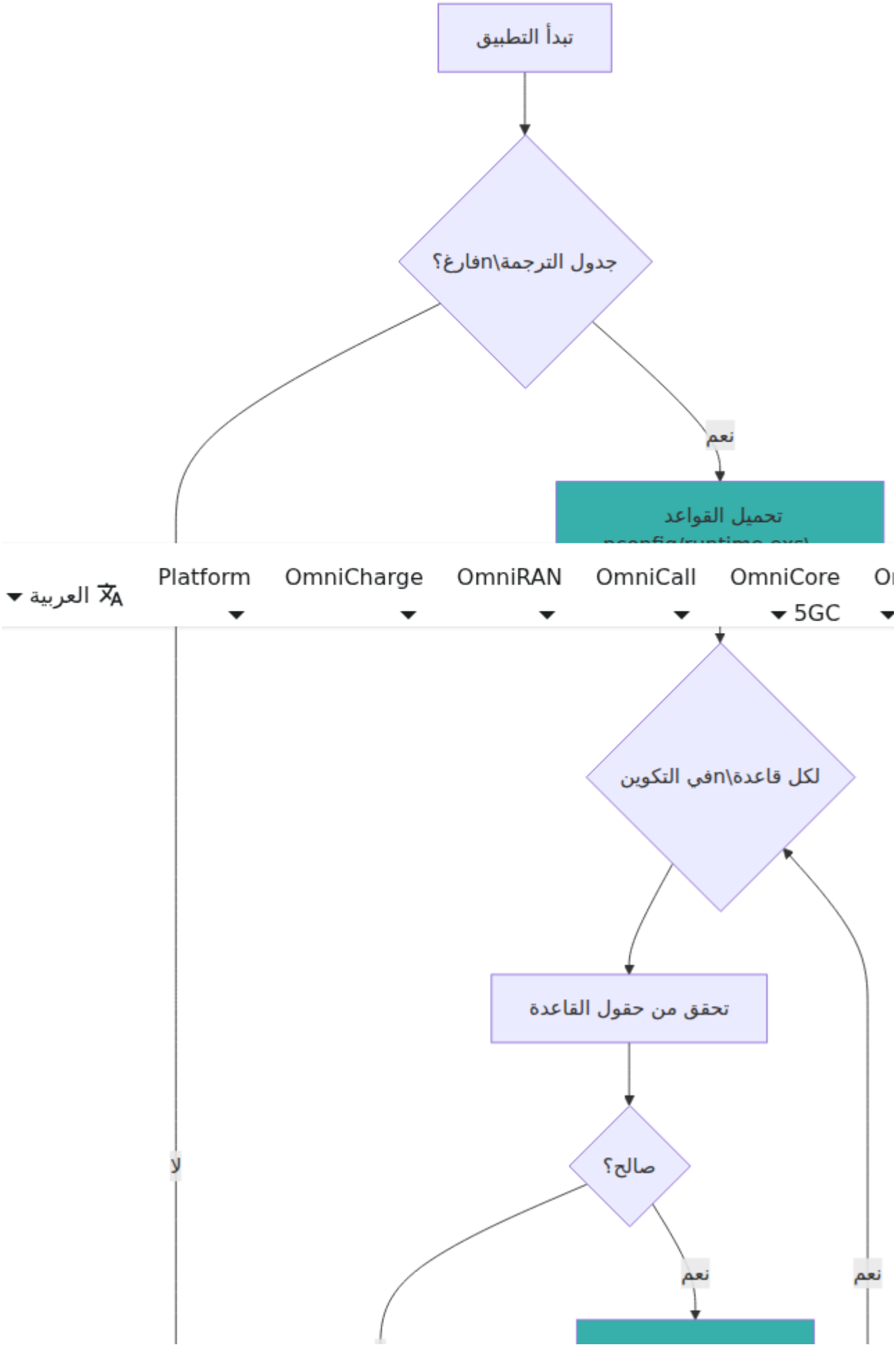
مثال على التكوين

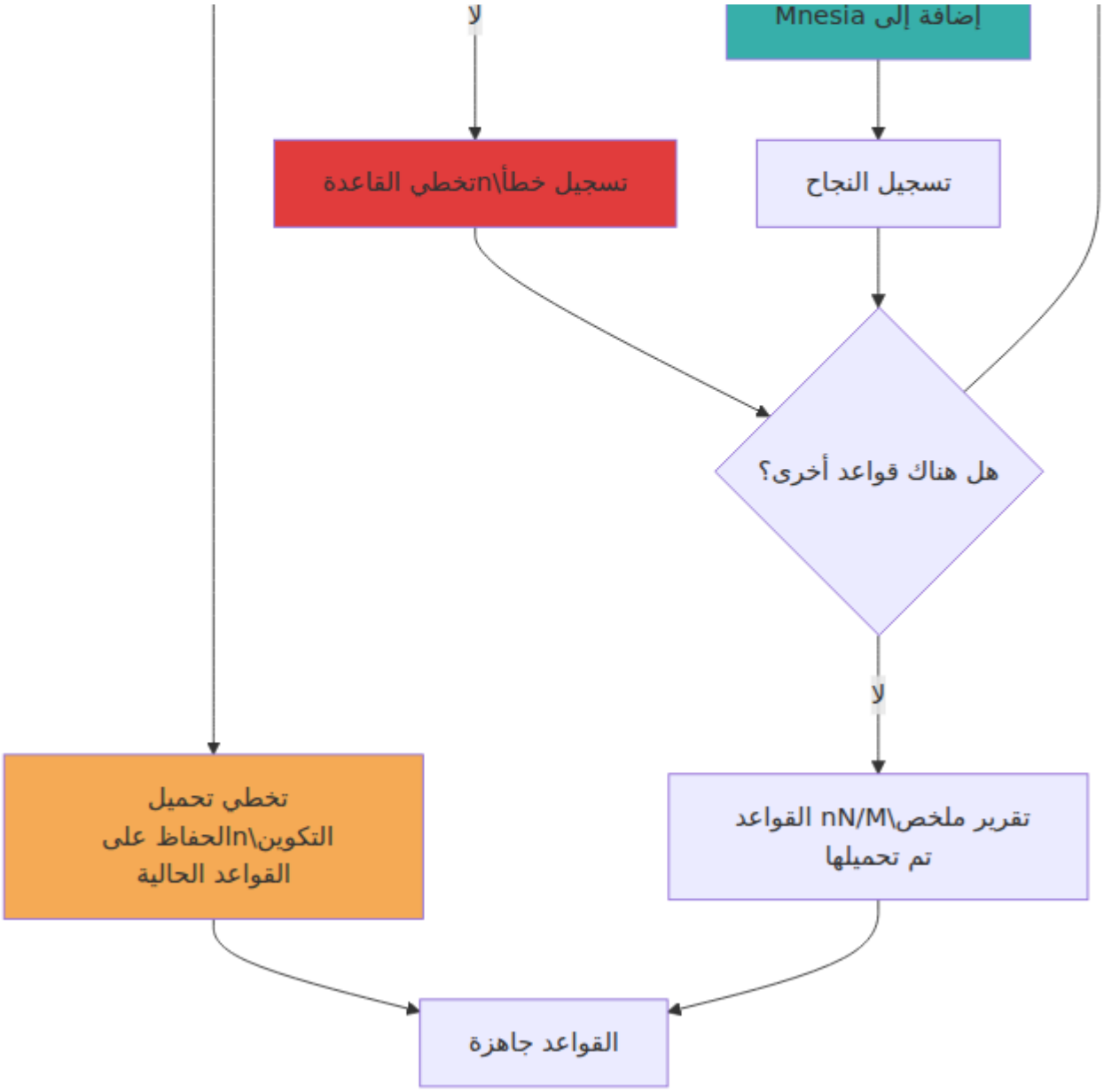
```
# config/runtime.exs
config :sms_c, :translation_rules, [
  # إضافة +1 إلى أرقام الولايات المتحدة المكونة من 10 أرقام
  %{
    calling_prefix: nil,
    called_prefix: nil,
    source_smsc: "us_domestic_smsc",
    calling_match: "^(\d{10})$",
    calling_replace: "+1\1",
    called_match: "^(\d{10})$",
    called_replace: "+1\1",
    priority: 10,
    description: "إضافة +1 إلى أرقام الولايات المتحدة المكونة من 10
أرقام من SMSC المحلي",
    enabled: true,
    continue: false
  },

  # إزالة الأصفار البادئة من التنسيق الدولي
  %{
    calling_prefix: "00",
    called_prefix: nil,
    source_smsc: nil,
    calling_match: "^00(.+)$",
    calling_replace: "+\1",
    called_match: nil,
    called_replace: nil,
    priority: 5,
    description: "تحويل بادئة 00 الدولية إلى +",
    enabled: true,
    continue: true  # الاستمرار في تطبيق المزيد من التنسيقات
  },

  # تنسيق أرقام المملكة المتحدة لبوابة معينة
  %{
    calling_prefix: "+44",
    called_prefix: "+44",
    source_smsc: nil,
    calling_match: "^\+44(.*)$",
    calling_replace: "0044\1",
    called_match: "^\+44(.*)$",
    called_replace: "0044\1",
    priority: 20,
```

```
    description: "تنسيق أرقام المملكة المتحدة لبوابة قديمة",
    enabled: true,
    continue: false
  }
]
```

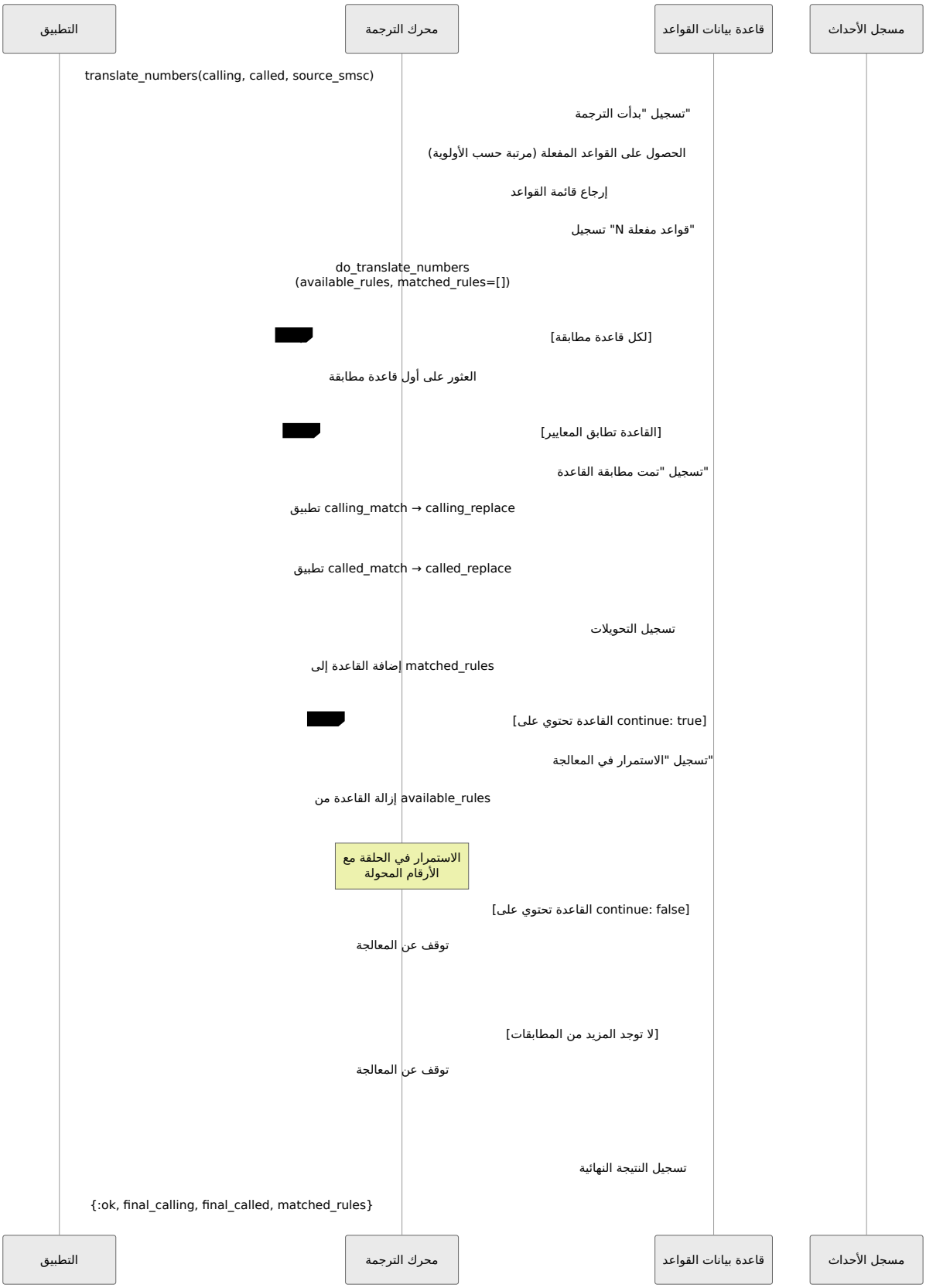
البدء

تدفق التهيئة


تبدأ التطبيق
جدول الترجمة\nفارغ؟
نعم
تحميل القواعد
العربية
Platform
OmniCharge
OmniRAN
OmniCall
OmniCore
5GC
لكل قاعدة\nفي التكوين
تحقق من حقول القاعدة
صالح؟
نعم
نعم



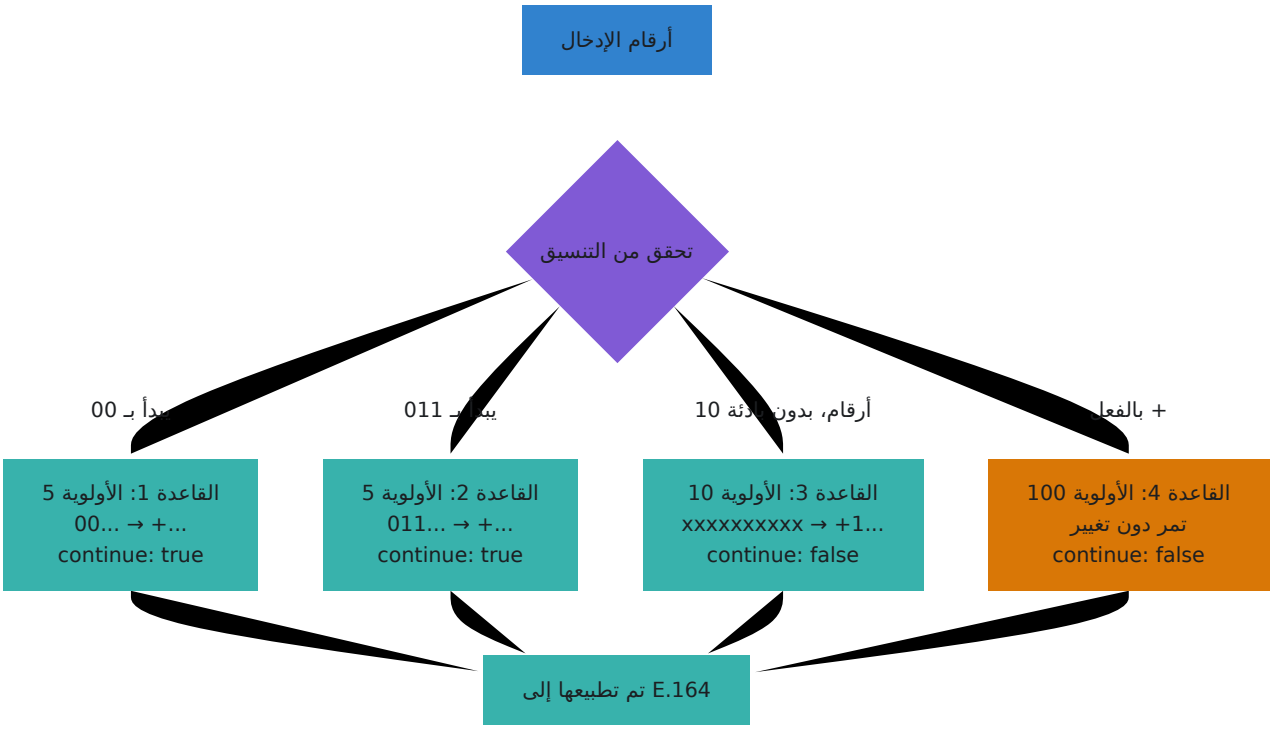
تدفق ترجمة الرسالة


التطبيق
محرك الترجمة
قاعدة بيانات القواعد
مسجل الأحداث
translate_numbers(calling, called, source_smsc)
"تسجيل "بدأت الترجمة
الحصول على القواعد المفعلة (مرتبة حسب الأولوية)
إرجاع قائمة القواعد
"قواعد مفعلة N" تسجيل
do_translate_numbers
(available_rules, matched_rules=[])
[لكل قاعدة مطابقة]
العثور على أول قاعدة مطابقة
[القاعدة تطابق المعايير]
"تسجيل "تمت مطابقة القاعدة
تطبيق calling_match → calling_replace
تطبيق called_match → called_replace
تسجيل التحويلات
إضافة القاعدة إلى matched_rules
[القاعدة تحتوي على continue: true]
"تسجيل "الاستمرار في المعالجة
إزالة القاعدة من available_rules
الاستمرار في الحلقة مع
الأرقام المحولة
[القاعدة تحتوي على continue: false]
توقف عن المعالجة
[لا توجد المزيد من المطابقات]
توقف عن المعالجة
تسجيل النتيجة النهائية
{:ok, final_calling, final_called, matched_rules}
التطبيق
محرك الترجمة
قاعدة بيانات القواعد
مسجل الأحداث

حالات الاستخدام الشائعة

تطبيع الأرقام الدولية

تطبيع تنسيقات دولية مختلفة إلى E.164:



تنسيق خاص بالبوابة

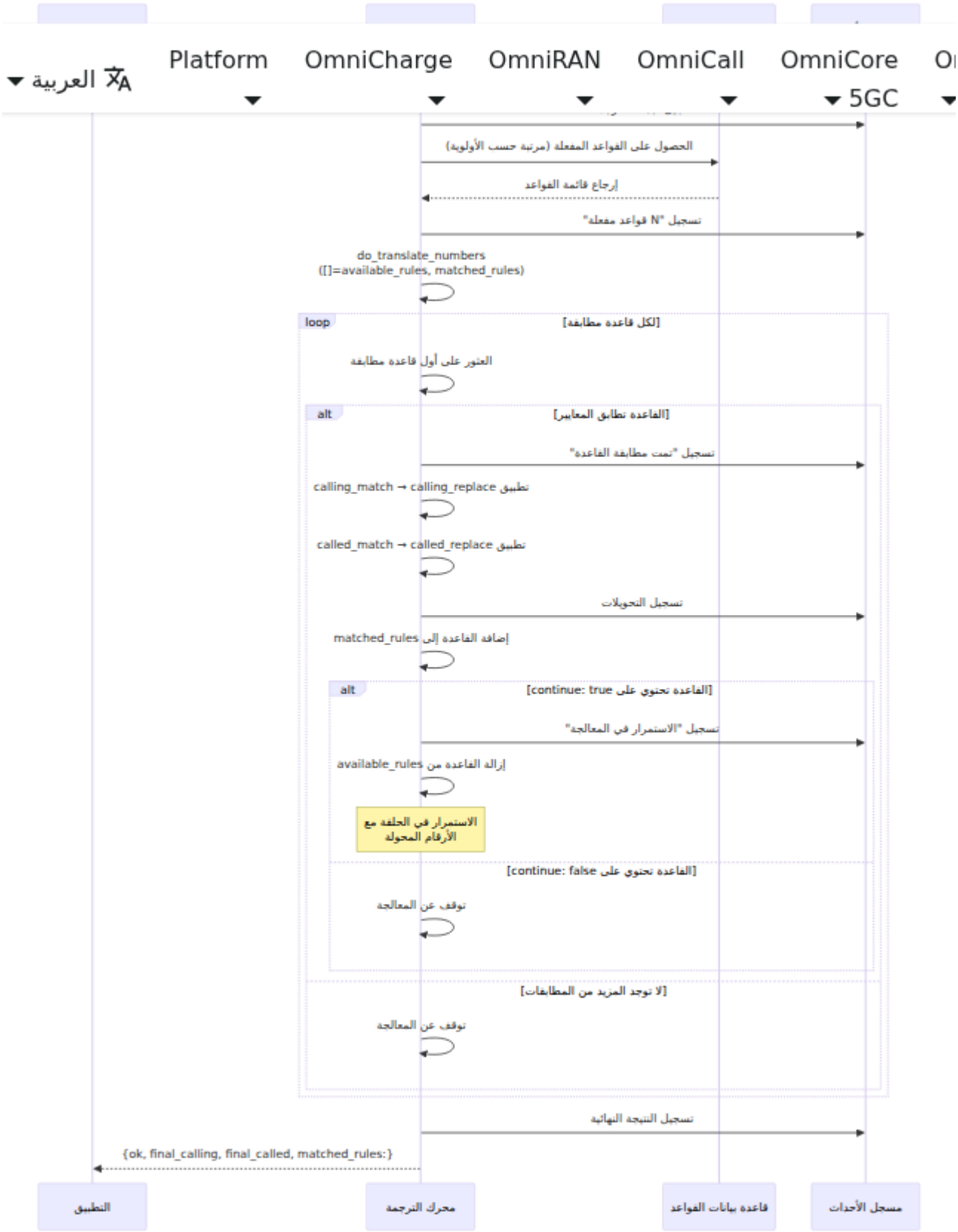
سلسلة القواعد لتنسيق الأرقام وفقًا لمتطلبات بوابة معينة:

Parse error on line 2: ... TD I[5551234567" :الإدخال"] --> S1[----------^
Expecting 'SQE', 'DOUBLECIRCLEEND', 'PE', '-)', 'STADIUMEND', 'SUBROUTINEEND', 'PIPE', 'CYLINDEREND', 'DIAMOND_STOP', 'TAGEND', 'TRAPEND', 'INVTRAPEND', 'UNICODE_TEXT', 'TEXT', 'TAGSTART', got 'STR'

المحاولة مجددا

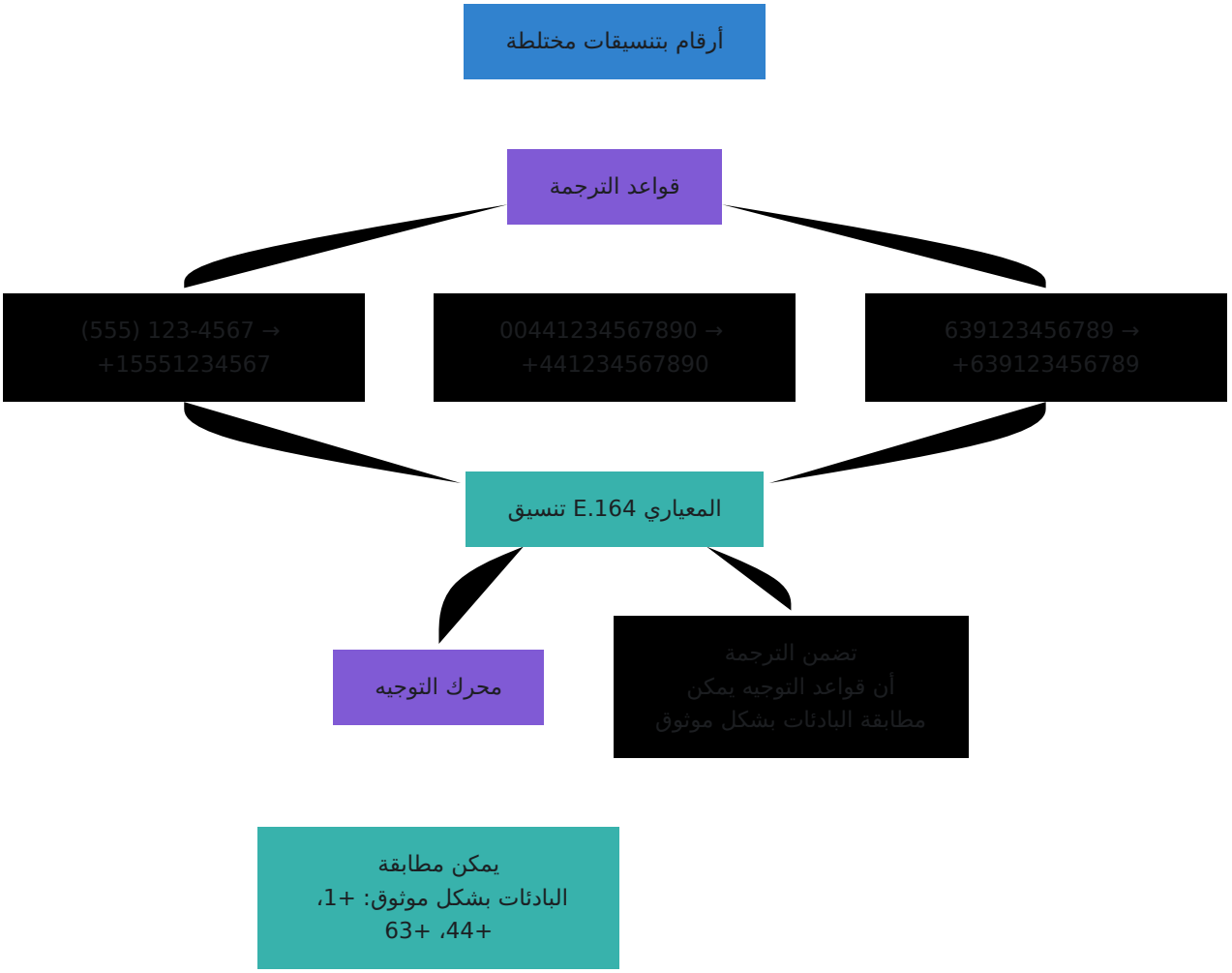
ترجمات خاصة بـ SMSC

تطبيق ترجمات مختلفة بناءً على مصدر الرسالة:



إعداد التوجيه بناءً على البادئة

تطبيع الأرقام قبل التوجيه لضمان مطابقة بادئات متسقة:



التعامل مع قابلية نقل الأرقام

التعامل مع الأرقام المنقولة التي تتطلب تغييرات في البادئة:

Parse error on line 18: ... style Input fill:#3182CE style R ----------^
Expecting 'SOLID_OPEN_ARROW', 'DOTTED_OPEN_ARROW', 'SOLID_ARROW', 'BIDIRECTIONAL_SOLID_ARROW', 'DOTTED_ARROW', 'BIDIRECTIONAL_DOTTED_ARROW', 'SOLID_CROSS', 'DOTTED_CROSS', 'SOLID_POINT', 'DOTTED_POINT', got 'TXT'

المحاولة مجددا

واجهة الويب

واجهة إدارة قواعد الترجمة

يمكن الوصول إلى واجهة إدارة القواعد في `/number_translation` (عبر قائمة التنقل):

الميزات:

- عرض جميع القواعد في جدول قابل للفرز حسب الأولوية
- إضافة قواعد جديدة مع التحقق من صحة النموذج
- تعديل القواعد الحالية
- تمكين/تعطيل القواعد دون حذف
- حذف القواعد مع تأكيد
- مؤشر مرئي للقواعد التي تحتوي على `continue: true`
- استيراد/تصدير القواعد بصيغة JSON

إضافة قاعدة:

1. املأ معايير المطابقة (اختياري):
   - بادئة الاتصال (مثل "+1"، "44")
   - بادئة المتصل (مثل "+639"، "1555")
   - المصدر (اتركه فارغًا لأي) SMSC
2. تحديد التحويلات (اختياري):
   - مطابقة regex لرقم الاتصال واستبداله
   - مطابقة regex لرقم المتصل واستبداله
3. تعيين الأولوية (1-255، أقل = أولوية أعلى)
4. تعيين الحالة:
   - **مفعل**: القاعدة نشطة
   - **استمرار المعالجة**: الاستمرار في تقييم المزيد من القواعد بعد هذه القاعدة
5. إضافة وصف
6. انقر على "إضافة قاعدة" أو "تحديث القاعدة"

تبديل الاستمرار في المعالجة:

- **توقف** (افتراضي): توقف عن المعالجة بعد مطابقة هذه القاعدة

- **استمرار**: تطبيق هذه القاعدة والاستمرار في تقييم القواعد المتبقية
- القواعد التي تم تمكين الاستمرار تظهر شارة زرقاء "↓ استمرار" في الجدول

تحرير قاعدة:

1. انقر على "تحرير" بجوار القاعدة
2. تعديل الحقول حسب الحاجة
3. انقر على "تحديث القاعدة"

مؤشرات جدول القواعد:

- شارة **مفعل/معطل** تظهر حالة القاعدة
- شارة **↓ استمرار** تظهر القواعد التي ستستمر في المعالجة
- شارة **الأولوية** تظهر ترتيب التقييم
- بخط أحادي المسافة لزيادة الوضوح regex يتم عرض أنماط

محاكي الترجمة

يمكن الوصول إلى المحاكي في `/translation_simulator` (عبر قائمة التنقل):

الميزات:

- اختبار منطق الترجمة مع أرقام فعلية
- **تحويل خطوة بخطوة** يظهر كل قاعدة تم تطبيقها
- رؤية القيم قبل/بعد لكل تحويل
- عرض القواعد التي تطابقت ولماذا
- تحميل سيناريوهات مثال للاختبار السريع
- عرض تاريخ الاختبار (آخر 10 اختبارات)

استخدام المحاكي:

1. إدخال معلمات الاختبار:
   - رقم الاتصال (من)
   - رقم المتصل (إلى)
   - SMSC (اختياري) المصدر
2. انقر على "اختبار الترجمة"
3. عرض النتائج الشاملة:

- **نتيجة الترجمة**: الأرقام النهائية بعد جميع التحويلات
- **القواعد المطبقة**: عدد وقائمة جميع القواعد التي تطابقت
- **تحويلات خطوة بخطوة**: عرض مفصل لكل قاعدة:
  - رقم الخطوة ومعلومات القاعدة
  - وصف القاعدة
  - قبل → بعد لكل من أرقام الاتصال والمتصل
  - مؤشر "↓ استمرار" للقواعد التي استمرت في المعالجة
  - التحويلات مميزة باللون الأخضر
  - القيم غير المتغيرة تم وضع علامة عليها كـ "تم تمريرها"

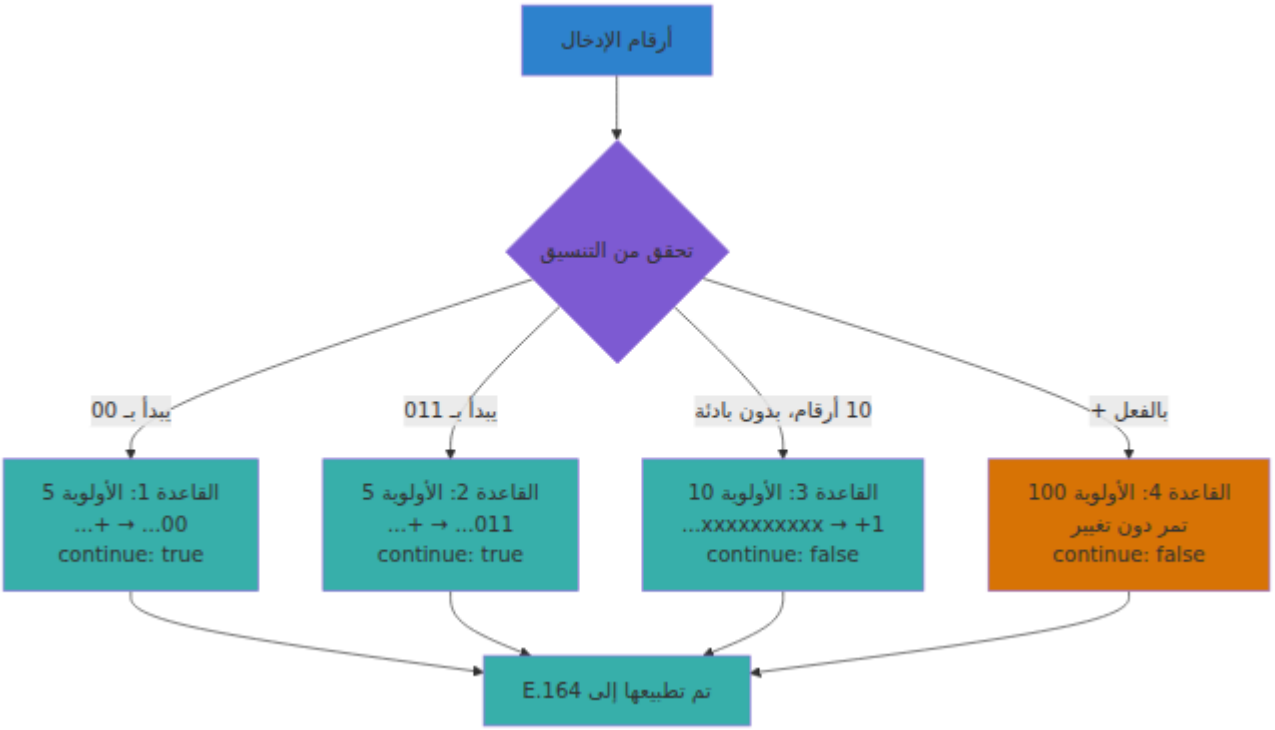
4. تحميل أمثلة مسبقة التكوين باستخدام أزرار الأمثلة
5. مراجعة تاريخ الاختبار لمقارنة السيناريوهات المختلفة

مثا�ة على الإخراج:

```
نتيجة الترجمة
═══════════════════════════════════════════

رقم الاتصال:  5551234567 ← +1-555-123-4567
رقم المتصل:   9078720155 ← +1-907-872-0155
✓ تمت الترجمة بواسطة 3 قاعدة(قواعد)

تحويلات خطوة بخطوة
═══════════════════════════════════════════

┌─ الخطوة 1 ──────────────────────────────┐
│ القاعدة #1 (الأولوية 10)                │
│ ↓ استمرار                               │
│ إضافة رمز البلد إلى أرقام مكونة من 10 أرقام │
│                                         │
│ المتصل: 9078720155 → +19078720155       │
└─────────────────────────────────────────┘

┌─ الخطوة 2 ──────────────────────────────┐
│ القاعدة #2 (الأولوية 20)                │
│ ↓ استمرار                               │
│ تنسيق رمز المنطقة مع شرطات              │
│                                         │
│ المتصل: +19078720155 → +1-907-872-0155  │
└─────────────────────────────────────────┘

┌─ الخطوة 3 ──────────────────────────────┐
│ القاعدة #3 (الأولوية 30)                │
│ التنسيق النهائي للبوابة                 │
│                                         │
│ المتصل: +1-907-872-0155 → +1-907-872-0155 │
└─────────────────────────────────────────┘
```

مرجع API

نظرة عامة على العمليات الأساسية



معلمات الترجمة

translate_numbers يقبل المعلمات التالية:

- `calling_number` (اختياري): رقم الهاتف الأصلي
- `called_number` (اختياري): رقم الهاتف الوجهة
- `source_smsc` (اختياري): معرف SMSC المصدر
- `message_id` (اختياري): لتسجيل الأحداث

الإ��جاع:

- `{:ok, translated_calling, translated_called, [rules_applied]}` - دائمًا ناجح
- إرجاع الأرقام الأصلية إذا لم تتطابق أي قواعد
- إرجاع قائمة بجميع القواعد التي تم تطبيقها (بالترتيب)

```
# مثال على الاستخدام
{:ok, new_calling, new_called, rules} =
  NumberTranslation.translate_numbers(
    calling_number: "5551234567",
    called_number: "9078720155",
    source_smsc: "domestic_gateway",
    message_id: "msg_123"
  )

# تحقق مما إذا كانت هناك أي ترجمة حدثت
if rules != [] do
  Logger.info("تم تطبيق #{length(rules)} قاعدة ترجمة")
  Enum.each(rules, fn rule ->
    Logger.info("  - القاعدة ##{rule.rule_id}: #
{rule.description}")
  end)
end
```

عمليات إدارة القواعد

```
# إضافة قاعدة جديدة
{:ok, rule} = NumberTranslation.add_rule(%{
  calling_prefix: nil,
  called_prefix: nil,
  source_smsc: "gateway1",
  calling_match: "^(\d{10})$",
  calling_replace: "+1\1",
  called_match: "^(\d{10})$",
  called_replace: "+1\1",
  priority: 10,
  description: "إضافة +1 إلى الأرقام المكونة من 10 أرقام",
  enabled: true,
  continue: false
})

# تحديث قاعدة
{:ok, updated_rule} = NumberTranslation.update_rule(rule_id, %{
  enabled: false,
  description: "معطلة للاختبار"
})

# حذف قاعدة
:ok = NumberTranslation.delete_rule(rule_id)

# الحصول على قاعدة معينة
rule = NumberTranslation.get_rule(rule_id)

# قائمة بجميع القواعد
all_rules = NumberTranslation.list_rules()

# قائمة بالقواعد المفعلة فقط (مرتبة حسب الأولوية)
enabled_rules = NumberTranslation.list_enabled_rules()
```

عمليات الاستيراد/التصدير

```
# تصدير جميع القواعد
backup = NumberTranslation.export_rules()
# الإرجاع: %{
#   version: "1.0",
#   exported_at: ~U[2024-01-15 10:30:00Z],
#   count: 5,
#   rules: [...]
# }

# حفظ في ملف JSON
json = Jason.encode!(backup, pretty: true)
File.write!("translation_rules_backup.json", json)

# استيراد القواعد (دمج مع الموجود)
{:ok, %{imported: 3, failed: 0}} =
  NumberTranslation.import_rules(backup, mode: :merge)

# استيراد القواعد (استبدال جميع الموجودات)
{:ok, %{imported: 5, failed: 0}} =
  NumberTranslation.import_rules(backup, mode: :replace)
```

أفضل الممارسات

تصميم القواعد

1. **الحفاظ على تنظيم الأولويات**:
   - **1-10**: قواعد التطبيع الحرجة (إضافة رموز البلدان، إصلاح التنسيقات)
   - **11-50**: تنسيق خاص بالبوابة
   - **51-100**: تحويلات اختيارية
   - **101+**: قواعد شاملة أو قواعد تصحيح
2. **استخدام الاستمرار بشكل استراتيجي**:
   - تمكين `continue: true` لقواعد التطبيع التي تعد الأرقام لمزيد من المعالجة

- تعطيل `continue: false` لقواعد التنسيق النهائي
- تجنب السلاسل الطويلة (3-4 قواعد كحد أقصى) للحفاظ على الأداء

3. **توثيق قواعدك**:

   - دائمًا أضف أوصافًا واضحة
   - تضمين أمثلة في الوصف (مثل "5551234567 → +15551234567")
   - توثيق الغرض والمدخلات/المخرجات المتوقعة

4. **اختبار أنماط regex**:

   - اختبار الأنماط باستخدام المحاكي قبل النشر
   - استخدام مجموعات التقاط (\1، \2) للتحويلات المرنة
   - الهروب من الأحرف الخاصة في regex (النقاط، الأقواس، إلخ)

الأداء

1. **تقليل عدد القواعد**:

   - دمج القواعد المماثلة حيثما كان ذلك ممكنًا
   - استخدام مطابقة البادئة لتقليل تقييمات regex
   - إزالة أو تعطيل القواعد غير المستخدمة

2. **تحسين أنماط regex**:

   - استخدام مطابقة البادئة أولاً (أسرع من regex)
   - الحفاظ على أنماط regex بسيطة
   - تجنب الأنماط الثقيلة على التراجع

3. **تحديد سلسلة القواعد**:

   - يمكن أن تؤثر السلاسل الطويلة (5+ قواعد) على الأداء
   - النظر في دمج خطوات متعددة في قاعدة واحدة إذا كان ذلك ممكنًا
   - مراقبة زمن الترجمة باستخدام مقاييس Telemetry

العمليات

1. **اختبار قبل النشر**:

- استخدام المحاكي مع أمثلة من العالم الحقيقي
- اختبار الحالات الحدية (الأرقام الفارغة، الأحرف الخاصة)
- التحقق من سلوك علامة الاستمرار

2. **النسخ الاحتياطي بانتظام**:

- تصدير القواعد قبل إجراء تغييرات كبيرة
- التحكم في إصدار الصادرات الخاصة بك
- اختبار الاستيرا��ات في بيئة غير إنتاجية أولاً

3. **مراقبة الترجمات**:

- تمكين تسجيل message_id لأغراض التصحيح
- تحقق من سجلات الأحداث لقرارات الترجمة
- مراقبة القواعد التي يتم تطبيقها

4. **نشر تدريجي**:

- إضافة قواعد جديدة كمعطلة أولاً
- اختبار باستخدام المحاكي
- تمكين ومراقبة
- ضبط حسب الحاجة

نصائح regex

1. **أنماط شائعة**:

- رقم الولايات المتحدة المكون من 10 أرقام: `^(\d{10})$`
- التنسيق الدولي: `^\+(\d+)$`
- إزالة الأصفار البادئة: `^0+(.+)$`
- إضافة شرطات: `^(\d{3})(\d{3})(\d{4})$` → `\1-\2-\3`

2. **مجموعات التقاط**:

- استخدام الأقواس للاحتفاظ: `^(\d{3})(\d{7})$`
- الإشارة في الاستبدال: `+1\1\2`
- عدة التقاطات: `^\+(\d{1,3})(\d+)$` → `00\1\2`

3. **الهروب من الأحرف الخاصة**:
    - نقطة حرفية: `\.`
    - زائد حرفي: `\+`
    - قوس حرفي: `\(` أو `\)`

استكشاف الأخطاء وإصلاحها

القاعدة لا تطابق

العرض: القاعدة المتوقعة لا تطابق، تمر الأرقام دون تغيير

الأسباب المحتملة:

- البادئة لا تطابق (تحقق من المطابقة الدقيقة للبادئة)
- المصدر لا تطابق SMSC
- نمط regex لا يطابق تنسيق الإدخال
- القاعدة معطلة
- قاعدة ذات أولوية أعلى تطابقت أولاً (مع continue: false)

الحلول:

1. استخدم المحاكي لرؤية القواعد التي يتم تقييمها
2. تحقق من حالة القاعدة (مفعل/معطل)
3. تحقق من مطابقة البادئة (حساسة لحالة الأحرف)
4. اختبار نمط regex بشكل منفصل
5. تحقق من ترتيب الأولوية

التحويل الخاطئ المطبق

العرض: تم تحويل الرقم ولكن النتيجة غير صحيحة

الأسباب المحتملة:

- نمط regex يطابق ولكن نمط الاستبدال خاطئ
- تطبيق قواعد متعددة بترتيب غير متوقع

- مراجع مجموعات الالتقاط غير صحيحة (\1، \2، إلخ)

الحلول:

1. استخدم المحاكي لرؤية التحويلات خطوة بخطوة
2. يلتقط المجموعات الصحيحة regex تحقق من أن نمط
3. تحقق من بناء جملة نمط الاستبدال
4. عبر الإنترنت regex في أداة اختبار regex اختبار
5. مراجعة أولوية القاعدة وعلامات الاستمرار

حلقة لا نهائية / تدهور الأداء

العرض: الترجمة تستغرق وقتًا طويلاً جدًا أو يبدو أنها تتوقف

ملاحظة: يجب ألا يحدث هذا بسبب منع الحلقات، ولكن إذا حدث:

الأسباب المحتملة:

- خطأ في منطق منع الحلقات
- طويل جدًا regex تقييم
- سلسلة قواعد طويلة جدًا

الحلول:

1. تحقق من سجلات التطبيق للبحث عن الأخطاء
2. مراجعة القواعد مع continue: true
3. تبسيط أنماط regex
4. تقليل عدد القواعد المتسلسلة
5. الإبلاغ عن خطأ إذا فشل منع الحلقة

سلسلة قواعد غير متوقعة

العرض: تم تطبيق المزيد من القواعد أكثر مما هو متوقع

الأسباب المحتملة:

- عندما لا ينبغي continue: true القواعد تحتوي على
- ترتيب الأولوية يسمح بمطابقات متعددة

- الرقم المحول يطابق قواعد إضافية

الحلول:

1. استخدم المحاكي لرؤية سلسلة القواعد الدقيقة
2. مراجعة علامات الاستمرار على جميع القواعد
3. ضبط الأولويات للتحكم في الترتيب
4. تعيين continue: false على القاعدة النهائية

الترجمة غير مطبقة قبل التوجيه

العرض: جهاز التوجيه يرى أرقامًا غير مترجمة

الأسباب المحتملة:

- الترجمة غير مدمجة في تدفق الرسالة
- تحدث الترجمة بعد التوجيه
- يتجاوز كود التطبيق الترجمة

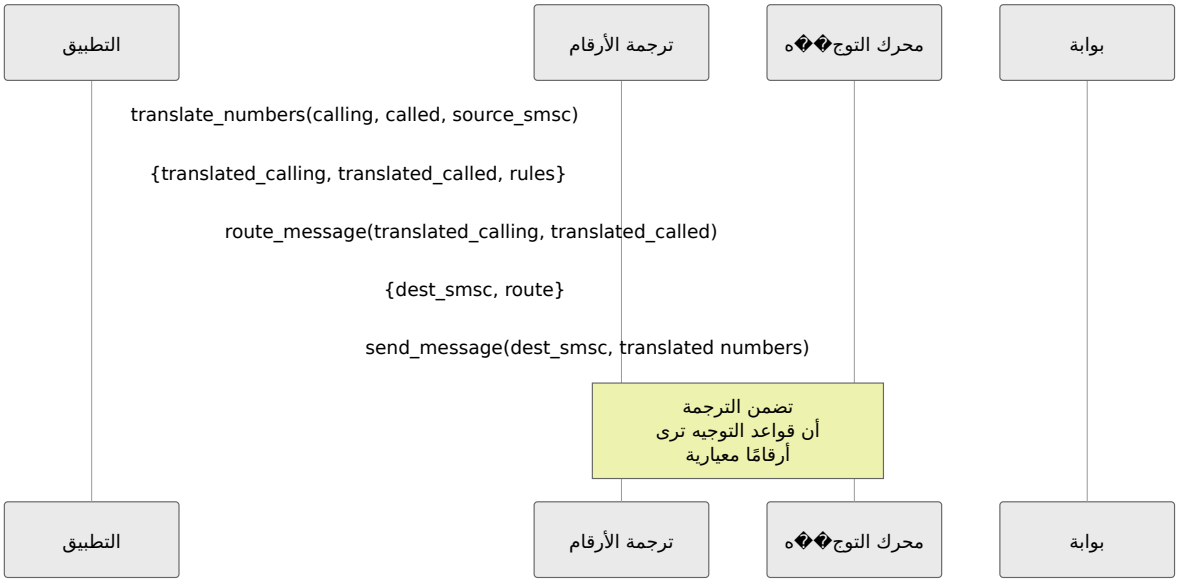
الحلول:

1. تحقق من تكامل التطبيق: يجب استدعاء الترجمة قبل التوجيه
2. تحقق من خط أنابيب معالجة الرسالة
3. مراجعة سجلات الأحداث لقرارات الترجمة
4. التأكد من استدعاء translate_numbers بالترتيب الصحيح

مواضيع متقدمة

التكامل مع التوجيه

تحدث الترجمة **قبل** التوجيه لضمان تنسيقات الأرقام المتسقة:



تسجيل الأحداث

يتم تسجيل قرارات الترجمة عبر EventLogger:

- `translation_started`: تبدأ الترجمة
- `translation_candidates`: عدد القواعد المفعلة
- `translation_matched`: تم مطابقة القاعدة وتطبيقها
- `translation_calling`: تم تحويل رقم الاتصال
- `translation_called`: تم تحويل رقم المتصل
- `translation_continue`: القاعدة تحتوي على continue=true، الاستمرار في التقييم
- `translation_none`: لم تتطابق أي قواعد

تمكين التسجيل عن طريق تمرير `message_id` إلى `translate_numbers/1`.

مقاييس Telemetry

مراقبة أداء الترجمة باستخدام Telemetry:

```
:telemetry.attach(
  "number-translation-handler",
  [:sms_c, :number_translation, :translate, :stop],
  fn _event_name, measurements, metadata, _config ->
    # measurements: %{duration: microseconds}
    # metadata: %{rules_applied: count, ...}
  end,
  nil
)
```

المقاييس الرئيسية للمراقبة:

- مدة الترجمة (p50، p95، p99)
- القواعد المطبقة لكل رسالة
- القواعد المطابقة مقابل غير المطابقة
- استخدام علامة الاستمرار

التجميع

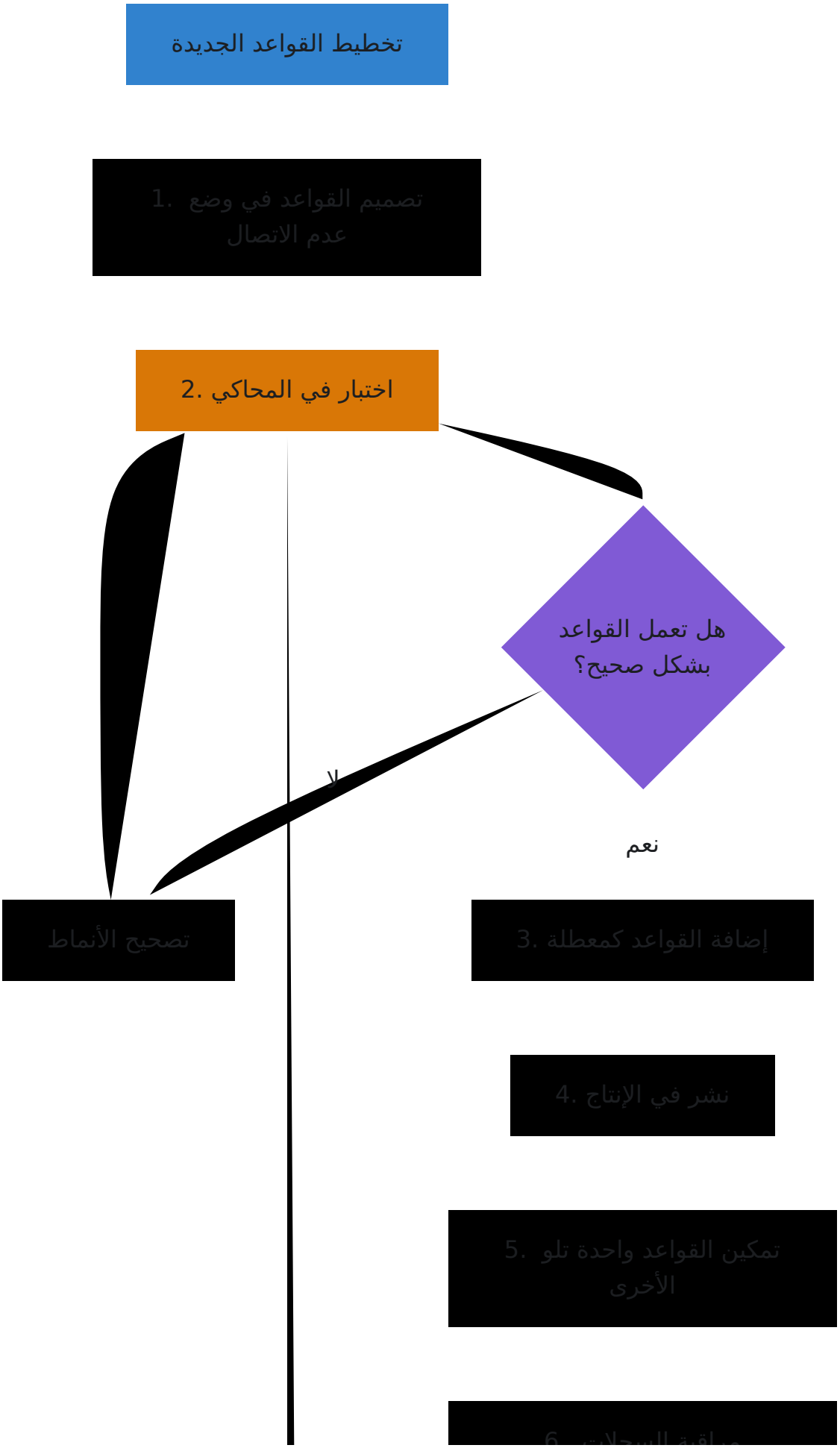
توزع جداول Mnesia تلقائيًا عبر العقد المجمعة. يتم تكرار قواعد الترجمة لضمان توفر عالي.

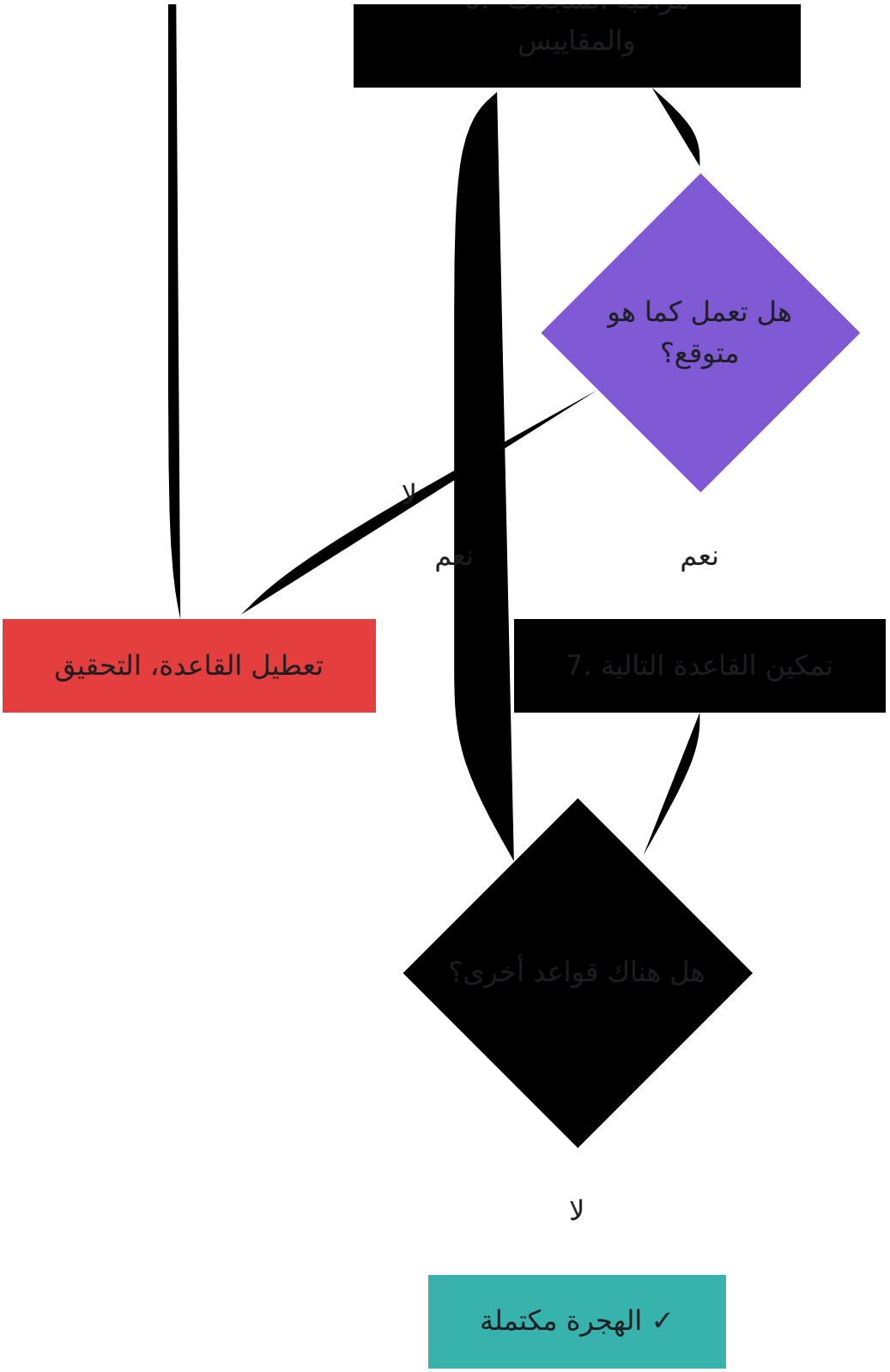
Parse error on line 25: ... style New fill:#3182CE style P ----------^ Expecting 'SOLID_OPEN_ARROW', 'DOTTED_OPEN_ARROW', 'SOLID_ARROW', 'BIDIRECTIONAL_SOLID_ARROW', 'DOTTED_ARROW', 'BIDIRECTIONAL_DOTTED_ARROW', 'SOLID_CROSS', 'DOTTED_CROSS', 'SOLID_POINT', 'DOTTED_POINT', got 'TXT'

المحاولة مجددا

استراتيجيات الهجرة

عند نشر قواعد ترجمة جديدة:


تخطيط القواعد الجديدة
1. تصميم القواعد في وضع عدم الاتصال
2. اختبار في المحاكي
هل تعمل القواعد بشكل صحيح؟
لا
نعم
تصحيح الأنماط
3. إضافة القواعد كمعطلة
4. نشر في الإنتاج
5. تمكين القواعد واحدة تلو الأخرى



أمثلة

المثال 1: تطبيع رقم الولايات المتحدة

المتطلب: تحويل تنسيقات أرقام الولايات المتحدة المختلفة إلى E.164 (+1XXXXXXXXXX)

```
# القاعدة 1: أرقام مكونة من 10 أرقام (أعلى أولوية)
%{
  calling_match: "^(\d{10})$",
  calling_replace: "+1\1",
  called_match: "^(\d{10})$",
  called_replace: "+1\1",
  priority: 5,
  description: "إضافة +1 إلى الأرقام المكونة من 10 أرقام",
  enabled: true,
  continue: false
}

# القاعدة 2: 1 + 10 أرقام (أولوية متوسطة)
%{
  calling_match: "^1(\d{10})$",
  calling_replace: "+1\1",
  called_match: "^1(\d{10})$",
  called_replace: "+1\1",
  priority: 10,
  description: "تحويل 1XXXXXXXXXX إلى +1XXXXXXXXXX",
  enabled: true,
  continue: false
}

# حالات الاختبار:
# "5551234567" → "+15551234567" (القاعدة 1)
# "15551234567" → "+15551234567" (القاعدة 2)
# "+15551234567" → "+15551234567" (لا تطابق، تمر دون تغيير)
```

المثال 2: تحويل بادئة دولية مع السلسلة

المتطلب: تحويل بادئة 00 إلى +، ثم تنسيق للبوابة

```
# القاعدة 1: تحويل 00 إلى + (تستمر إلى القاعدة التالية)
%{
  calling_match: "^00(.+)$",
  calling_replace: "+\1",
  called_match: "^00(.+)$",
  called_replace: "+\1",
  priority: 5,
  description: "تحويل بادئة 00 الدولية إلى +",
  enabled: true,
  continue: true  # الاستمرار في التنسيق
}

# القاعدة 2: تنسيق للبوابة (تتوقف عن المعالجة)
%{
  calling_match: "^\+(\d+)$",
  calling_replace: "00\1",
  called_match: "^\+(\d+)$",
  called_replace: "00\1",
  priority: 10,
  description: "تنسيق الأرقام + كـ 00 للبوابة",
  enabled: true,
  continue: false  # التوقف بعد ذلك
}

# حالة الاختبار:
# الخطوة 1: "441234567890+" → "00441234567890" (القاعدة 1، تستمر)
# الخطوة 2: "00441234567890" → "441234567890+" (القاعدة 2، تتوقف)
# النتيجة: "00441234567890"
# القواعد المطبقة: [القاعدة 1، القاعدة 2]
```

المثال 3: معالجة خاصة بـ SMSC

المصدر SMSC **المتطلب**: تطبيق قواعد مختلفة بناءً على

```
# القاعدة 1: SMSC (الأولوية 5) تمر دون تغيير - موثوق
%{
  source_smsc: "trusted_gateway",
  calling_match: nil,  # لا تحويل
  calling_replace: nil,
  called_match: nil,
  called_replace: nil,
  priority: 5,
  description: "تمر الأرقام من البوابة الموثوقة",
  enabled: true,
  continue: false
}

# القاعدة 2: SMSC (الأولوية 10) تطبيع - غير موثوق
%{
  source_smsc: "untrusted_gateway",
  calling_match: "^(.*)$",
  calling_replace: "+VALIDATE\1",
  called_match: "^(.*)$",
  called_replace: "+VALIDATE\1",
  priority: 10,
  description: "إضافة بادئة التحقق للمصدر غير الموثوق",
  enabled: true,
  continue: false
}

# القاعدة 3: قاعدة شاملة لبقية SMSCs (الأولوية 100)
%{
  source_smsc: nil,  # حرف بدل
  calling_match: "^(\d{10})$",
  calling_replace: "+1\1",
  called_match: "^(\d{10})$",
  called_replace: "+1\1",
  priority: 100,
  description: "افتراضي: إضافة +1 إلى الأرقام المكونة من 10 أرقام",
  enabled: true,
  continue: false
}
```

المثال 4: سلسلة تنسيق متعددة الخطوات

المتطلب: تطبيع ← إضافة رمز البلد ← تنسيق مع شرطات

```
# القاعدة 1: إزالة الأصفار البادئة (تستمر)
%{
  calling_match: "^0+(.+)$",
  calling_replace: "\1",
  called_match: "^0+(.+)$",
  called_replace: "\1",
  priority: 5,
  description: "إزالة الأصفار البادئة",
  enabled: true,
  continue: true
}

# القاعدة 2: إضافة رمز البلد إذا كان مفقودًا (تستمر)
%{
  calling_match: "^(\d{10})$",
  calling_replace: "+1\1",
  called_match: "^(\d{10})$",
  called_replace: "+1\1",
  priority: 10,
  description: "إضافة +1 إلى الأرقام المكونة من 10 أرقام",
  enabled: true,
  continue: true
}

# القاعدة 3: تنسيق مع شرطات (تتوقف)
%{
  calling_match: "^\+1(\d{3})(\d{3})(\d{4})$",
  calling_replace: "+1-\1-\2-\3",
  called_match: "^\+1(\d{3})(\d{3})(\d{4})$",
  called_replace: "+1-\1-\2-\3",
  priority: 15,
  description: "تنسيق كـ +1-XXX-XXX-XXXX",
  enabled: true,
  continue: false
}

# حالة ا��اختبار:
# الإدخال: "005551234567"
# الخطوة 1: "005551234567" → "5551234567" (القاعدة 1، تستمر)
# الخطوة 2: "5551234567" → "+15551234567" (القاعدة 2، تستمر)
# الخطوة 3: "+15551234567" → "+1-555-123-4567" (القاعدة 3، تتوقف)
```

```
# 4567-123-555-1+" :النتيجة"
# [3 القاعدة ،2 القاعدة ،1 القاعدة :القواعد المطبقة
```

الدعم

للمشاكل أو الأسئلة:

- تحقق من مجموعة الاختبار في `test/sms_c/messaging/number_translation_test.exs` للحصول على أمثلة
- استخدم المحاكي لتصحيح منطق الترجمة
- راجع سجلات الأحداث لقرارات الترجمة
- تحقق من محتويات جدول Mnesia: `:mnesia.table_info(:translation_rule, :size)`
- راقب مقاييس Telemetry لمشاكل الأداء

دليل عمليات SMS-C

العودة إلى فهرس الوثائق | **الملف التعريفي الرئيسي ←**

إجراءات التشغيل اليومية، والمراقبة، ومهام الصيانة لفرق عمليات SMS-C.

جدول المحتويات

- العمليات اليومية
- المراقبة
- تتبع الرسائل
- إدارة المسارات
- إدارة الواجهة الأمامية
- إدارة ترجمة الأرقام
- صيانة النظام
- النسخ الاحتياطي والاستعادة
- تخطيط السعة
- استجابة الحوادث

العمليات اليومية

فحص الصحة الصباحية

قم بإجراء هذه الفحوصات في بداية كل يوم:

1. تحقق من حالة النظام

```
# فحص صحة API
curl https://api.example.com:8443/api/status

# الاستجابة المتوقعة:
# {"status":"ok","application":"OmniMessage","timestamp":"2025-10-30T08:00:00Z"}
```

2. مراجعة مقاييس Prometheus

الوصول إلى لوحة معلومات Prometheus والتحقق من:

- معدل مرور الرسائل (آخر 24 ساعة)
- معدل فشل التوجيه (يجب أن يكون < 1%)
- تراكم قائمة الانتظار (يجب أن يكون < 1000 معلق)
- معدل نجاح التسليم (يجب أن يكون > 95%)
- حالة اتصال الواجهة الأمامية (جميع الواجهات الأمامية المتوقعة نشطة)

3. تحقق من قائمة انتظار الرسائل

الوصول إلى واجهة الويب: `https://sms-admin.example.com/message_queue`

مراجعة:

- إجمالي الرسائل المعلقة (يجب أن يكون منخفضًا)
- عمر أقدم رسالة (يجب أن يكون < 5 دقائق)
- الرسائل ذات محاولات التسليم العالية (تحقق إذا كانت > 3)
- الرسائل الميتة (تحقق من أي موجودة)

4. مراجعة حالة الواجهة الأمامية

الوصول إلى واجهة الويب: `https://sms-admin.example.com/frontend_status`

تحقق من:

- جميع الواجهات الأمامية المتوقعة نشطة
- لا توجد انقطاعات غير منتهية
- لا توجد أخطاء في الواجهة الأمامية في آخر 24 ساعة

5. تحقق من سجلات التطبيق

الوصول إلى واجهة الويب: `https://sms-admin.example.com/logs` أو تحقق من ملفات السجل

ابحث عن:

- رسائل مستوى الخطأ

- فشل التوجيه
- فشل الشحن
- مشاكل اتصال قاعدة البيانات
- مشاكل عقد الكتلة

مراقبة حجم الرسائل

تحقق من عدد الرسائل كل ساعة:

استخدم استعلام Prometheus:

```
# الرسائل المستلمة في الساعة
increase(sms_c_message_received_count[1h])

# الرسائل المرسلة في الساعة
increase(sms_c_delivery_succeeded_count[1h])

# حساب معدل التسليم
rate(sms_c_delivery_succeeded_count[1h]) /
rate(sms_c_message_received_count[1h])
```

الأنماط المتوقعة:

- ساعات العمل: حجم أعلى
- ليالي/عطلات نهاية الأسبوع: حجم أقل
- معدل التسليم: يجب أن يكون > 95%

شروط التنبيه:

- انخفاض مفاجئ في الرسائل (> 50% انخفاض)
- زيادة مفاجئة في الرسائل (> 200% زيادة)
- انخفاض معدل التسليم تحت 90%

المراقبة

المقاييس الرئيسية للمراقبة

مقاييس معالجة الرسائل

عدد الرسائل المستلمة (`sms_c_message_received_count`):

- **ما هو**: إجمالي الرسائل التي تدخل النظام
- **ت��بيه**: انخفاض أو زيادة مفاجئة
- **استعلام**: `rate(sms_c_message_received_count[5m])`

مدة معالجة الرسالة (`sms_c_message_processing_stop_duration`):

- **ما هو**: وقت المعالجة من البداية إلى النهاية
- **تنبيه**: p95 > 1000ms
- **استعلام**: `histogram_quantile(0.95, sms_c_message_processing_stop_duration)`

مقاييس التوجيه

فشل التوجيه (`sms_c_routing_failed_count`):

- **ما هو**: الرسائل التي لم يمكن توجيهها
- **تنبيه**: أي فشل (> 0)
- **استعلام**: `increase(sms_c_routing_failed_count[5m])`

المسار المطابق (`sms_c_routing_route_matched_count`):

- **ما هو**: أي المسارات يتم استخدامها
- **تنبيه**: المسارات ذات الأولوية العالية لا تطابق
- **استعلام**: `sms_c_routing_route_matched_count`

مقاييس التسليم

معدل نجاح التسليم:

- **ما هو**: نسبة التسليم الناجح

- **تنبيه**: معدل < 95%
- **استعلام**: `rate(sms_c_delivery_succeeded_count[5m]) / rate(sms_c_delivery_queued_count[5m])`

محاولات التسليم (`sms_c_delivery_succeeded_attempt_count`):

- **ما هو**: عدد المحاولات ��للازمة للتسليم
- **تنبيه**: p95 > 2 (الكثير من المحاولات)
- **استعلام**: `histogram_quantile(0.95, sms_c_delivery_succeeded_attempt_count)`

مقاييس قائمة الانتظار

حجم قائمة الانتظار (`sms_c_queue_size_size`):

- **ما هو**: إجمالي الرسائل في قائمة الانتظار
- **تنبيه**: الحجم > 10,000
- **استعلام**: `sms_c_queue_size_size`

عمر أقدم رسالة (`sms_c_queue_oldest_message_age_seconds`):

- **ما هو**: عمر أقدم رسالة معلقة
- **تنبيه**: العمر > 300 ثانية
- **استعلام**: `sms_c_queue_oldest_message_age_seconds`

إعداد لوحة المعلومات

لوحات معلومات العمليات:

1. **معدل مرور الرسائل** (رسم بياني)
    - الرسائل المستلمة (معدل 5 دقائق)
    - الرسائل المرسلة (معدل 5 دقائق)
    - نطاق الوقت: آخر 24 ساعة
2. **حالة قائمة الانتظار** (إحصائيات فردية)
    - الرسائل المعلقة الحالية

- عمر أقدم رسالة
- عدد الرسائل الفاشلة

3. (رسم بياني) **أداء التسليم**

- معدل النجاح بمرور الوقت
- معدل ال��شل بمرور الوقت
- نطاق الوقت: آخر 24 ساعة

4. (جدول) **حالة التوجيه**

- معرف المسار
- عدد المطابقات (الساعة الأخيرة)
- SMSC الوجهة
- الأولوية

5. (جدول) **حالة الواجهة الأمامية**

- اسم الواجهة الأمامية
- الحالة (نشط/منتهي)
- آخر ظهور
- عدد الرسائل (الساعة الأخيرة)

6. (إحصائيات فردية) **صحة النظام**

- زمن استجابة API (p95)
- زمن استعلام قاعدة البيانات (p95)
- زمن بحث ENUM (p95)

إعداد التنبيهات

التنبيهات الحرجة (استجابة فورية مطلوبة):

```yaml
# لا يوجد مسار موجود - لا يمكن تسليم الرسائل
- alert: RoutingFailures
  expr: increase(sms_c_routing_failed_count[5m]) > 0
  severity: critical
  description: "{{ $value }} messages failed routing in last 5
minutes"

# تراكم قائمة الانتظار - المعالجة تتأخر
- alert: QueueBacklog
  expr: sms_c_queue_size_pending > 10000
  severity: critical
  description: "Queue has {{ $value }} pending messages"

# الرسائل تتقدم في العمر - التسليم متوقف
- alert: OldMessagesInQueue
  expr: sms_c_queue_oldest_message_age_seconds > 300
  severity: critical
  description: "Oldest message is {{ $value }} seconds old"

# الواجهة الأمامية مفصولة - لا يوجد مسار تسليم
- alert: FrontendDisconnected
  expr: sms_c_frontend_status_count{status="disconnected"} > 0
  severity: critical
  description: "{{ $value }} frontends disconnected"
```

تنبيهات التحذير (تحتاج إلى تحقيق):

```
# معدل نجاح التسليم يتناقص
- alert: LowDeliveryRate
  expr: rate(sms_c_delivery_succeeded_count[10m]) /
rate(sms_c_delivery_queued_count[10m]) < 0.90
  severity: warning
  description: "Delivery success rate is {{ $value }}"

# الكثير من محاولات التسليم
- alert: HighRetryRate
  expr: histogram_quantile(0.95,
sms_c_delivery_succeeded_attempt_count) > 2
  severity: warning
  description: "95th percentile delivery attempts: {{ $value }}"

# بحث ENUM بطيء أو فاشل
- alert: SlowEnumLookups
  expr: histogram_quantile(0.95, sms_c_enum_lookup_stop_duration)
> 5000
  severity: warning
  description: "ENUM lookups taking > 5 seconds"

# منخفض ENUM معدل نجاح ذاكرة التخزين المؤقت
- alert: LowEnumCacheHitRate
  expr: rate(sms_c_enum_cache_hit_count[10m]) /
(rate(sms_c_enum_cache_hit_count[10m]) +
rate(sms_c_enum_cache_miss_count[10m])) < 0.70
  severity: warning
  description: "ENUM cache hit rate: {{ $value }}"
```

تتبع الرسائل

العثور على رسالة معينة

حسب معرف الرسالة:

1. انتقل إلى **واجهة الويب**: `/message_queue`
2. أدخل معرف الرسالة في مربع البحث
3. عرض التفاصيل الكاملة وسجل الأحداث

عبر API:

```
curl https://api.example.com:8443/api/messages/12345
```

حسب رقم الهاتف:

1. انتقل إلى **واجهة الويب**: `/message_queue`
2. أدخل رقم الهاتف في مربع البحث
3. عرض جميع الرسائل لذلك الرقم

تتبع دورة حياة الرسالة

عرض سجل الأحداث:

1. انقر على الرسالة في قائمة الانتظار، عرض قسم "الأحداث" في **واجهة الويب**:
2. **API**: `GET /api/events/12345`

تسلسل الأحداث الشائع:

```
1. message_inserted - تم إنشاء الرسالة
   ↓
2. number_translated - تم تطبيع الأرقام (إذا تم تكوينه)
   ↓
3. message_routed - تم اتخاذ قرار التوجيه
   ↓
4. charging_attempted - تحقق من الشحن (إذ تم تمكينه)
   ↓
5. message_delivered - تم التسليم بنجاح
```

تسلسل التسليم الفاشل:

```
1. message_inserted
   ↓
2. message_routed
   ↓
3. delivery_attempt_1 - المحاولة الأولى فشلت
   ↓
4. delivery_attempt_2 - المحاولة الثانية فشلت (تأخير 2 دقيقة)
   ↓
5. delivery_attempt_3 - المحاولة الثالثة فشلت (تأخير 4 دقائق)
   ↓
6. message_dead_letter - تجاوز حد المحاولات
```

تحقق من حالة التسليم

الرسائل المعلقة:

- الحالة: "معلقة"
- deliver_after: طابع زمني مستقبلي
- delivery_attempts: 0 أو عدد منخفض

الرسائل المرسلة:

- الحالة: "تم التسليم"
- deliver_time: طابع زمني للتسليم
- dest_smsc: الواجهة الأمامية التي قامت بالتسليم

الرسائل الفاشلة:

- الحالة: "معلقة" مع عدد محاولات تسليم مرتفع
- deadletter: true (إذا انتهت)
- تحقق من سجل الأحداث لأسباب الفشل

توجيه الرسائل بناءً على الموقع

استرجاع الرسائل بناءً على الموقع، مما يسمح للواجهات الأم��مية باستلام SMS-C يدعم الرسائل الموجهة تلقائيًا للمشتركين المسجلين في موقعهم.

كيف يعمل:

عندما تستعلم واجهة أمامية عن الرسائل المعلقة باستخدام `get_messages_for_smsc(smsc_name)`، يعيد النظام الرسائل بطريقتين:

1. الرسائل التي يتطابق فيها `dest_smsc` صراحة مع اسم الواجهة الأمامية - **التوجيه الصريح**
2. الرسائل التي - **التوجيه بناءً على الموقع**:
   - `dest_smsc` هو `null` (لم يتم توجيهها صراحة)
   - لديه سجل موقع نشط `destination_msisdn`
   - حقل `location` للموقع يتطابق مع اسم الواجهة الأمامية
   - لم تنته صلاحية الموقع

سيناريو المثال:

مشترك برقم MSISDN `+447700900123` يسجل في الواجهة الأمامية `uk_gateway`:

```
# يسجل المشترك (ينشئ سجل موقع)
POST /api/locations
{
  "msisdn": "+447700900123",
  "imsi": "234150123456789",
  "location": "uk_gateway",
  "expires": "2025-11-01T12:00:00Z"
}
```

عندما تصل رسالة لهذا المشترك بدون توجيه صريح:

```
# تم تقديم الرسالة بدون dest_smsc
POST /api/messages
{
  "source_msisdn": "+15551234567",
  "destination_msisdn": "+447700900123",
  "message_body": "Hello",
  "source_smsc": "api"
  # ملاحظة: dest_smsc هو null
}
```

هذه الرسالة تلقائيًا عندما تستعلم `uk_gateway` ستتلقى الواجهة الأمامية:

```
# الواجهة الأمامية تستعلم عن الرسائل
GET /api/messages/queue?smsc=uk_gateway

# على الرغم من أن dest_smsc هو null تعيد الرسالة
# لأن المشترك الوجهة مسجل في uk_gateway
```

متطلبات الموقع:

لكي يعمل التوجيه بناءً على الموقع:

- يجب أن تحتوي جدول `locations` على إدخال لـ `destination_msisdn`
- يجب أن يتطابق حقل `location` مع اسم SMSC المستعلم
- يجب أن يكون طابع `expires` في المستقبل

مراقبة التوجيه بناءً على الموقع:

تحقق من سجلات المواقع:

```
# عبر API
GET /api/locations/{msisdn}

# تحقق مما إذا كان الموقع قد انتهى
# يجب أن يكون حقل expires > الوقت الحالي
```

المشاكل الشائعة:

- **الرسالة لم تسلم**: تحقق مما إذا كان الموقع قد انتهى
- **واجهة أمامية خاطئة**: تحقق من أن حقل `location` يتطابق مع اسم الواجهة الأمامية المتوقع
- **الموقع غير موجود**: قد يحتاج المشترك إلى إعادة التسجيل

التدخلات اليدوية

إعادة محاولة الرسالة الفاشلة:

```
# إعادة تعيين delivery_attempts و deliver_after
curl -X PATCH https://api.example.com:8443/api/messages/12345 \
  -H "Content-Type: application/json" \
  -d '{
    "delivery_attempts": 0,
    "deliver_after": "2025-10-30T12:00:00Z"
  }'
```

تغيير الوجهة:

```
# توجيه إلى SMSC مختلف
curl -X PATCH https://api.example.com:8443/api/messages/12345 \
  -H "Content-Type: application/json" \
  -d '{
    "dest_smsc": "backup_gateway"
  }'
```

حذف رسالة عالقة:

```
curl -X DELETE https://api.example.com:8443/api/messages/12345
```

إدارة المسارات

عرض المسارات الحالية

واجهة الويب: انتقل إلى `/sms_routing`

عبر API:

```
# قائمة بجميع المسارات
curl https://api.example.com:8443/api/routes
```

تحقق من استخدام المسار:

استعلام Prometheus:

```
# (الساعة الأخيرة) الرسائل الموجهة بواسطة كل مسار
increase(sms_c_routing_route_matched_count[1h])
```

إضافة مسار جديد

واجهة الويب:

1. انتقل إلى `/sms_routing`
2. انقر على "إضافة مسار جديد"
3. املأ الحقول:
   - **بادئة الاتصال**: بادئة رقم المصدر (اختياري)
   - **بادئة الاتصال**: بادئة رقم الوجهة (مطلوبة للتوجيه الجغرافي)
   - **المصدر SMSC**: فلتر النظام المصدر (اختياري)
   - **الوجهة SMSC**: بوابة الوجهة (مطلوبة ما لم يكن الرد التلقائي/الإسقاط)
   - **فقط على الشبكة**: تقييد هذا المسار إلى الوجهات على الشبكة فقط (يتطلب Diameter/HSS)
   - **الأولوية**: أولوية المسار (1-255، أقل = أولوية أعلى)
   - **الوزن**: وزن توزيع الحمل (1-100)
   - **الوصف**: وصف قابل للقراءة البشرية
   - **مفعل**: تحقق للتفعيل الفوري
4. انقر على "حفظ المسار"

مثال: مسار جغرافي:

- بادئة الاتصال: `44+`
- الوجهة SMSC: `uk_gateway`
- الأولوية: 50
- الوزن: 100
- وصف: "توجيه المملكة المتحدة"

مثال: مسار موزع الحمل:

إنشاء مسارين بنفس المعايير ولكن بأوزان مختلفة:

المسار 1:

- بادئة الاتصال: `44+`
- الوجهة SMSC: `uk_primary`
- الأولوية: 50
- الوزن: 70
- الوصف: "المملكة المتحدة الرئيسية (70%)"

المسار 2:

- بادئة الاتصال: `44+`
- الوجهة SMSC: `uk_backup`
- الأولوية: 50
- الوزن: 30
- الوصف: "المملكة المتحدة الاحتياطية (30%)"

مثال: مسار فقط على الشبكة:

تقييد ناقل خارجي أو تطبيق داخلي لإرسال الرسائل فقط إلى المشتركين على الشبكة. يتم إسقاط الرسائل إلى الوجهات خارج الشبكة.

- المصدر SMSC: `carrier_smpp_bind`
- الوجهة SMSC: `local_msc`
- فقط على الشبكة: تحقق
- الأولوية: 50
- الوزن: 100
- الوصف: "الناقل X — فقط على الشبكة"

يتطلب ذلك تمكين Diameter/HSS. إذا كان HSS (فشل مغلق) غير متاح، يتم إسقاط الرسائل.

اختبار المسارات

محاكي التوجيه:

1. انتقل إلى `/simulator`
2. أدخل معلمات الاختبار:
   - رقم الاتصال: `+15551234567`
   - رقم الاتصال: `+447700900000`

    - المصدر: (اختياري) SMSC
    - نوع المصدر: (اختياري)
3. انقر على "محاكاة التوجيه"
4. مراجعة النتائج:
    - **المسار المحدد**: أي مسار تم اختياره
    - **جميع المطابقات**: أي المسارات تطابقت مع المعايير
    - **التقييم**: لماذا تطابق كل مسار أو لم يتطابق

اختبر قبل الإنتاج:

- اختبر جميع المسارات الجديدة في المحاكي
- تحقق من اختيار المسار الصحيح
- تحقق من ترتيب الأولويات
- تحقق من توزيع الوزن

تعديل مسار موجود

واجهة الويب:

1. انتقل إلى `/sms_routing`
2. ابحث عن المسار في القائمة
3. انقر على "تحرير"
4. تعديل الحقول
5. انقر على "حفظ المسار"

التعديلات الشائعة:

- **تعطيل المسار**: قم بإلغاء تحديد "مفعل" (إزالة مؤقتة)
- **تعديل الوزن**: تغيير توزيع الحمل
- **تغيير الأولوية**: إعادة ترتيب تقييم المسار
- **تحديث الوجهة**: التبديل إلى SMSC مختلف

حذف مسار

واجهة الويب:

1. انتقل إلى `/sms_routing`

2. ابحث عن المسار في القائمة
3. انقر على "حذف"
4. تأكيد الحذف

تحذير: حذف المسارات دائم. اعتبر تعطيلها بدلاً من ذلك.

تصدير/استيراد المسارات

تصدير المسارات (نسخة احتياطية):

1. انتقل إلى `/sms_routing`
2. انقر على "تصدير المسارات"
3. احفظ ملف JSON

استيراد المسارات:

1. انتقل إلى `/sms_routing`
2. انقر على "استيراد المسارات"
3. حدد ملف JSON
4. اختر وضع الاستيراد:
    - **دمج**: إضافة إلى المسارات الموجودة
    - **استبدال**: حذف الكل والاستيراد

حالات الاستخدام:

- النسخ الاحتياطي قبل التغييرات الكبيرة
- نسخ المسارات بين البيئات
- استعادة الكوارث
- إصدار التكوين

إدارة الواجهة الأمامية

مراقبة اتصالات الواجهة الأمامية

واجهة الويب: انتقل إلى `/frontend_status`

تحقق من:

- جميع الواجهات الأمامية المتوقعة "نشطة"
- أوقات آخر ظهور حديثة (< 90 ثانية)
- لا توجد واجهات أمامية منتهية غير متوقعة

عبر API:

```
# الحصول على الواجهات الأمامية النشطة
curl https://api.example.com:8443/api/frontends/active

# الحصول على الإحصائيات
curl https://api.example.com:8443/api/frontends/stats
```

التحقيق في الانقطاعات

الواجهة الأمامية منتهية:

1. تحقق من سجلات الواجهة الأمامية للأخطاء
2. تحقق من الاتصال الشبكي بـ SMS-C
3. تأكد من أن الواجهة الأمامية تعمل
4. تحقق من منطق تسجيل الواجهة الأمامية (يجب أن تعيد التسجيل كل 60 ثانية)

التسجيل لا يظهر:

1. تحقق من أن الواجهة الأمامية تستدعي POST `/api/frontends/register`
2. تحقق من سجلات API لأخطاء التسجيل
3. تحقق من تنسيق الحمولة JSON
4. اختبر التسجيل يدويًا باستخدام curl

مثال على التسجيل اليدوي:

```
curl -X POST https://api.example.com:8443/api/frontends/register \
  -H "Content-Type: application/json" \
  -d '{
    "frontend_name": "test_gateway",
    "frontend_type": "smpp",
    "ip_address": "10.0.1.50",
    "hostname": "gateway.example.com"
  }'
```

عرض تاريخ الواجهة الأمامية

واجهة الويب:

1. انتقل إلى `/frontend_status`
2. ابحث عن الواجهة الأمامية في القائمة
3. انقر على "التاريخ"
4. مراجعة التسجيلات السابقة

عبر API:

```
curl https://api.example.com:8443/api/frontends/history/uk_gateway
```

حالات الاستخدام:

- التحقيق في موثوقية الاتصال
- تتبع أنماط وقت تشغيل الواجهة الأمامية
- تحديد تغييرات التكوين

إدارة ترجمة الأرقام

تتم إدارة قواعد ترجمة الأرقام عبر `config/runtime.exs`. تتطلب التغييرات إعادة تشغيل التطبيق.

عرض قواعد الترجمة النشطة

تحقق من ملف التكوين:

```
cat config/runtime.exs | grep -A 20 "translation_rules:"
```

المهام الشائعة لترجمة الأرقام

إضافة رمز الدولة إلى الأرقام المحلية:

قم بتحرير `config/runtime.exs`:

```
%{
  calling_prefix: nil,
  called_prefix: nil,
  source_smsc: nil,
  calling_match: "^(\d{10})$",
  calling_replace: "+1\1",
  called_match: "^(\d{10})$",
  called_replace: "+1\1",
  priority: 100,
  description: "إضافة +1 إلى الأرقام الأمريكية المكونة من 10 أرقام",
  enabled: true
}
```

تطبيع التنسيق الدولي:

```
%{
  calling_prefix: nil,
  called_prefix: nil,
  source_smsc: nil,
  calling_match: "^00(\d+)$",
  calling_replace: "+\1",
  called_match: "^00(\d+)$",
  called_replace: "+\1",
  priority: 10,
  description: "تحويل بادئة 00 إلى +",
  enabled: true
}
```

إزالة رمز الناقل المحدد:

```
%{
  calling_prefix: nil,
  called_prefix: "101",
  source_smsc: "carrier_a",
  calling_match: nil,
  calling_replace: nil,
  called_match: "^101(\d+)$",
  called_replace: "\1",
  priority: 5,
  description: "إزالة رمز الناقل من الناقل A",
  enabled: true
}
```

اختبار قواعد الترجمة

بعد تغييرات التكوين:

1. أعد تشغيل التطبيق لتحميل القواعد الجديدة
2. قدم رسالة اختبار بمصدر/وجهة يجب أن تتطابق
3. تحقق من سجل الأحداث لحدث `number_translated`
4. تحقق من أن الأرقام تم تحويلها بشكل صحيح

تعطيل قاعدة الترجمة

قم بتعيين `enabled: false` في القاعدة:

```
%{
  ...
  enabled: false
}
```

أعد تشغيل التطبيق.

صيانة النظام

صيانة قاعدة البيانات

تحقق من حجم قاعدة البيانات:

استخدم أدوات إدارة قاعدة البيانات الخاصة بك لمراقبة حجم تخزين CDR:

- **MySQL/MariaDB**: استعلام `information_schema.tables` لحجم قاعدة البيانات
- **PostgreSQL**: استخدم دالة `pg_database_size()` أو أمر `\l+` في psql

تنظيف سجلات CDR القديمة:

يجب أرشفة سجلات CDR وحذفها دوريًا بناءً على سياسة الاحتفاظ الخاصة بك:

- تكوين الأرشفة التلقائية بناءً على متطلبات العمل (عادةً 30-90 يومًا في قاعدة البيانات التشغيلية)
- أرشفة السجلات القديمة إلى مستودع البيانات أو التخزين البارد
- حذف السجلات المؤرشفة من قاعدة البيانات التشغيلية في دفعات لتجنب التنافس على القفل

تحسين الجداول:

قم بتحسين جداول قاعدة البيانات دوريًا للحفاظ على الأداء:

- **MySQL/MariaDB**: قم بتشغيل أمر `OPTIMIZE TABLE` خلال فترات انخفاض الحركة
- **PostgreSQL**: قم بتشغيل `VACUUM ANALYZE` بانتظام (أو تمكين autovacuum)

تشغيل أسبوعيًا خلال فترة انخفاض الحركة للحفاظ على الأداء الأمثل.

صيانة قاعدة بيانات Mnesia

تحقق من حجم جدول Mnesia:

```
# في وحدة التحكم IEx
:mnesia.table_info(:sms_route, :size)
:mnesia.table_info(:translation_rule, :size)
```

نسخ احتياطي لجداول Mnesia:

```
# تصدير المسارات (واجهة الويب)
# انتقل إلى /sms_routing
# انقر على "تصدير المسارات"

# أو عبر النسخ الاحتياطي لـ Mnesia
:mnesia.backup("/var/backups/sms_c/mnesia_backup.bup")
```

استعادة Mnesia:

```
# عبر استيراد واجهة الويب
# أو استعادة النسخ الاحتياطي:
:mnesia.restore("/var/backups/sms_c/mnesia_backup.bup", [])
```

تدوير السجلات

تكوين logrotate لسجلات التطبيق:

```
# /etc/logrotate.d/sms_c
/var/log/sms_c/*.log {
    daily
    rotate 30
    compress
    delaycompress
    notifempty
    create 0644 sms_user sms_group
    sharedscripts
    postrotate
        systemctl reload sms_c || true
    endscript
}
```

إعادة تشغيل التطبيق

إعادة تشغيل سلسة (زمن تعطل صفر في الكتلة):

```
# إعادة تشغيل عقدة واحدة في كل مرة
systemctl restart sms_c

# انتظر حتى تنضم العقدة إلى الكتلة
# كرر لكل عقدة
```

إعادة تشغيل طارئة (جميع العقد):

```
systemctl restart sms_c
```

بعد إعادة التشغيل:

- تحقق من إعادة اتصال جميع الواجهات الأمامية
- تحقق من Prometheus لاستمرارية المقاييس
- راقب السجلات بحثًا عن الأخطاء
- تحقق من استئناف معالجة الرسائل

النسخ الاحتياطي والاستعادة

ما يجب نسخه احتياطيًا

1. **ملفات التكوين**:
   - `config/runtime.exs`
   - `config/config.exs`
   - `config/prod.exs` (إذا كان موجودًا)
2. **جداول التوجيه** (Mnesia):
   - تصدير عبر واجهة الويب
   - أو أمر النسخ الاحتياطي لـ Mnesia

3. **قاعدة بيانات CDR SQL**:
    - نسخة احتياطية كاملة يومية
    - نسخ احتياطية لسجل المعاملات (مستمرة)
4. **شهادات TLS**:
    - `priv/cert/*.crt`
    - `priv/cert/*.key`

إجراءات النسخ الاحتياطي

نسخة احتياطية يومية للتكوين:

```
#!/bin/bash
# /opt/sms_c/scripts/backup_config.sh

BACKUP_DIR="/var/backups/sms_c/$(date +%Y%m%d)"
mkdir -p $BACKUP_DIR

# النسخ الاحتياطي للتكوين
cp -r /opt/sms_c/config $BACKUP_DIR/

# النسخ الاحتياطي للشهادات
cp -r /opt/sms_c/priv/cert $BACKUP_DIR/

# تعيين الأذونات
chmod 600 $BACKUP_DIR/cert/*

echo "اكتمل النسخ الاحتياطي للتكوين: $BACKUP_DIR"
```

نسخة احتياطية لقاعدة البيانات:

```
#!/bin/bash
# /opt/sms_c/scripts/backup_database.sh

BACKUP_DIR="/var/backups/sms_c/database"
DATE=$(date +%Y%m%d_%H%M%S)

mkdir -p $BACKUP_DIR

# النسخ الاحتياطي لقاعدة بيانات CDR SQL
# MySQL/MariaDB: استخدم mysqldump مع --single-transaction لضمان التناسق
# PostgreSQL: استخدم pg_dump -F c للتنسيق المخصص

# هيكلية المثال (تكييفها مع قاعدة بياناتك):
# - استخدم أداة النسخ الاحتياطي المناسبة (mysqldump، pg_dump)
# - تمكين النسخ الاحتياطي الآمن للمعاملات لضمان التناسق
# - ضغط الإخراج لتوفير المساحة
# - تكوين فترة الاحتفاظ (مثل 30 يومًا)

# إزالة النسخ الاحتياطية القديمة
find $BACKUP_DIR -name "sms_c_*.gz" -mtime +30 -delete

echo "اكتمل النسخ الاحتياطي لقاعدة البيانات: sms_c_${DATE}"
```

نسخة احتياطية لجداول التوجيه:

```
#!/bin/bash
# /opt/sms_c/scripts/backup_routes.sh

BACKUP_DIR="/var/backups/sms_c/routes"
DATE=$(date +%Y%m%d)

mkdir -p $BACKUP_DIR

# تصدير عبر API
curl https://api.example.com:8443/api/routes/export \
  > $BACKUP_DIR/routes_${DATE}.json

echo "اكتمل النسخ الاحتياطي للمسارات: routes_${DATE}.json"
```

جدولة النسخ الاحتياطي (crontab):

```
# يوميًا في الساعة 2 صباحًا
0 2 * * * /opt/sms_c/scripts/backup_config.sh
0 2 * * * /opt/sms_c/scripts/backup_database.sh
0 2 * * * /opt/sms_c/scripts/backup_routes.sh
```

إجراءات الاستعادة

استعادة التكوين:

```
# إيقاف التطبيق
systemctl stop sms_c

# استعادة ملفات التكوين
cp -r /var/backups/sms_c/20251030/config/* /opt/sms_c/config/

# استعادة الشهادات
cp -r /var/backups/sms_c/20251030/cert/* /opt/sms_c/priv/cert/

# بدء التطبيق
systemctl start sms_c
```

استعادة قاعدة بيانات CDR SQL:

استخدم أدوات الاستعادة المناسبة لقاعدة بياناتك:

- **MySQL/MariaDB**: فك الضغط وتوجيهه إلى عميل mysql
- **PostgreSQL**: استخدم pg_restore مع النسخ الاحتياطية بالتنسيق المخصص

مهم: أوقف تطبيق SMS-C قبل استعادة قاعدة البيانات لتجنب تعارض البيانات.

استعادة جداول التوجيه:

1. انتقل إلى واجهة الويب `/sms_routing`
2. انقر على "استيراد المسارات"
3. حدد ملف JSON الاحتياطي
4. اختر وضع "استبدال"
5. تأكيد الاستيراد

تخطيط السعة

مراقبة اتجاهات النمو

اتجاه حجم الرسائل:

استعلام Prometheus (متوسط 30 يومًا):

```
avg_over_time(sms_c_message_received_count[30d])
```

معدل نمو قاعدة البيانات:

```sql
-- نمو البيانات الشهري
SELECT
  DATE_FORMAT(inserted_at, '%Y-%m') AS month,
  COUNT(*) AS message_count,
  ROUND(SUM(LENGTH(message_body)) / 1024 / 1024, 2) AS data_mb
FROM message_queues
GROUP BY month
ORDER BY month DESC
LIMIT 12;
```

مؤشرات السعة

استخدام وحدة المعالجة المركزية:

- **طبيعي**: متوسط < 50%
- **مرتفع**: مستمر < 70%
- **حرج**: < 90%

استخدام الذاكرة:

- **طبيعي**: < 70% من المتاح
- **مرتفع**: < 80%
- **حرج**: < 90%

استخدام القرص:

- **طبيعي**: < 60% ممتلئ
- **مرتفع**: < 75%
- **حرج**: < 85%

عمق قائمة الانتظار:

- **طبيعي**: < 1000 معلق
- **مرتفع**: < 5000 معلق
- **حرج**: < 10,000 معلق

توصيات التوسع

متى يجب التوسع عموديًا (ترقية الموارد):

- وحدة المعالجة المركزية باستمرار > 70%
- الذاكرة باستمرار > 80%
- عنق زجاجة في عقدة واحدة

متى يجب التوسع أفقيًا (إضافة عقد):

- وحدة المعالجة المركزية > 50% في جميع العقد
- حجم الرسائل < 5,000 رسالة/ثانية
- الحاجة إلى توزيع جغرافي
- الحاجة إلى توافر عالي

توسيع قاعدة البيانات:

- نسخ قراءة للاستعلامات التقارير
- تحسين تجميع الاتصالات
- تحسين الفهارس
- تقسيم الجداول الكبيرة حسب التاريخ

استجابة الحوادث

مستويات الخطورة

حرج (استجابة فورية):

- لا يتم تسليم أي رسائل
- جميع الواجهات الأمامية مفصولة
- قاعدة البيانات غير متاحة
- API معطلة تمامًا

مرتفع (استجابة خلال ساعة واحدة):

- معدل نجاح التسليم < 80%
- عدة واجهات أمامية مفصولة
- فشل التوجيه > 10%
- تراكم قائمة الانتظار في تزايد

متوسط (استجابة خلال 4 ساعات):

- واجهة أمامية واحدة مفصولة
- معدل نجاح التسليم 80-95%
- معالجة الرسائل بطيئة
- فشل بحث ENUM

منخفض (استجابة خلال 24 ساعة):

- تدهور طفيف في الأداء
- مشكلة في مسار واحد
- تنبيهات تحذيرية غير حرجة

قائمة التحقق من الحوادث

1. تقييم الخطورة:

- تحقق من تنبيهات Prometheus
- مراجعة مقاييس لوحة المعلومات

- تحقق من حالة قائمة الانتظار للرسائل
- تحقق من اتصالات الواجهة الأمامية

2. **جمع المعلومات**:

- هل كانت هناك تغييرات في التكوين مؤخرًا؟
- هل كانت هناك نشرات حديثة؟
- حالة الاعتماديات الخارجية (OCS، DNS)؟
- رسائل الخطأ في السجلات؟

3. **الإجراءات الفورية**:

- إيقاف التغييرات الجارية
- التراجع عن النشرات الأخيرة إذا كانت السبب المشتبه به
- تمكين السجلات التفصيلية إذا لزم الأمر
- إبلاغ المعنيين

4. **التحقيق**:

- مراجعة سجلات التطبيق
- تحقق من استخدام موارد النظام
- فحص أداء قاعدة البيانات
- اختبار الاعتماديات الخارجية

5. **الحل**:

- تطبيق الإصلاح
- اختبار في المحاكي
- نشر في الإنتاج
- مراقبة التحسين

6. **بعد الحادث**:

- توثيق السبب الجذري
- تحديث المراقبة/التنبيهات
- تنفيذ تدابير وقائية
- تحديث كتيبات التشغيل

الحوادث الشائعة

تراكم قائمة الانتظار العالي:

1. تحقق من معدل نجاح التسليم
2. تحقق من أن ��لواجهات الأمامية متصلة وتستعلم
3. تحقق من أداء قاعدة البيانات
4. مراجعة Prometheus للزجاجات
5. النظر في زيادة حجم/فترة الدفعة

فشل التوجيه:

1. مراجعة تكوين التوجيه
2. اختبار في محاكي التوجيه
3. تحقق من وجود مسارات مفقودة
4. تحقق من وجود مسار شامل
5. تحقق من سجلات الأحداث لأسباب الفشل

انقطاعات الواجهة الأمامية:

1. تحقق من حالة نظام الواجهة الأمامية
2. تحقق من الاتصال الشبكي
3. مراجعة سجلات الواجهة الأمامية
4. اختبار التسجيل اليدوي عبر API
5. تحقق من قواعد جدار الحماية

معالجة الرسائل البطيئة:

1. تحقق من أداء استعلام قاعدة البيانات
2. مراجعة تكوين العاملين في الدفعات
3. تحقق من وجود موارد كافية (CPU/ذاكرة)
4. تحقق من تأخيرات بحث ENUM
5. مراجعة أداء نظام الشحن

لإجراءات استكشاف الأخطاء التفصيلية، راجع دليل استكشاف الأخطاء.

دليل تحسين الأداء

← الملف | العودة إلى فهرس الوثائق README الرئيسي

يشرح هذا الدليل كيفية تحسين أداء SMS-C لسيناريوهات العمل المختلفة.

نظرة عامة على الأداء

يقدم SMS-C **1,750 رسالة/ثانية** من خلال استخدام Mnesia لتخزين الرسائل في الذاكرة مع أرشفة قاعدة البيانات SQL التلقائية للاحتفاظ بسجلات CDR.

مقاييس الأداء الرئيسية

تم القياس على Intel i7-8650U @ 1.90GHz (8 نوى):

العملية	الإنتاجية	زمن الاستجابة (متوسط)	التحسين
إدراج رسالة (مع التوجيه)	1,750 رساله/ ثانية	0.58 ملي ثانية	أسرع 21 مرة من SQL
إدراج رسالة (بسيط)	1,750 رساله/ ثانية	0.57 ملي ثانية	أسرع 21 مرة من SQL
الحصول على الرسائل لـ SMSC	800 رساله/ ثانية	1.25 ملي ثانية	استعلام في الذاكرة
الذاكرة لكل إدراج	62 كيلوبايت	-	50 بنسبة تقليل%

السعة: ~150 مليون رسالة في اليوم على عقدة واحدة

جدول المحتويات

- معمارية تخزين الرسائل

- تحسين Mnesia
- تكوين أرشفة CDR
- تحسين الاستعلامات
- اختبار الأداء

معمارية تخزين الرسائل

معمارية تخزين مزدوجة لتحقيق أداء مثالي SMS-C يستخدم:

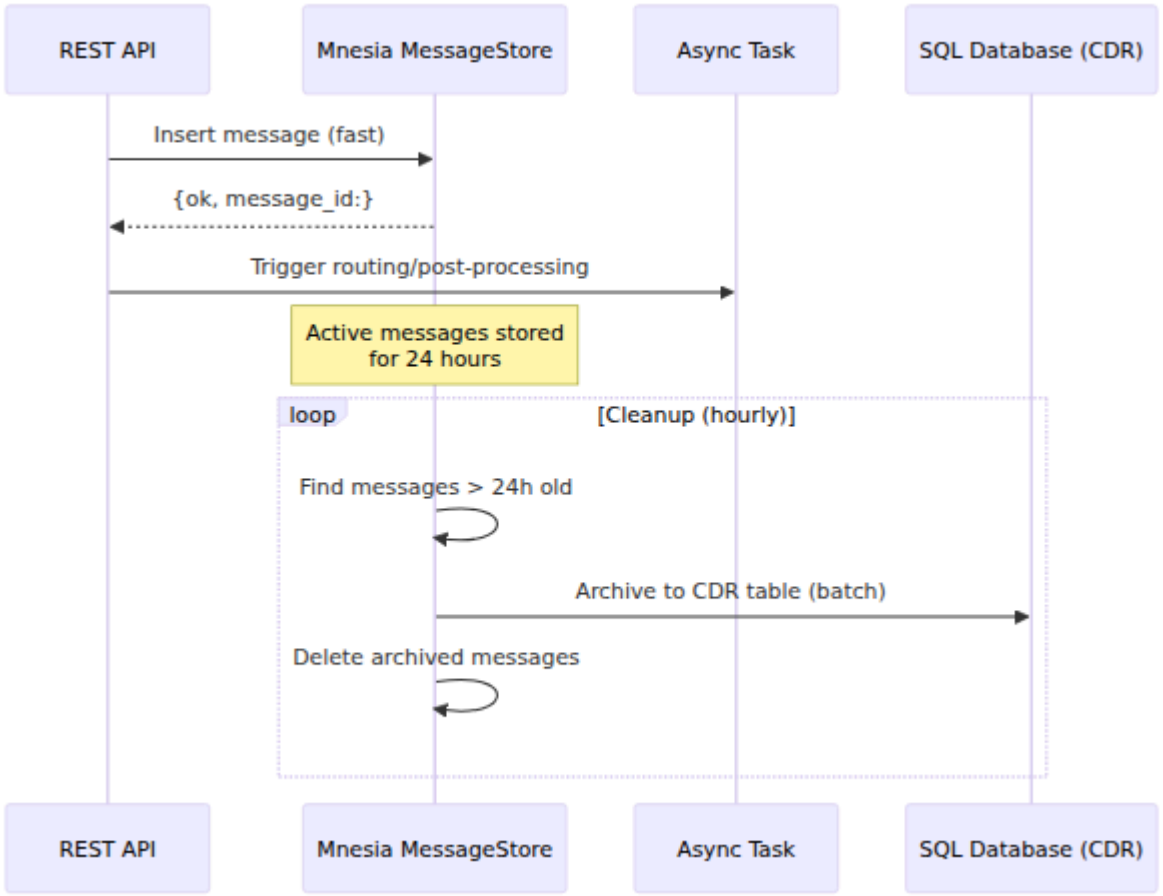
تخزين الرسائل النشطة (Mnesia)

- **الغرض**: إدراج الرسائل بسرعة فائقة، التوجيه، والتسليم
- **التخزين**: في الذاكرة مع استمرارية القرص (disc_copies)
- **الأداء**: 1,750 رساله/ثانية من الإنتاجية، 0.58مللي ثانية زمن الاستجابة
- **الاحتفاظ**: قابل للتكوين (الافتراضي: 24 ساعة)
- **التجميع**: يدعم Mnesia الموزعة للتوسع الأفقي

أرشيف CDR (قاعدة بيانات SQL)

- **الغرض**: تاريخ الرسائل على المدى الطويل والتقارير
- **التخزين**: قاعدة بيانات SQL (MySQL/MariaDB أو PostgreSQL) للأرشفة الدائمة
- **الأداء**: كتابة مجمعة لتقليل تحميل قاعدة البيانات
- **الاحتفاظ**: دائم (أو وفقًا لسياسة الاحتفاظ بالبيانات)
- **الاستعلامات**: التحليلات، التقارير، الامتثال

تدفق البيانات



تحسين Mnesia

تكوين الاحتفاظ بالرسائل

```
# config/runtime.exs
config :sms_c,
  message_retention_hours: 24  # Default: 24 hours
```

إرشادات الضبط:

- **حجم عالي (>1M رسالة/يوم)**: احتفاظ 12-24 ساعة
  - يقلل من حجم جدول Mnesia
  - استعلامات أسرع
  - أرشفة أكثر تكرارًا إلى MySQL

- **حجم متوسط (100K-1M رساله/يوم)**: 48-24 ساعة احتفاظ
    - توازن جيد لمعظم النشر
    - حافة كافية لمنطق إعادة المحاولة
- **حجم منخفض (<100K رساله/يوم)**: 168-48 ساعة احتفاظ
    - تاريخ رسائل أطول في تخزين سريع
    - أرشفة أقل تكرارًا

مؤشرات جدول Mnesia

يقوم MessageStore تلقائيًا بإنشاء مؤشرات على:

- `status` - لتصفية الرسائل المعلقة/المسلمة
- `dest_smsc` - لاستعلامات محددة لـ SMSC
- `expires` - لمعالجة انتهاء الصلاحية
- `destination_msisdn` - لاستعلامات المشتركين
- `source_msisdn` - لاستعلامات المشتركين

استمرارية قرص Mnesia

تُخزن الرسائل كـ `disc_copies` مما يوفر:

- □ أداء في الذاكرة
- □ استمرارية تلقائية على القرص
- □ استرداد من الأعطال
- □ عدم فقدان البيانات عند إعادة التشغيل

تكوين أرشفة CDR

يتولى `BatchInsertWorker` أرشفة CDR إلى MySQL باستخدام الكتابات المجمعة:

```
# config/runtime.exs
config :sms_c,
  batch_insert_batch_size: 100,           # CDR batch size
  batch_insert_flush_interval_ms: 100     # Auto-flush interval
```

إرشادات ضبط CDR

أرشفة عالية الحجم

```
batch_insert_batch_size: 200
batch_insert_flush_interval_ms: 200
```

- الأحجام الأكبر تقلل من تحميل MySQL
- زمن استجابة أعلى لكتابات CDR (مقبول للأرشفة)

متوازن (موصى به)

```
batch_insert_batch_size: 100
batch_insert_flush_interval_ms: 100
```

- توازن جيد لمعظم النشر
- يتم كتابة CDRs في غضون 100مللي ثانية

متطلبات CDR في الوقت الحقيقي

```
batch_insert_batch_size: 20
batch_insert_flush_interval_ms: 20
```

- كتابة CDR أسرع للامتثال
- المزيد من عمليات الكتابة إلى MySQL

تحسين الاستعلامات

استخدام مؤشرات Mnesia بفعالية

الاستعلامات التي تستخدم الحقول المؤشرة هي الأسرع:

```
# استعلامات سريعة (استخدم المؤشرات)
MessageStore.list(status: :pending)
MessageStore.list(dest_smsc: "gateway-1")
Messaging.get_messages_for_smsc("gateway-1")

# استعلامات أبطأ (مسح كامل للجدول)
MessageStore.list(limit: :infinity)  # Returns all messages
```

مجموعة اتصالات MySQL

قم بتكوين مجموعة اتصالات MySQL، للاستعلامات والأرشفة CDR:

```
# config/runtime.exs
config :sms_c, SmsC.Repo,
  pool_size: 10  # Increase for heavy CDR reporting
```

إرشادات:

- النشر القياسي: `pool_size: 10`
- تقارير CDR الثقيلة: `pool_size: 20-30`
- أرشفة فقط: `pool_size: 5`

اختبار الأداء

تشغيل الاختبارات

يتضمن المشروع اختبارات قائمة على Benchee لاختبار الأداء:

```
# اختبار واجهة برمجة التطبيقات SMS وغير التزامن بين يقارن) الخام
(المتزامن)
mix run benchmarks/raw_sms_bench.exs

# اختبار واجهة برمجة التطبيقات العامة للرسائل
mix run benchmarks/message_api_bench.exs
```

تفسير النتائج

نموذج الناتج:

```
Name                                          ips        average
deviation         median         99th %
submit_message_raw_async (batch)        4.65 K        0.22 ms
±41.72%       0.184 ms        0.55 ms
submit_message_raw (sync)             0.0696 K       14.36 ms
±33.42%       12.57 ms       33.71 ms
```

المقاييس الرئيسية:

- **ips**: التكرارات في الثانية (كلما كان أعلى كان أفضل)
- **average**: متوسط زمن التنفيذ (كلما كان أقل كان أفضل)
- **median**: القيمة المتوسطة، أكثر تمثيلاً من المتوسط للتوزيعات المنحرفة
- **99th %**: 99 زمن الاستجابة في النسبة المئوية (مهم للامتثال لـ SLA)

خط الأساس للأداء

الأداء المتوقع على الأجهزة الحديثة (Intel i7-8650U، 8 نوى):

المقياس	insert_message (Mnesia)	السابق (MySQL)
الإنتاجية (مع التوجيه)	1,750 رساله/ثانية	83 رساله/ثانية
الإنتاجية (بسيط)	1,750 رساله/ثانية	89 رساله/ثانية
زمن الاستجابة (متوسط)	0.58 مللي ثانية	16 مللي ثانية
زمن الاستجابة (p99)	<5 مللي ثانية	30 مللي ثانية
الذاكرة لكل عملية	62 كيلوبايت	121 كيلوبايت
تحسين الأداء	**أسرع 21 مرة**	-

التحسينات الرئيسية:

- ✅ إزالة مكالمات ترجمة الأرقام المكررة
- ✅ معالجة غير متزامنة (التوجيه، الشحن، الأحداث)
- ✅ تخزين Mnesia في الذاكرة مقابل MySQL القرص I/O
- ✅ تقليل الذاكرة بنسبة 50%

المراقبة

إحصائيات وقت التشغيل

تحقق من إحصائيات عامل الدفعة:

```
SmsC.Messaging.BatchInsertWorker.stats()
```

ترجع:

```
%{
  total_enqueued: 10000,
  total_flushed: 9900,
  total_batches: 99,
  current_queue_size: 100,
  flush_errors: 0,
  last_flush_at: ~U[2025-10-22 12:34:56Z],
  last_flush_count: 100,
  last_flush_duration_ms: 45
}
```

المقاييس الرئيسية للمراقبة

1. **حجم الطابور**: `current_queue_size` يجب أن يكون أقل من `batch_size` معظم الوقت
2. **مدة التفريغ**: `last_flush_duration_ms` يجب أن تكون < 100مللي ثانية لـ batch_size=100
3. **أخطاء التفريغ**: `flush_errors` يجب أن تكون 0 أو منخفضة جدًا
4. **الإنتاجية**: `total_flushed / uptime` يجب أن تتطابق مع الحمل المتوقع

التنبيهات

قم بإعداد تنبيهات المراقبة لـ:

- حجم الطا��ور بشكل مستمر عند الحد الأقصى (يشير إلى ضغط خلفي)
- زيادة مدة التفريغ (تدهور أداء قاعدة البيانات)
- أخطاء التفريغ > 0 (مشاكل في الاتصال بقاعدة البيانات)
- الإنتاجية أقل من المتوقع (تدهور الأداء)

استكشاف الأخطاء وإصلاحها

العرض: إنتاجية منخفضة

الأسباب المحتملة:

1. نفاد مجموعة اتصالات قاعدة البيانات: زيادة `pool_size`

2. قاعدة بيانات بطيئة: تحقق من أداء الاستعلام، أضف مؤشرات
3. زمن انتقال الشبكة: تحسين مسار الشبكة إلى قاعدة البيانات
4. حجم الدفعة صغير جدًا: زيادة `batch_insert_batch_size`

العرض: زمن استجابة مرتفع

الأسباب المحتملة:

1. فترة التفريغ مرتفعة جدًا: تقليل `batch_insert_flush_interval_ms`
2. حجم الدفعة مرتفع جدًا: تقليل `batch_insert_batch_size`
3. القرص، تحسين الجداول I/O كتابة قاعدة البيانات بطيئة: تحقق من
4. استخدام واجهة برمجة التطبيقات غير المتزامنة عندما تحتاج إلى متزامنة: التحويل إلى نقطة نهاية متزامنة

العرض: مشاكل في الذاكرة

الأسباب المحتملة:

1. الطابور يتراكم: الرسائل تتجمع أسرع من التفريغ
2. حجم الدفعة كبير جدًا: تقليل `batch_insert_batch_size`
3. في الإحصائيات `flush_errors` فشل التفريغ: تحقق من
4. الحاجة إلى إعادة تشغيل العامل: `Supervisor.terminate_child/2` وإعادة التشغيل

أفضل الممارسات

1. **ابدأ بالإعدادات الافتراضية** (100/100مللي ثانية) وضبطها بناءً على السلوك الملحوظ
2. **راقب في الإنتاج** لمدة أسبوع على الأقل قبل تحسين الأداء
3. **اختبر تغييرات التكوين** في بيئة الاختبار مع حمل مشابه للإنتاج
4. **استخدم الاختبارات** للتحقق من تغييرات التكوين
5. **وثق قرارات الضبط** الخاصة بك للرجوع إليها في المستقبل
6. **قم بإعداد التنبيهات** قبل تحسين الأداء لالتقاط التراجعات
7. **اعتبر المناطق الزمنية** - الحمل الأقصى يختلف حسب المنطقة

تكوينات مثال

التكوين: مجمع عالي الحجم

```
# config/prod.exs
config :sms_c,
  batch_insert_batch_size: 200,
  batch_insert_flush_interval_ms: 200

config :sms_c, SmsC.Repo,
  pool_size: 50
```

التكوين: رسائل حقيقية في الوقت الحقيقي للمؤسسات

```
# config/prod.exs
config :sms_c,
  batch_insert_batch_size: 20,
  batch_insert_flush_interval_ms: 10

config :sms_c, SmsC.Repo,
  pool_size: 20
```

التكوين: تطوير/اختبار

```
# config/dev.exs
config :sms_c,
  batch_insert_batch_size: 10,
  batch_insert_flush_interval_ms: 50

config :sms_c, SmsC.Repo,
  pool_size: 5
```

قراءة إضافية

- دليل أداء Ecto
- وثائق Benchee
- تحت الضغط Phoenix

توجيه الرسائل القصيرة

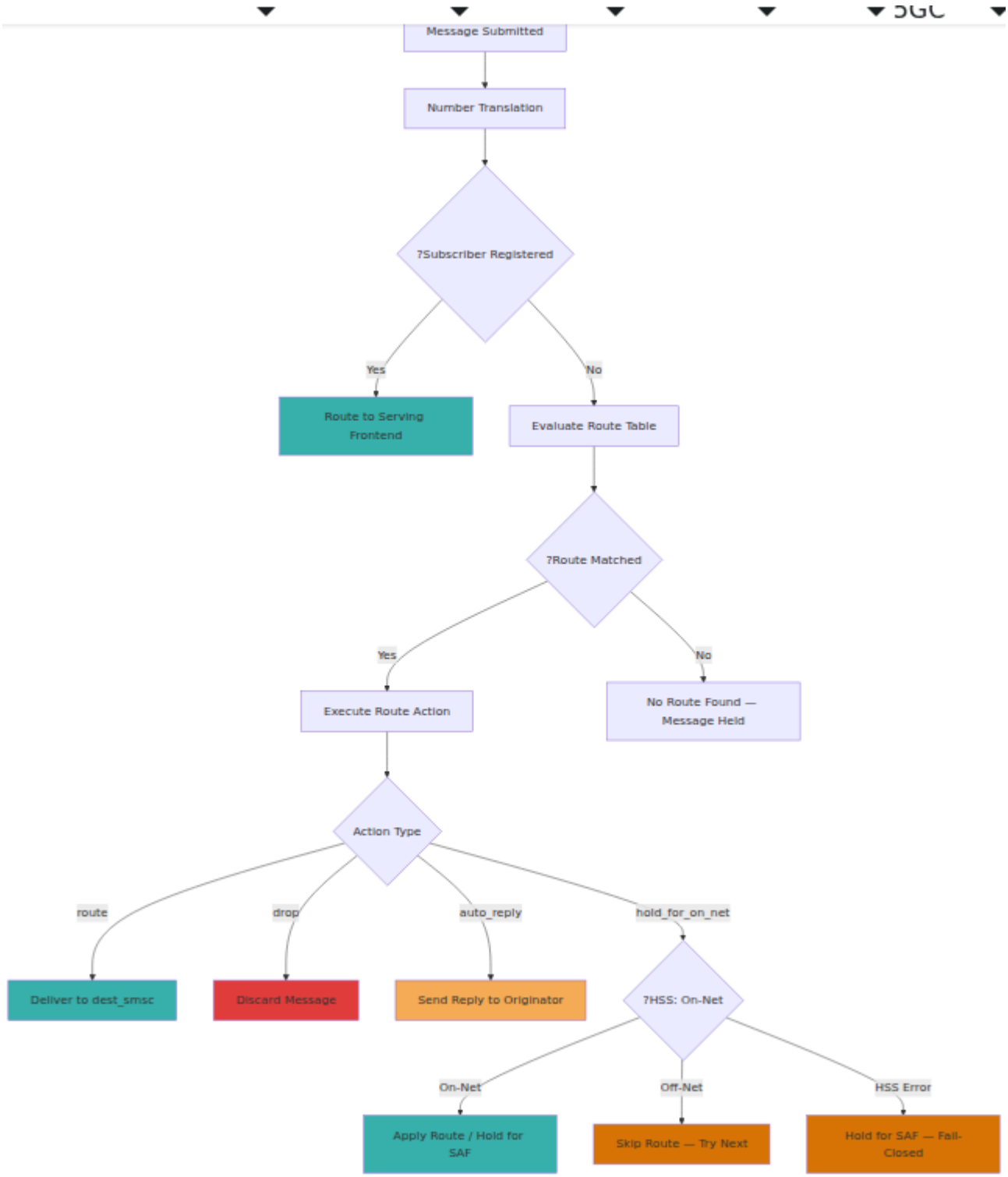
← العودة إلى فهرس الوثائق

نظرة عامة

يحدد محرك توجيه SMS-C وجهة كل رسالة بعد دخولها النظام. يتم تخزين المسارات في Mnesia، ويمكن تعديلها في وقت التشغيل عبر API، أو لوحة التحكم، وتدعم مطابقة التعبيرات العادية، وترتيب الأولويات، وتوازن الحمل الموزون، وقرارات الشبكة الداخلية/الخارجية المعتمدة على HSS.

خط معالجة الرسائل

تتبع كل رسالة هذا الخط من التقديم إلى التوجيه:



ترتيب الأولويات

1. **التوجيه المعتمد على الموقع** — إذا كان لدى المشترك الوجهة تسجيل نشط، يتم التوجيه مباشرة إلى الواجهة الأمامية الخدمية. يتم تخطي جميع قواعد التوجيه.
2. **تقييم جدول التوجيه** — يتم تقييم جميع المسارات المفعلة مقابل الرسالة. يتم تسجيل كل مسار مع تفاصيل النجاح/الفشل لكل معيار مطابقة.

3. **تنفيذ إجراء التوجيه** — يحدد إجراء المسار المختار ما يحدث للرسالة.

نموذج بيانات المسار

الحقل	النوع	الوصف
`route_id`	عدد صحيح	م��رف فريد يتم إنشاؤه تلقائيًا
`calling_regex`	سلسلة	تعبير عادي لرقم المصدر (حرف = `nil` عام)
`called_regex`	سلسلة	تعبير عادي لرقم الوجهة (حرف = `nil` عام)
`source_smsc`	سلسلة	اسم SMSC المصدر (حرف عام = `nil`)
`dest_smsc`	سلسلة	SMSC الوجهة. مطلوب لإجراء `route`، اختياري لـ `hold_for_on_net`
`source_type`	سلسلة	تصفية حسب نوع المصدر: `ims`، `circuit_switched`، `smpp`، أو `nil`
`action`	سلسلة	إجراء التوجيه: `route`، `drop`، `auto_reply`، `hold_for_on_net`
`auto_reply_message`	سلسلة	نص الرد (مطلوب عندما يكون الإجراء `auto_reply`)
`charged`	سلسلة	الشحن: `yes`، `no`، `default`
`weight`	عدد صحيح	وزن توازن الحمل (1–100)
`priority`	عدد صحيح	الأولوية (1–255، أقل = أعلى أولوية)
`description`	سلسلة	تسمية قابلة للقراءة البشرية
`enabled`	منطقي	تبديل نشط/غير نشط

الحقل	النوع	الوصف
`originating_on_net_only`	منطقي	المصدر MSISDN يتم إسقاطه إذا كان خارج الشبكة وفقًا لـ HSS
`terminating_on_net_only`	م��طقي	الوجهة MSISDN يتم إسقاطه إذا كان خارج الشبكة وفقًا لـ HSS

إجراءات التوجيه

`route` (افتراضي)

هذا هو إجراء التوجيه القياسي. يوصل الرسالة إلى `dest_smsc`.

```
{
  "action": "route",
  "called_regex": "^61",
  "dest_smsc": "australia-gw",
  "priority": 100,
  "description": "Route Australian numbers to AU gateway"
}
```

`drop`

يتخلص من الرسالة. لا حاجة لـ `dest_smsc`.

```
{
  "action": "drop",
  "called_regex": "^900",
  "priority": 5,
  "description": "Block premium-rate numbers"
}
```

`auto_reply`

يرسل `auto_reply_message` مرة أخرى إلى المنشئ. يتم وضع علامة على الرسالة الأصلية كرسالة تم الرد عليها تلقائيًا.

```
{
  "action": "auto_reply",
  "called_regex": "^911$",
  "auto_reply_message": "Emergency: SMS to 911 is not supported.
Please dial 911.",
  "priority": 1,
  "description": "911 auto-response"
}
```

`hold_for_on_net`

يتحقق من MSISDN الوجهة مقابل HSS عبر Diameter Sh. يعتمد السلوك على النتيجة:

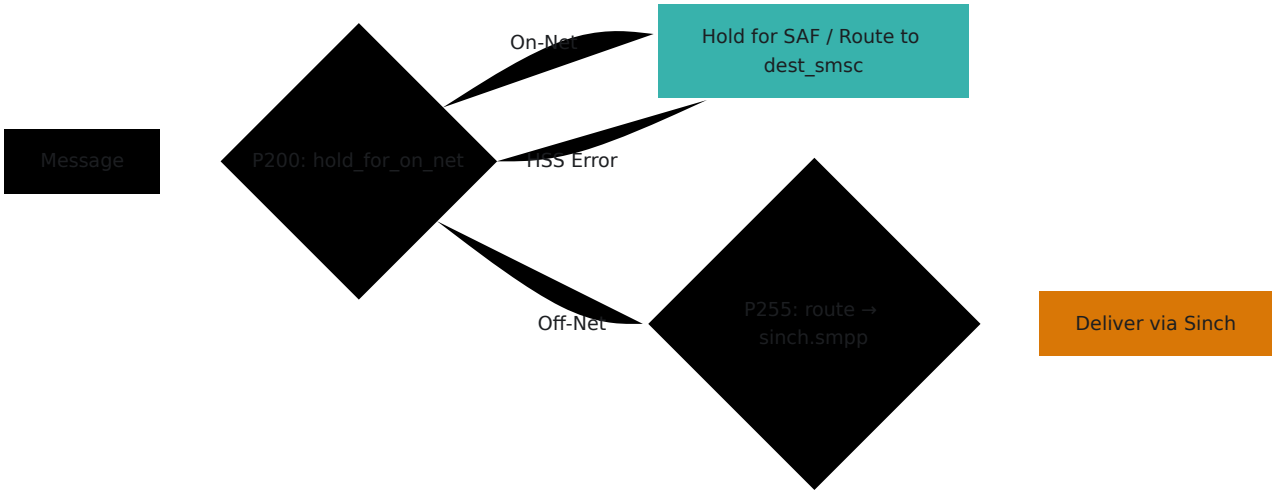
- **داخل الشبكة**: يتم تطبيق التوجيه. إذا تم تعيين `dest_smsc`، إذا يتم التوصيل هناك. إذا كانت `dest_smsc` `null`، يتم الاحتفاظ بالرسالة للتسليم بالتخزين والإرسال (SAF) عند تسجيل المشترك.
- **خارج الشبكة**: لا ينطبق التوجيه. يتخطى محرك التوجيه هذا المسار ويواصل تقييم المسار المطابق التالي.
- **خطأ Diameter أو تعطيل HSS**: يتم الاحتفاظ بالرسالة لـ SAF. فشل مغلق.

```
{
  "action": "hold_for_on_net",
  "priority": 200,
  "description": "Hold on-net subscribers for local delivery"
}
```

مع وجهة صريحة:

```
{
  "action": "hold_for_on_net",
  "dest_smsc": "ims-frontend-01",
  "priority": 200,
  "description": "Route on-net to IMS frontend"
}
```

الإعداد النموذجي: الاحتفاظ داخل الشبكة + الاحتياطي خارج الشبكة

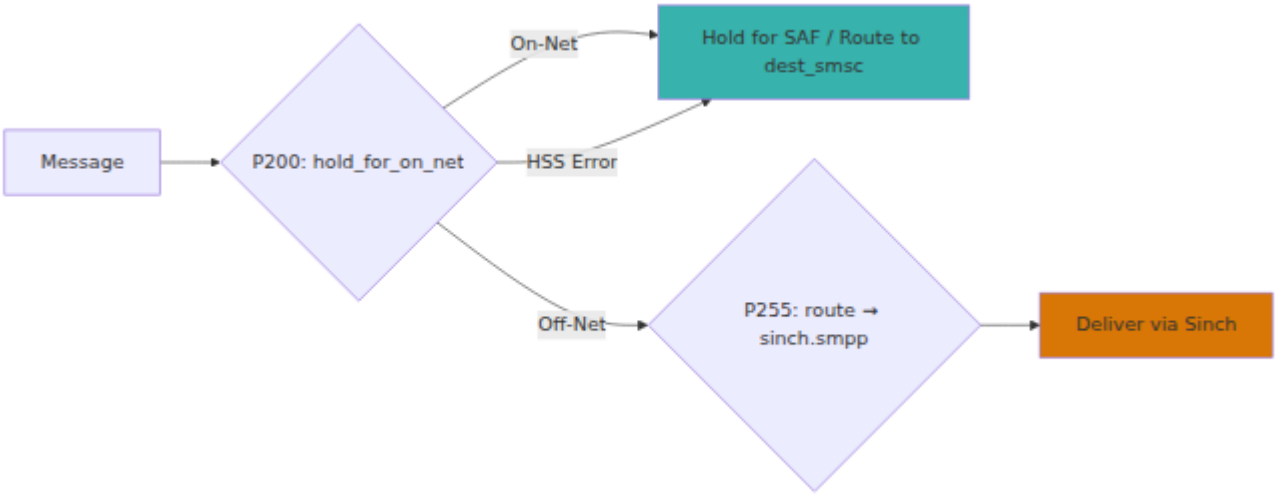


هذا يضمن عدم تسرب المشتركين داخل ال��بكة إلى بوابة خارج الشبكة. تسقط الأرقام خارج الشبكة بشكل نظيف.

ذاكرة التخزين المؤقت لنتائج HSS

تعتبر عمليات البحث في HSS مكلفة (جولة Diameter إلى HSS). يقوم النظام بتخزين النتائج لكل MSISDN لتجنب الغمر الزائد.

- **للذاكرة المؤقتة**: ساعتان **TTL** (قابلة للتكوين عبر `hss_cache_ttl_seconds`)
- **الغمر الكسول**: يتم استعلام HSS فقط عندما يحتاج التوجيه فعليًا إلى حالة داخل الشبكة (`hold_for_on_net`، `originating_on_net_only`، `terminating_on_net_only`)
- **مشاركة عبر المسارات**: إذا كانت هناك مسارات متعددة تحتاج إلى حالة نفس MSISDN، يحدث غمر HSS واحد فقط
- **إعادة التوجيه القسري**: يتجاوز الذاكرة المؤقتة (غمر جديد) عندما يتم إعادة توجيه الرسالة بشكل صريح



علامات تنفيذ داخل الشبكة

هذه هي علامات منطقية على أي مسار تحدد ما إذا كان المسار ينطبق بناءً على حالة HSS. تعمل بشكل مستقل عن إجراء التوجيه.

`originating_on_net_only`

إذا كانت `true`، يتم التحقق من MSISDN المصدر مقابل HSS. إذا كان خارج الشبكة، يتم إسقاط الرسالة. يستخدم ذاكرة التخزين المؤقت لـ HSS.

حالة الاستخدام: تقييد ربط SMPP أو الناقل الخارجي لقبول حركة المرور فقط من المنشئين داخل الشبكة.

`terminating_on_net_only`

إذا كانت `true`، يتم التحقق من MSISDN الوجهة مقابل HSS. إذا كان خارج الشبكة، يتم إسقاط الرسالة. يستخدم ذاكرة التخزين المؤقت لـ HSS.

حالة الاستخدام: تقييد مسار للتسليم فقط إلى المشتركين داخل الشبكة.

سلوك الفشل المغلق

تفشل جميع الفحوصات داخل الشبكة بشكل مغلق:

- HSS → يتم إسقاط الرسالة تعيد خطأ
- Diameter → يتم إسقاط الرسالة معطل

- غير مت��حة → يتم إسقاط الرسالة HSS

تسجيل تقييم المسار

يتم تسجيل كل مسار تم تقييمه لرسالة كحدث `route_evaluated` على تلك الرسالة. يتضمن السجل:

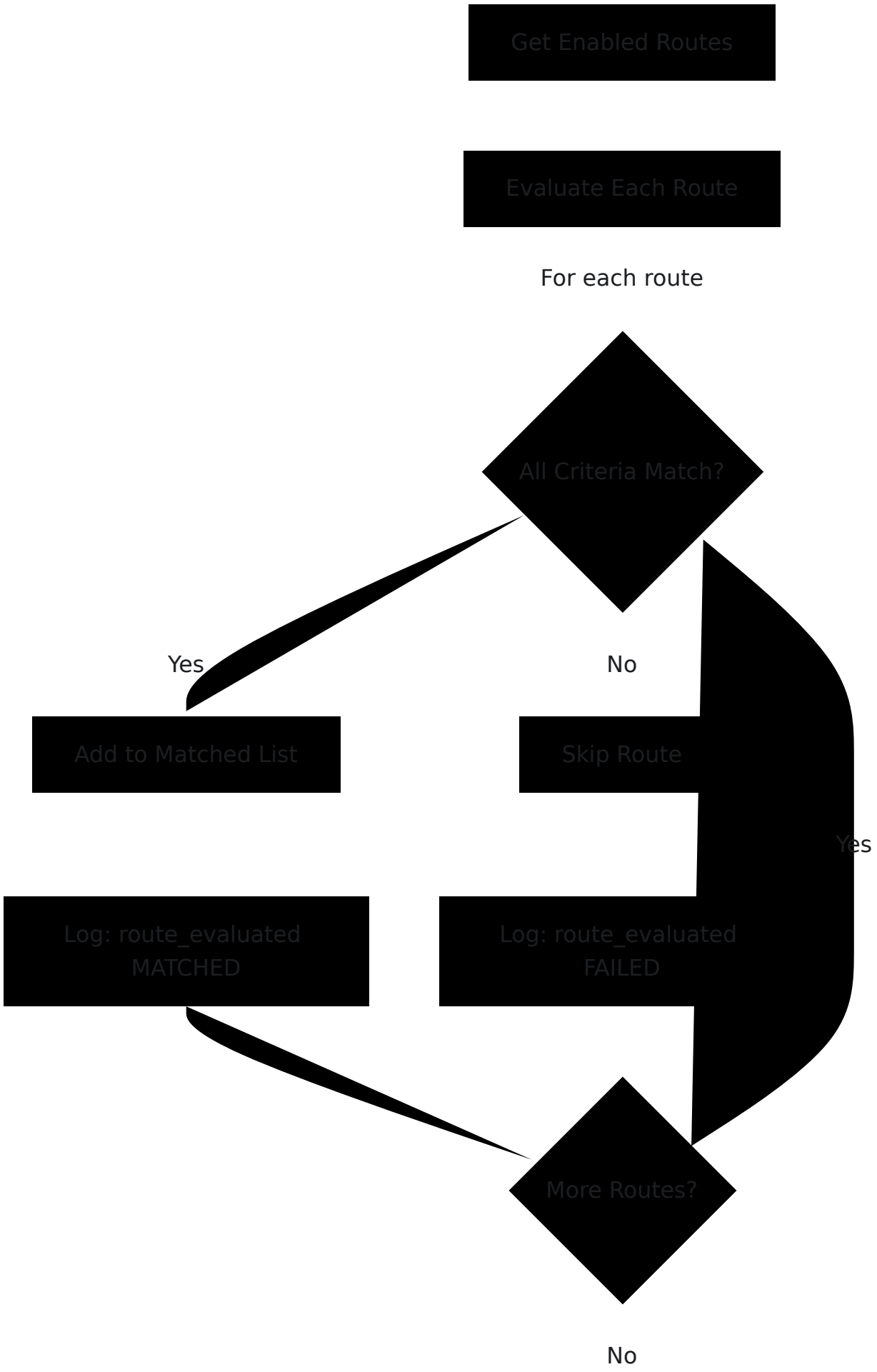
- معرف المسار، الإجراء، والأولوية
- حالة النجاح/الفشل لكل معيار مطابقة
- تفاصيل ما تم مقارنته (اسم، الرقم مقابل نمط التعبير العادي، SMSC مقابل المصدر، إلخ.)

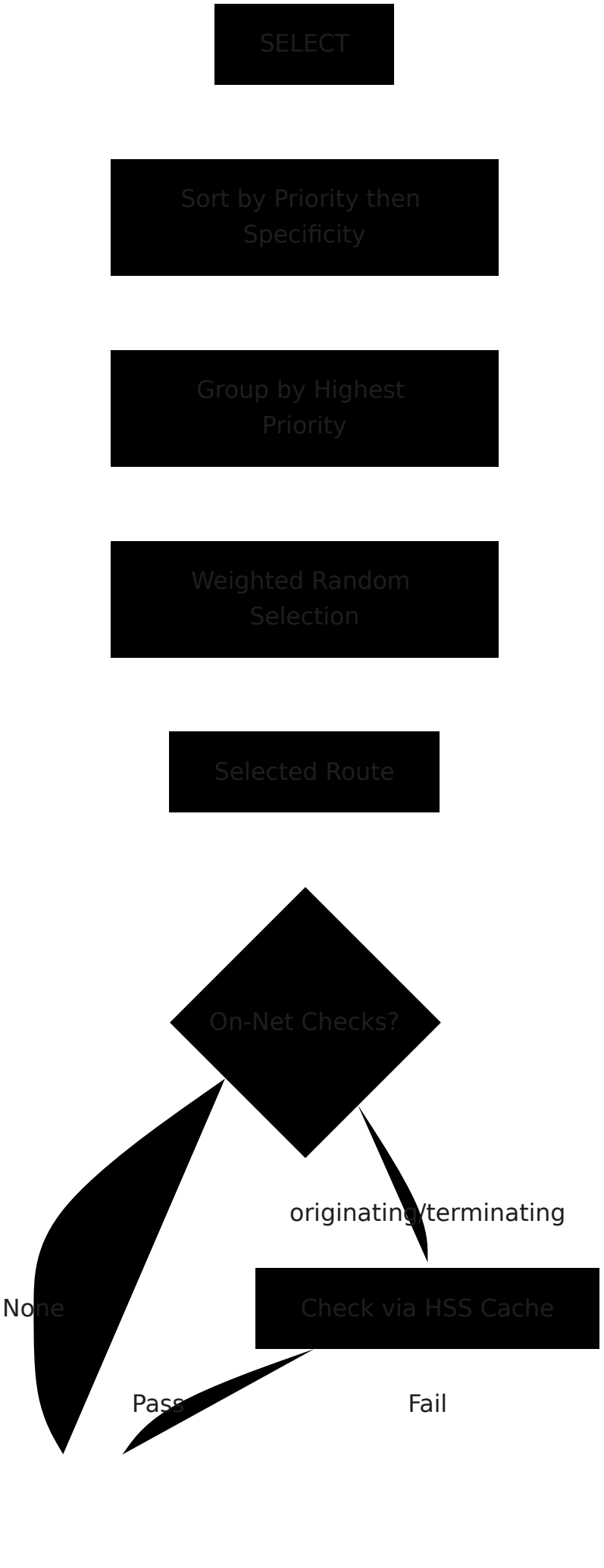
أمثلة على إدخالات سجل الأحداث:

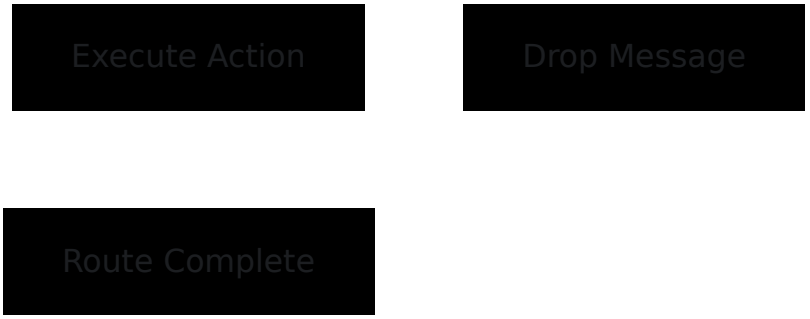
```
Route #4 [hold_for_on_net] MATCHED: P200 "Hold on-net" |
  calling_regex: ✓ (wildcard), called_regex: ✓ (wildcard),
  source_smsc: ✓ (wildcard), source_type: ✓ (wildcard),
  enum_result: ✓ (wildcard)

Route #3 [route] FAILED: P255 "Sinch fallback" → sinch.smpp |
  calling_regex: ✓ (wildcard), called_regex: ✓ (wildcard),
  source_smsc: ✓ (wildcard), source_type: ✗ (ims vs
circuit_switched),
  enum_result: ✓ (wildcard)
```

تكون هذه الأحداث مرئية في سجل نشاط الرسائل في لوحة التحكم وعبر واجهة API للأحداث.

خوارزمية تقييم المسار


Get Enabled Routes
Evaluate Each Route
For each route
All Criteria Match?
Yes
No
Add to Matched List
Skip Route
Yes
Log: route_evaluated MATCHED
Log: route_evaluated FAILED
More Routes?
No


SELECT
Sort by Priority then Specificity
Group by Highest Priority
Weighted Random Selection
Selected Route
On-Net Checks?
originating/terminating
None
Check via HSS Cache
Pass
Fail



تسجيل الخصوصية

عندما تتطابق مسارات متعددة، تحدد الخصوصية أيها مفضل ضمن نفس مستوى الأولوية:

المعيار	النقاط
طول نمط `called_regex`	الطول × 100
طول نمط `calling_regex`	الطول × 50
تحديد `source_smsc`	+25
تحديد `enum_result_domain`	+15
تحديد `source_type`	+10
تحديد `enum_domain`	+5

تتفوق الأنماط الأطول والأكثر خصوصية دائمًا على الحروف العامة.

التكوين

تحميل المسارات من التكوين

يتم تحميل المسارات المعرفة في `config/runtime.exs` تحت `:sms_routes` عند بدء التشغيل الأول عندما يكون جدول التوجيه فارغًا. تحافظ عمليات إعادة التشغيل اللاحقة على المسارات المضافة عبر API.

```
config :sms_c, :sms_routes, [
  %{
    called_regex: ~r/^68987/,
    dest_smsc: "pacific-gw",
    priority: 100,
    description: "Pacific Islands routing"
  },
  %{
    action: :hold_for_on_net,
    priority: 200,
    description: "Hold on-net for SAF"
  },
  %{
    action: :auto_reply,
    called_regex: ~r/^911$/,
    auto_reply_message: "Emergency: Please dial 911.",
    priority: 1,
    description: "911 auto-response"
  }
]
```

معلمات تكوين المسار

المعلمة	النوع	مطلوب	الافتراضي	ف
`action`	Atom/String	لا	`:route`	جراء التوجيه `:drop`، `:auto_rep` `:hold_for`
`calling_regex`	Regex	لا	`nil`	العادي لرقم المصدر. `ni` الجميع
`called_regex`	Regex	لا	`nil`	العادي لرقم الوجهة ق مع الجميع
`source_smsc`	سلسلة	لا	`nil`	حسب اسم صدر SMSC ق مع الجميع
`dest_smsc`	سلسلة	مشروط	`nil`	طلوب لإجراء اختياري لـ `hold_for_` خدم من قبل `drop`/`aut`
`source_type`	Atom	لا	`nil`	ة حسب نوع المصدر: `:i` `:circuit_` `:smpp`
`auto_reply_message`	سلسلة	مشروط	`nil`	عندما يكون الإجراء `:au`
`charged`	Atom	لا	`:default`	ياسة الشحن `:no`، `:def`

المعلمة	النوع	مطلوب	الافتراضي	صف
`weight`	عدد صحيح	لا	`100`	ل الحمل (1–100)
`priority`	عدد صحيح	لا	`100`	يم (1–255، قل = أعلى)
`description`	سلسلة	لا	`""`	قابل للقراءة البشرية
`enabled`	منطقي	لا	`true`	مسار نشطًا
`originating_on_net_only`	منطقي	لا	`false`	لب أن يكون ل MSISDN شبكة وفقًا لـ
`terminating_on_net_only`	منطقي	لا	`false`	لب أن يكون هة MSISDN شبكة وفقًا لـ HSS

HSS تكوين ذاكرة التخزين المؤقت لـ

المعلمة	النوع	الافتراضي	الوصف
`hss_cache_ttl_seconds`	عدد صحيح	`7200`	مدة الحياة لنتائج البحث المخزنة في ذاكرة التخزين المؤقت لـ HSS (بالثواني)

API

إنشاء مسار

```
POST /api/routes
```

```
{
  "action": "hold_for_on_net",
  "priority": 200,
  "description": "Hold on-net subscribers for local delivery"
}
```

قائمة المسارات

```
GET /api/routes
```

تحديث المسار

```
PUT /api/routes/:id
```

حذف المسار

```
DELETE /api/routes/:id
```

استيراد/تصدير

```
POST /api/routes/import
GET /api/routes/export
```

يدعم الاستيراد أوضاع `merge` (إضافة إلى الموجودة) و `replace` (مسح واستبدال الكل).

استكشاف الأخطاء وإصلاحها

لم يتم العثور على مسار

الأعراض: تبقى الرسالة معلقة مع `sms_routing_failed`، يظهر سجل الأحداث `dest_smsc: null`

الأسباب ��لمحتملة:

- لا توجد مسارات مفعلة تتطابق مع معايير الرسالة
- لا تتطابق أنماط التعبيرات العادية للمسار مع تنسيق الرقم (مثل عدم وجود رمز الدولة)
- جميع المسارات المطابقة تحتوي على `hold_for_on_net`، والوجهة خارج الشبكة ولكن لا يوجد مسار احتياطي

الحل:

1. تحقق من سجل الأحداث الخاص بالرسالة — يتم تسجيل كل مسار تم تقييمه مع النجاح/الفشل لكل معيار
2. استخدم محاكي التوجيه في لوحة التحكم للاختبار بنفس الأرقام
3. تأكد من وجود مسار احتياطي شامل عند رقم أولوية مرتفع (مثل 255)

تم توجيه المشترك داخل الشبكة إلى خارج الشبكة

الأعراض: يتم تسليم الرسائل إلى المشتركين داخل الشبكة عبر بوابة خارجية بدلاً من الاحتفاظ بها للتسليم المحلي

الأسباب المحتملة:

- لا يوجد مسار `hold_for_on_net` مُكون
- يحتوي مسار `hold_for_on_net` على رقم أولوية أعلى من مسار الاحتياطي خارج الشبكة
- فشل البحث في HSS (تحقق من أحداث `hss_dip_error`)
- Diameter غير مفعل في التكوين

الحل:

1. أضف مسار `hold_for_on_net` برقم أولوية أقل من الاحتياطي خارج الشبكة

2. تحقق من تمكين Diameter وأن أقران HSS متصلون عبر لوحة التحكم
3. تحقق من حالة ذاكرة التخزين المؤقت لـ HSS ونتائج الغمر الأخيرة في سجل الأحداث

قديمة HSS ذاكرة التخزين المؤقت لـ

الأعراض: تم نقل المشترك أو توفيره ولكن لا يزال يتم التوجيه بناءً على حالة قديمة داخل الشبكة/خارج الشبكة

الحل:

1. أعد توجيه الرسالة عبر API — تتجاوز إعادة التوجيه القسري ذاكرة التخزين المؤقت لـ HSS
2. انتظر انتهاء صلاحية TTL (الافتراضي ساعتان) للذاكرة المؤقتة
3. إذا كانت التحديثات الأسرع مطلوبة قلل من `hss_cache_ttl_seconds`

دليل استكشاف أخطاء SMS-C

← الملف | العودة إلى فهرس الوثائق README الرئيسي

دليل شامل لتشخيص وحل مشكلات SMS-C الشائعة.

جدول المحتويات

- أدوات التشخيص
- مشكلات تسليم الرسائل
- مشكلات التوجيه
- مشكلات الأداء
- مشكلات قاعدة البيانات
- مشكلات الاتصال بالواجهة الأمامية
- مشكلات الشحن/الفوترة
- مشكلات البحث عن ENUM
- مشكلات العنقود
- مشكلات API
- مشكلات واجهة الويب
- مشكلات موارد النظام

أدوات التشخيص

فحص الصحة السريع

```
# 1. تحقق من حالة API
curl https://api.example.com:8443/api/status

# 2. تحقق من نقطة نهاية مقاييس Prometheus
curl https://api.example.com:9568/metrics | grep sms_c

# 3. تحقق من سجلات التطبيق
tail -f /var/log/sms_c/application.log

# 4. تحقق من حالة العملية
systemctl status sms_c

# 5. تحقق من اتصال قاعدة بيانات SQL CDR (MySQL/MariaDB)
mysql -u sms_user -p -h db.example.com -e "SELECT 1"

# لـ PostgreSQL:
# psql -U sms_user -h db.example.com -d sms_c_prod -c "SELECT 1"
```

تحليل السجلات

عرض الأخطاء الأخيرة:

```
# آخر 100 إدخال سجل بمستوى خطأ
tail -1000 /var/log/sms_c/application.log | grep "\[error\]"

# البحث عن أنماط أخطاء محددة
grep "routing_failed" /var/log/sms_c/application.log

# العثور على أخطاء قاعدة البيانات SQL
grep -i "database\|sql\|ecto" /var/log/sms_c/application.log |
grep error
```

مراقبة السجلات في الوقت الحقيقي:

```
# متابعة السجلات مع فلتر
tail -f /var/log/sms_c/application.log | grep -E "
(error|warning|critical)"
```

استعلامات المقاييس

تحقق من معدل معالجة الرسائل:

```
# الرسائل في الثانية
rate(sms_c_message_received_count[5m])

# معدل نجاح التسليم
rate(sms_c_delivery_succeeded_count[5m]) /
rate(sms_c_delivery_queued_count[5m])
```

تحقق من حالة الطابور:

```
# عمق الطابور الحالي
sms_c_queue_size_pending

# عمر الرسالة الأقدم (بالثواني)
sms_c_queue_oldest_message_age_seconds
```

تحقق من أداء النظام:

```
# زمن تأخير معالجة الرسائل (p95)
histogram_quantile(0.95,
sms_c_message_processing_stop_duration_bucket)

# زمن تأخير التوجيه (p95)
histogram_quantile(0.95, sms_c_routing_stop_duration_bucket)
```

مشكلات تسليم الرسائل

الرسائل غير المسلمة

الأعراض:

- الرسائل عالقة في حالة "معلقة"
- عدد الرسائل المعلقة مرتفع
- عدم وجود إشعارات تسليم

خطوات التشخيص:

1. **تحقق من اتصالات الواجهة الأمامية**:

```
curl https://api.example.com:8443/api/frontends/active
```

المتوقع: قائمة بالواجهات الأمامية النشطة المشكلة: قائمة فارغة أو واجهات أمامية مفقودة

2. **تحقق من طابور الرسائل**:

الوصول إلى واجهة الويب: `/message_queue`

- تصفية حسب الحالة: "معلقة"
- تحقق من قيمة `dest_smsc`
- تحقق من أن `deliver_after` ليست في المستقبل

3. **تحقق من التوجيه**:

الوصول إلى واجهة الويب: `/simulator`

- اختبار مع معلمات الرسالة الفعلية
- تحقق من تطابق المسار وأن الوجهة صحيحة

4. **تحقق من استعلام الواجهة الأمامية**:

مراجعة سجلات نظام الواجهة الأمامية:

- هل الواجهة الأمامية تستعلم `/api/messages`؟

- هل الواجهة الأمامية ترسل رأس `smsc` بشكل صحيح؟

الحلول:

لا توجد واجهات أمامية متصلة:

```
# تحقق من حالة نظام الواجهة الأمامية
systemctl status frontend_service

# تحقق من أن الواجهة الأمامية يمكنها الوصول إلى API
curl -k https://api.example.com:8443/api/status

# تسجيل الواجهة الأمامية يدويًا
curl -X POST https://api.example.com:8443/api/frontends/register \
  -H "Content-Type: application/json" \
  -d '{
    "frontend_name": "test_gateway",
    "frontend_type": "smpp",
    "ip_address": "10.0.1.50"
  }'
```

خاطئ SMSC الرسائل موجهة إلى:

- مراجعة تكوين ا��توجيه
- تحقق من أولويات المسار
- اختبار في محاكي التوجيه
- تحقق من أن اسم الواجهة الأمامية يتطابق مع `dest_smsc` في الرسائل

الرسائل المجدولة للمستقبل:

- تحقق من الطابع الزمني لـ `deliver_after`
- إعادة تعيين إذا لزم الأمر:

```
curl -X PATCH https://api.example.com:8443/api/messages/12345 \
  -H "Content-Type: application/json" \
  -d '{"deliver_after": "2025-10-30T12:00:00Z"}'
```

الرسائل الفاشلة مع المحاولات المتكررة

الأعراض:

- زيادة عداد `delivery_attempts`
- رسائل مع عدد محاولات مرتفع (> 3)
- تأخيرات تراجع أسي

خطوات التشخيص:

1. **تحقق من سجل الأحداث**:

```
curl https://api.example.com:8443/api/events/12345
```

ابحث عن:

- أحداث فشل التسليم
- أوصاف الأخطاء
- طوابع زمنية للمحاولات المتكررة

2. **تحقق من سجلات الواجهة الأمامية**:

- لماذا تفشل الواجهة الأمامية في التسليم؟
- أخطاء الشبكة؟
- أخطاء البروتوكول؟
- النظام السفلي غير متاح؟

حلول��I:

مشكلات الشبكة المؤقتة:

- الانتظار لإعادة المحاولة (تلقائي)
- مراقبة التسليم الناجح

الإخفاقات المستمرة:

```
# توجيه إلى بوابة بديلة
curl -X PATCH https://api.example.com:8443/api/messages/12345 \
  -H "Content-Type: application/json" \
  -d '{"dest_smsc": "backup_gateway"}'

# إعادة تعيين عداد المحاولات
curl -X PATCH https://api.example.com:8443/api/messages/12345 \
  -H "Content-Type: application/json" \
  -d '{"delivery_attempts": 0, "deliver_after": "2025-10-
30T12:00:00Z"}'
```

رقم الوجهة غير صالح:

- تحقق من تنسيق الرقم
- تحقق من قواعد ترجمة الأرقام
- احذف الرسالة إذا كانت غير صالحة حقًا

الرسائل الميتة

الأعراض:

- `deadletter: true` في الرسالة
- الرسائل بعد وقت انتهاء الصلاحية
- الحالة لا تزال "معلقة"

خطوات التشخيص:

1. **العثور على الرسائل الميتة**:

الوصول إلى واجهة الويب: `/message_queue`

- تصفية حسب الحالة المنتهية
- تحقق من طوابع انتهاء الصلاحية

2. **تحقق من سبب انتهاء الصلاحية**:

- مراجعة سجل الأحداث
- تحقق من تاريخ محاولات التسليم
- تحقق من أن التوجيه كان ناجحًا

الحلول:

تمديد انتهاء الصلاحية:

```
# إضافة 24 ساعة إلى انتهاء الصلاحية
curl -X PATCH https://api.example.com:8443/api/messages/12345 \
  -H "Content-Type: application/json" \
  -d '{"expires": "2025-10-31T12:00:00Z", "deadletter": false}'
```

مشكلات التوجيه

لا يوجد مسار موجود

الأعراض:

- خطأ: `no_route_found`
- زيادة مقياس `sms_c_routing_failed_count`
- سجل الأحداث يظهر "routing_failed"

خطوات التشخيص:

1. **تحقق من وجود المسارات**:

الوصول إلى واجهة الويب: `/sms_routing`

- تحقق من تكوين المسارات
- تحقق من أن هناك مسار واحد على الأقل مفعل

2. **اختبار التوجيه**:

الوصول إلى واجهة الويب: `/simulator`

- إدخال معلمات الرسالة (المصدر SMSC، رقم المتصل، الرقم المدعو)
- مراجعة نتائج التقييم
- تحقق من سبب عدم تطابق المسارات

3. **تحقق من معايير ��لمسار**:

- هل تتطلب المطابقات البادئة؟
- هل فلتر SMSC المصدر صارم جدًا؟
- هل جميع المسارات معطلة؟

الحلول:

لا توجد مسارات مكونة:

إضافة مسار شامل:

```
بادئة الاتصال: (فارغ)
بادئة المدعو: (فارغ)
المصدر SMSC: (فارغ)
الوجهة SMSC: default_gateway
الأولوية: 255
الوزن: 100
مفعل: ✓
الوصف: مسار افتراضي شامل
```

المسارات محددة جدًا:

إضافة مسار أوسع:

```
بادئة المدعو: +
الوجهة SMSC: international_gateway
الأولوية: 200
الوزن: 100
مفعل: ✓
الوصف: شامل دولي
```

جميع المسارات معطلة:

- تفعيل المسارات المناسبة عبر واجهة الويب
- تحقق من أن التكوين لم يعطل المسارات عن طريق الخطأ

المسار الخاطئ المحدد

الأعراض:

- توجيه الرسائل إلى وجهة غير متوقعة
- بوابة خاطئة تتلقى حركة المرور
- عدم توزيع الحمل كما هو متوقع

خطوات التشخيص:

1. **استخدام محاكي التوجيه**:

الوصول إلى واجهة الويب: `/simulator`

- اختبار مع معلمات الرسالة الفعلية
- مراجعة قسم "جميع المطابقات"
- تحقق من أولويات ودرجات التخصص

2. **تحقق من أولويات المسار**:

- الرقم الأقل = أولوية أعلى
- يتم تقييم المسارات بترتيب الأولوية
- ضمن نفس الأولوية، يتم تطبيق الأوزان

3. **تحقق من تخصص المسار**:

تقييم التخصص:

- بادئة المدعو الأطول: +100 نقطة لكل حرف
- بادئة المتصل الأطول: +50 نقطة لكل حرف
- المصدر المحدد: +25 نقطة SMSC
- نوع المصدر المحدد: +10 نقاط
- مجال ENUM المحدد: +15 نقطة

الحلول:

تعديل الأولويات:

اجعل المسار المحدد أعلى أولوية:

```
المسار المميز:
  بادئة المدعو: +1555
  الأولوية: 10 (أولوية عالية)

المسار العام:
  بادئة المدعو: +1
  الأولوية: 50 (أولوية أقل)
```

تعديل الأوزان:

تغيير توزيع تحميل التوازن:

```
الأول (70%):
  الوزن: 70

النسخة الاحتياطية (30%):
  الوزن: 30
```

إضافة مسار أكثر تخصصًا:

تجاوز المسار العام لحالة محددة:

```
المسار المحدد:
  بادئة المدعو: +15551234
  الوجهة SMSC: dedicated_gateway
  الأولوية: 1

المسار العام:
  بادئة المدعو: +1
  الوجهة SMSC: general_gateway
  الأولوية: 50
```

الرد التلقائي لا يعمل

الأعراض:

- تم تكوين مسار الرد التلقائي ولكن لا يتم تفعيله
- عدم إرسال رسائل الرد

- سجل الأحداث يفتقر إلى حدث الرد التلقائي

خطوات التشخيص:

1. **تحقق من تكوين المسار**:

- `auto_reply: true`
- `auto_reply_message` يحتوي على نص
- المسار مفعل
- يتطابق المسار مع معايير الرسالة

2. **اختبار في المحاكي**:

- تحقق من اختيار المسار
- تحقق من وجود إشارة "auto_reply"

3. **تحقق من سجل الأحداث**:

```
curl https://api.example.com:8443/api/events/12345 | grep
auto_reply
```

الحلول:

المسار لا يتطابق:

- توسيع المعايير (إزالة الفلاتر)
- تحقق من الأولوية (يجب أن تكون أعلى من المسارات العادية)
- تحقق من حالة التفعيل

الرسالة غير محددة:

تحرير المسار، إضافة الرسالة:

```
الرد التلقائي: ✓
"شكرًا لك على رسالتك. سنرد قريبًا" :رسالة الرد التلقائي.
```

الأولوية خاطئة:

يجب أن تحتوي مسارات الرد التلقائي على أولوية عالية (رقم منخفض):

```
مسار الرد التلقائي:
  الأولوية: 10

المسار العادي:
  الأولوية: 50
```

مشكلات الأداء

زمن معالجة الرسائل مرتفع

الأعراض:

- `sms_c_message_processing_stop_duration` p95 > 1000ms
- استجابات API بطيئة
- تراكم الطابور

خطوات التشخيص:

1. **تحقق من زمن تأخير المكونات**:

```
# زمن تأخير التوجيه
histogram_quantile(0.95, sms_c_routing_stop_duration_bucket)

# زمن تأخير البحث عن ENUM
histogram_quantile(0.95, sms_c_enum_lookup_stop_duration_bucket)

# زمن تأخير الشحن
histogram_quantile(0.95, sms_c_charging_succeeded_duration_bucket)

# زمن تأخير التسليم
histogram_quantile(0.95, sms_c_delivery_succeeded_duration_bucket)
```

2. **تحقق من موارد النظام**:

```
# استخدام وحدة الم��الجة المركزية
top -b -n 1 | grep sms_c

# استخدام الذاكرة
ps aux | grep beam.smp
```

الحلول:

التوجيه بطيء (الكثير من المسارات):

- تقليل عدد المسارات المفعلة
- دمج المسارات المشابهة
- تحسين معايير المسار

البحث عن ENUM بطيء:

- تحقق من زمن استجابة خادم DNS
- زيادة مهلة الاتصال
- استخدام خوادم DNS أسرع/أقرب
- تعطيل ENUM إذا لم يكن مطلوبًا

الشحن بطيء:

- تحقق من أداء OCS
- زيادة مهلة OCS
- تعطيل الشحن إذا لم يكن مطلوبًا
- استخدام الشحن غير المتزامن

قاعدة البيانات بطيئة:

- زيادة حجم مجموعة الاتصالات
- إضافة فهارس
- تحسين الاستعلامات
- ترقية موارد قاعدة البيانات

تغييرات التكوين:

```
# config/config.exs
# زيادة حجم الدفعة لزيادة الإنتاجية
config :sms_c,
  batch_insert_batch_size: 200,
  batch_insert_flush_interval_ms: 200

# زيادة مجموعة قاعدة البيانات
config :sms_c, SmsC.Repo,
  pool_size: 50
```

إنتاجية الرسائل منخفضة

الأعراض:

- معالجة < 100 رسالة/ثانية
- غير المتزامن ولكن لا يزال بطيئًا API استخدام
- مرتفعة API أوقات استجابة

خطوات التشخيص:

1. **تحقق من عامل الدفعة**:

```
# في وحدة التحكم الإنتاجية (iex)
SmsC.Messaging.BatchInsertWorker.stats()
```

ابحث عن:

- `current_queue_size` قريب من الحد الأقصى
- `flush_errors` > 0
- `last_flush_duration_ms` مرتفع جدًا

2. **تحقق من الاختناقات**:

```
# زمن استعلام قاعدة البيانات
ecto_pools_query_time

# زمن طابور مجموعة الاتصال
ecto_pools_queue_time
```

الحلول:

اختناق قاعدة البيانات:

زيادة حجم المجموعة:

```
config :sms_c, SmsC.Repo,
  pool_size: 50  # زيادة من 20
```

تكوين الدفعة:

ضبط الإنتاجية:

```
config :sms_c,
  batch_insert_batch_size: 200,  # دفعات أكبر
  batch_insert_flush_interval_ms: 200  # فترة أطول
```

استخدام نقطة النهاية غير المتزامنة:

```
# استخدم :إنتاجية عالية /create_async
curl -X POST
https://api.example.com:8443/api/messages/create_async

# ليس: /api/messages (متزامن)
```

تراكم الطابور المتزايد

الأعراض:

- `sms_c_queue_size_pending` في زيادة
- عمر الرسالة الأقدم في زيادة

- المعالجة لا تستطيع مواكبة معدل الوارد

خطوات التشخيص:

1. **تحقق من معدل الوارد مقابل معدل التسليم**:

```
# معدل الوارد
rate(sms_c_message_received_count[5m])

# معدل التسليم
rate(sms_c_delivery_succeeded_count[5m])
```

2. **تحقق من سعة الواجهة الأمامية**:

- هل الواجهات الأمامية تستعلم بشكل متكرر بما فيه الكفاية؟
- هل الواجهات الأمامية تعالج الرسائل بسرعة كافية؟
- هل هناك أي أخطاء في الواجهة الأمامية؟

3. **تحقق من معدل نجاح التسليم**:

```
rate(sms_c_delivery_succeeded_count[5m]) /
rate(sms_c_delivery_attempted_count[5m])
```

الحلول:

الواجهات الأمامية لا تستعلم:

- تحقق من اتصال الواجهة الأمامية
- تحقق من فترة الاستعلام (يجب أن تكون 5-10 ثواني)
- إعادة تشغيل خدمات الواجهة الأمامية

الواجهات الأمامية بطيئة جدًا:

- إضافة المزيد من مثيلات الواجهة الأمامية
- تحسين معالجة الواجهة الأمامية
- زيادة التزامن في الواجهة الأمامية

معدل إعادة المحاولة مرتفع:

- التحقيق في فشل التسليم
- إصلاح المشكلات السفلية
- توجيه إلى بوابات بديلة

ذروة مؤقتة:

- الانتظار لتفريغ الطابور
- المراقبة حتى تعود الأمور إلى طبيعتها
- النظر في ترقية السعة إذا كانت متكررة

مشكلات قاعدة البيانات

فشل الاتصال

الأعراض:

- خطأ: "غير قادر على الاتصال بقاعدة البيانات"
- API 500 تعيد أخطاء
- التطبيق لا يبدأ

خطوات التشخيص:

1. **تحقق من حالة قاعدة بيانات SQL CDR**:

```
# MySQL/MariaDB
systemctl status mysql

# PostgreSQL
systemctl status postgresql

# اختبار الاتصال (MySQL/MariaDB)
mysql -u sms_user -p -h db.example.com -e "SELECT 1"

# اختبار الاتصال (PostgreSQL)
psql -U sms_user -h db.example.com -d sms_c_prod -c "SELECT 1"
```

2. **تحقق من الشبكة**:

```
# اختبار اس??ضافة قاعدة البيانات
ping db.example.com

# تحقق من اتصال المنفذ (MySQL/MariaDB: 3306، PostgreSQL: 5432)
telnet db.example.com 3306
# أو
telnet db.example.com 5432
```

3. **تحقق من بيانات الاعتماد**:

```
# تحقق من متغيرات البيئة
echo $DB_USERNAME
echo $DB_HOSTNAME
echo $DB_PORT

# حاول الاتصال يدويًا بنفس بيانات الاعتماد (MySQL/MariaDB)
mysql -u $DB_USERNAME -p$DB_PASSWORD -h $DB_HOSTNAME

# لـ PostgreSQL:
# psql -U $DB_USERNAME -h $DB_HOSTNAME -d sms_c_prod
```

الحلول:

قاعدة البيانات متوقفة:

```
# بدء قاعدة البيانات (MySQL/MariaDB)
systemctl start mysql

# بدء قاعدة البيانات (PostgreSQL)
systemctl start postgresql
```

بيانات اعتماد خاطئة:

تحديث التكوين:

```
export DB_USERNAME=correct_user
export DB_PASSWORD=correct_password

# إعادة تشغيل التطبيق
systemctl restart sms_c
```

مشكلة في الشبكة:

- تحقق من قواعد جدار الحماية
- تحقق من مجموعات الأمان (السحابة)
- تحقق من الاتصال عبر VPN/الشبكة

استنفاد مجموعة الاتصال:

زيادة حجم المجموعة:

```
config :sms_c, SmsC.Repo,
  pool_size: 50  # زيادة من القيمة الحالية
```

استعلامات بطيئة

الأعراض:

- زمن استعلام قاعدة البيانات مرتفع
- استجابات API بطيئة
- تراكم طابور مجموعة الاتصال

خطوات التشخيص:

1. **تحقق من سجل الاستعلامات البطيئة**:

```sql
-- MySQL/MariaDB: تفعيل سجل الاستعلامات البطيئة
SET GLOBAL slow_query_log = 'ON';
SET GLOBAL long_query_time = 1;  -- سجل الاستعلامات > 1 ثانية

-- عرض الاستعلامات البطيئة (MySQL/MariaDB)
SELECT * FROM mysql.slow_log ORDER BY query_time DESC LIMIT 10;

-- PostgreSQL: تفعيل سجل الاستعلامات البطيئة في postgresql.conf
-- log_min_duration_statement = 1000  # بالميلي ثانية
-- ثم تحقق من سجلات PostgreSQL
```

2. **تحقق من الفهارس المفقودة**:

```sql
-- تحقق من فهارس الجدول
SHOW INDEX FROM message_queues;

-- الفهارس المتوقعة:
-- - source_smsc
-- - dest_smsc
-- - send_time
-- - inserted_at
```

3. **تحقق من إحصائيات الجدول**:

```sql
-- أحجام الجداول (MySQL/MariaDB)
SELECT
  table_name,
  table_rows,
  ROUND(data_length / 1024 / 1024, 2) AS data_mb,
  ROUND(index_length / 1024 / 1024, 2) AS index_mb
FROM information_schema.tables
WHERE table_schema = 'sms_c_prod';

-- أحجام الجداول (PostgreSQL)
-- SELECT schemaname, tablename,
--
pg_size_pretty(pg_total_relation_size(schemaname||'.'||tablename))
AS size
-- FROM pg_tables WHERE schemaname = 'public';
```

الحلول:

الفهارس المفقودة:

```
CREATE INDEX idx_message_queues_source_smsc ON
message_queues(source_smsc);
CREATE INDEX idx_message_queues_dest_smsc ON
message_queues(dest_smsc);
CREATE INDEX idx_message_queues_send_time ON
message_queues(send_time);
CREATE INDEX idx_message_queues_status ON message_queues(status);
```

تجزئة الجدول:

```
-- MySQL/MariaDB
OPTIMIZE TABLE message_queues;
OPTIMIZE TABLE frontend_registrations;

-- PostgreSQL
-- VACUUM ANALYZE message_queues;
-- VACUUM ANALYZE frontend_registrations;
```

الكثير من البيانات:

تنظيف السجلات القديمة:

```
-- حذف الرسائل التي تم تسليمها والتي يزيد عمرها عن 30 يومًا
DELETE FROM message_queues
WHERE status = 'delivered'
AND deliver_time < DATE_SUB(NOW(), INTERVAL 30 DAY)
LIMIT 10000;
```

امتلاء مساحة القرص

الأعراض:

- خطأ: "القرص ممتلئ"
- لا يمكن الكتابة إلى قاعدة البيانات

- تعطل التطبيق

خطوات التشخيص:

1. **تحقق من استخدام القرص**:

```
df -h

# تحقق من دليل قاعدة بيانات SQL (MySQL/MariaDB)
du -sh /var/lib/mysql

# تحقق من دليل قاعدة بيانات SQL (PostgreSQL)
du -sh /var/lib/postgresql
```

2. **العثور على الملفات الكبيرة**:

```
# العثور على أكبر الملفات (MySQL/MariaDB)
find /var/lib/mysql -type f -exec du -h &#123;&#125; + | sort -rh
| head -20

# العثور على أكبر الملفات (PostgreSQL)
find /var/lib/postgresql -type f -exec du -h &#123;&#125; + | sort
-rh | head -20

# تحقق من ملفات السجل
du -sh /var/log/sms_c/*
```

الحلول:

تنظيف البيانات القديمة:

```
-- حذف الرسائل القديمة
DELETE FROM message_queues
WHERE inserted_at < DATE_SUB(NOW(), INTERVAL 90 DAY)
LIMIT 100000;
```

تدوير السجلات:

```
# فرض تدوير السجل
logrotate -f /etc/logrotate.d/sms_c

# مسح ملفات السجل القديمة
find /var/log/sms_c -name "*.log.*" -mtime +30 -delete
```

توسيع القرص:

- تغيير حجم الوحدة (السحابة)
- إضافة قرص جديد وتوسيع الوحدة
- نقل البيانات إلى قرص أكبر

مشكلات الاتصال بالواجهة الأمامية

الواجهة الأمامية لا تظهر كنشطة

الأعراض:

- حالة الواجهة الأمامية تظهر "منتهية"
- الواجهة الأمامية غير موجودة في القائمة النشطة
- الرسائل لا يتم تسليمها إلى الواجهة الأمامية

خطوات التشخيص:

1. **تحقق من التسجيل**:

```
curl https://api.example.com:8443/api/frontends/active | grep frontend_name
```

2. **تحقق من سجلات الواجهة الأمامية**:

- هل الواجهة الأمامية تستعلم `/api/frontends/register`؟
- أي أخطاء في API؟
- تكرار التسجيل (يجب أن يكون كل 60 ثانية)

3. **تحقق من سجلات API**:

```
grep "frontend.*register" /var/log/sms_c/application.log | tail
-20
```

الحلول:

الواجهة الأمامية لا تسجل:

اختبار التسجيل اليدوي:

```
curl -X POST https://api.example.com:8443/api/frontends/register \
  -H "Content-Type: application/json" \
  -d '&amp;#123;
    "frontend_name": "uk_gateway",
    "frontend_type": "smpp",
    "ip_address": "10.0.1.50"
  &amp;#125;'
```

إذا كانت ناجحة، فإن المشكلة في كود/تكوين الواجهة الأمامية.

تسجيل ينتهي الوقت:

تنتهي صلاحية الواجهات الأمامية بعد 90 ثانية. تأكد من التسجيل كل 60 ثانية:

```
# يجب على الواجهة الأمامية استدعاء التسجيل كل 60 ثانية
while True:
    register_with_smsc()
    time.sleep(60)
```

مشكلات الشبكة:

- تحقق من جدار الحماية بين الواجهة الأمامية وAPI
- تحقق من حل DNS
- اختبار باستخدام curl من خادم الواجهة الأمامية

الواجهة الأمامية تتصل/تنفصل بشكل متكرر

الأعراض:

- حالة الواجهة الأمامية تتقلب بين النشطة/المنتهية
- عدد التسجيلات مرتفع في السجل
- اتصال غير مستقر

خطوات التشخيص:

1. **تحقق من صحة الواجهة الأمامية**:

- هل عملية الواجهة الأمامية مستقرة؟
- أي أعطال أو إعادة تشغيل؟
- مشكلات الموارد (وحدة المعالجة المركزية/الذاكرة)؟

2. **تحقق من استقرار الشبكة**:

```
# تحقق من فقدان الحزم
ping -c 100 api.example.com

# تحقق من إعادة تعيين الاتصال
netstat -s | grep -i reset
```

3. **تحقق من توقيت التسجيل**:

- هل هو متكرر جدًا؟ (كل بضع ثوان)
- هل هو نادر جدًا؟ (> 90 ثانية)

الحلول:

الواجهة الأمامية غير مستقرة:

- إصلاح مشكلات تطبيق الواجهة الأمامية
- زيادة موارد الواجهة الأمامية
- تحقق من سجلات الواجهة الأمامية للأخطاء

مشكلات الشبكة:

- تحقق من الاتصال المتقطع
- مراجعة سجلات جدار الحماية
- تحقق من اختبارات صحة موازن الحمل

توقيت التسجيل خاطئ:

تصحيح الفترة:

```
REGISTRATION_INTERVAL = 60  # ثواني
```

مشكلات الشحن/الفوترة

فشل الشحن

الأعراض:

- زيادة `sms_c_charging_failed_count`
- سجل الأحداث يظهر "charging_failed"
- الرسائل موسومة كـ `charge_failed: true`

خطوات التشخيص:

1. **تحقق من اتصال OCS**:

```
# اختبار API OCS
curl -X POST http://ocs.example.com:2080/jsonrpc \
  -H "Content-Type: application/json" \
  -d '&#123;
    "method": "SessionSv1.Ping",
    "params": [],
    "id": 1
  &#125;'
```

المتوقع: `{"result":"Pong"}`

2. **تحقق من سجلات OCS**:

```
tail -f /var/log/ocs/ocs.log
```

3. **تحقق من التكوين**:

```
# تحقق من عنوان URL لـ OCS
grep ocs_url config/runtime.exs
```

الحلول:

غير متاح OCS:

```
# تحقق من حالة OCS
systemctl status ocs

# بدء إذا لزم الأمر
systemctl start ocs
```

خطأ في التكوين:

تحديث التكوين:

```
config :sms_c,
  ocs_url: "http://correct-host:2080/jsonrpc",
  ocs_tenant: "correct_tenant"
```

تعطيل الشحن مؤقتًا:

```
config :sms_c,
  default_charging_enabled: false
```

إعادة تشغيل التطبيق.

مشكلات الحساب:

- تحقق من وجود الحساب في OCS
- تحقق من أن الحساب لديه رصيد
- تحقق من أن خطط التسعير مكونة

الشحن بطيء جدًا

الأعراض:

- `sms_c_charging_succeeded_duration` p95 > 500ms
- معالجة الرسائل بطيئة عند تمكين الشحن
- سريعة عند تعطيل الشحن

خطوات التشخيص:

1. **تحقق من زمن تأخير الشحن**:

```
histogram_quantile(0.95, sms_c_charging_succeeded_duration_bucket)
```

2. **تحقق من أداء OCS**:

```
# زمن استجابة OCS
curl -w "%&#123;time_total&#125;\n" -X POST
http://ocs.example.com:2080/jsonrpc \
  -H "Content-Type: application/json" \
  -d '&#123;"method":"SessionSv1.Ping","params":[],"id":1&#125;'
```

3. **تحقق من زمن تأخير الشبكة**:

```
# اختبار استضافة OCS
ping -c 10 ocs.example.com
```

الحلول:

بطيء OCS:

- تحسين تكوين OCS
- إضافة موارد OCS
- استخدام محرك تسعير أسرع

زمن تأخير الشبكة:

- نشر OCS بالقرب من SMS-C

- استخدام مسار شبكة مباشر
- تجنب VPN/الأنفاق إذا كان ذلك ممكنًا

مهلة قصيرة جدًا:

زيادة المهلة:

```
config :sms_c,
  ocs_timeout: 5000  # 5 ثواني
```

مشكلات البحث عن ENUM

فشل عمليات البحث عن ENUM

الأعراض:

- `sms_c_enum_lookup_stop_duration` تظهر فشلاً
- سجل الأحداث يظهر أخطاء ENUM
- المسارات مع `enum_result_domain` لا تتطابق

خطوات التشخيص:

1. **تحقق من تكوين ENUM**:

```
grep -A 10 "enum_" config/runtime.exs
```

2. **اختبار اتصال DNS**:

```
# اختبار خادم DNS
dig @8.8.8.8 e164.arpa

# اختبار استعلام ENUM
# لـ: 15551234567+
dig @8.8.8.8 NAPTR 7.6.5.4.3.2.1.5.5.5.1.e164.arpa
```

3. **تحقق من خادم DNS**:

```
# هل DNS المخصص قابل للوصول؟
ping 10.0.1.53

# اختبار المنفذ
nc -zv 10.0.1.53 53
```

الحلول:

خادم DNS غير قابل للوصول:

استخدام DNS بديل:

```
config :sms_c,
  enum_dns_servers: [
    &#123;"8.8.8.8", 53&#125;,  # Google Public DNS
    &#123;"1.1.1.1", 53&#125;   # Cloudflare DNS
  ]
```

مجال ENUM خاطئ:

تحديث المجال:

```
config :sms_c,
  enum_domains: ["e164.arpa"]  # استخدام المجال القياسي
```

مهلة قصيرة جدًا:

زيادة المهلة:

```
config :sms_c,
  enum_timeout: 10000  # 10 ثواني
```

تعطيل ENUM (إذا لم يكن مطلوبًا):

```
config :sms_c,
  enum_enabled: false
```

ENUM مشكلا��ذاكرة التخزين المؤقت

الأعراض:

- معدل نجاح التخزين المؤقت منخفض (< 70%)
- حجم التخزين المؤقت ينمو بلا حدود
- استخدام الذاكرة مرتفع

خطوات التشخيص:

1. **تحقق من إحصائيات التخزين المؤقت**:

```
# معدل نجاح التخزين المؤقت
rate(sms_c_enum_cache_hit_count[5m]) /
(rate(sms_c_enum_cache_hit_count[5m]) +
rate(sms_c_enum_cache_miss_count[5m]))

# حجم التخزين المؤقت
sms_c_enum_cache_size_size
```

2. **تحقق من نمط الحركة**:

- هل الأرقام تتكرر؟
- هل فترة التخزين المؤقت مناسبة؟

الحلول:

(متوقع) **معدل نجاح منخفض**:

- حركة المرور إلى أرقام فريدة (طبيعي)
- 70 > المراقبة ولكن لا تنبه إذا كانت%

التخزين المؤقت ينمو:

مسح التخزين المؤقت عبر صفحة اختبار NAPTR أو إعادة تشغيل التطبيق.

استخدام الذاكرة مرتفع:

- متوقع مع تخزين مؤقت كبير
- مراقبة استخدام الذاكرة الكلية للنظام

- النظر في تعديل فترة التخزين المؤقت

مشكلات العن��ود

العقدة لا يمكنها الانضمام إلى العنقود

الأعراض:

- تشغيل عقدة واحدة
- استعلامات العنقود تعيد نتائج محلية فقط
- أخطاء توزيع Erlang

خطوات التشخيص:

1. **تحقق من أسماء العقد**:

```
# في وحدة التحكم IEx
Node.self()
# المتوقع: :sms@node1.example.com

Node.list()
# المتوقع: قائمة بالعقد الأخرى
```

2. **تحقق من ملف الكوكي**:

```
# تحقق من ملف الكوكي
cat ~/.erlang.cookie

# تحقق من أنه نفس الشيء على جميع العقد
```

3. **تحقق من الشبكة**:

```
# هل يمكن للعقد الوصول إلى بعضها البعض؟
ping node2.example.com

# تحقق من المنافذ
nc -zv node2.example.com 4369
nc -zv node2.example.com 9100-9200
```

الحلول:

عدم تطابق الكوكي:

تعيين نفس الكوكي على جميع العقد:

```
export ERLANG_COOKIE=same_secret_value_here

# أو تحديث ~/.erlang.cookie
echo "same_secret_value_here" > ~/.erlang.cookie
chmod 400 ~/.erlang.cookie
```

جدار الحماية يحظر:

فتح المنافذ المطلوبة:

```
# EPMD
iptables -A INPUT -p tcp --dport 4369 -j ACCEPT

# توزيع Erlang
iptables -A INPUT -p tcp --dport 9100:9200 -j ACCEPT
```

مشكلات DNS:

استخدام عناوين IP بدلاً من أسماء المضيفين:

```
config :sms_c,
  cluster_nodes: [
    :"sms@10.0.1.10",
    :"sms@10.0.1.11"
  ]
```

انقسام العنقود

الأعراض:

- العقد تعمل ولكن غير متصلة
- بيانات مختلفة على عقد مختلفة
- عدم تناسق Mnesia

خطوات التشخيص:

1. **تحقق من اتصال العقد**:

```
# على كل عقدة (IEx)
Node.list()
```

2. **تحقق من Mnesia**:

```
:mnesia.system_info(:running_db_nodes)
```

الحلول:

إعادة الاتصال بالعقد:

```
# إيقاف جميع العقد
systemctl stop sms_c

# بدء عقدة واحدة أولاً
systemctl start sms_c  # على node1

# الانتظار حتى تبدأ بالكامل، ثم بدء الآخرين
systemctl start sms_c  # على node2
systemctl start sms_c  # على node3
```

عدم تناسق Mnesia:

- تصدير المسارات من العقدة الصحيحة
- إيقاف جميع العقد
- حذف دليل Mnesia

- بدء العقد
- استيراد المسارات

مشكلات API

لا تستج��ب API

الأعراض:

- مهلة الاتصال
- اتصال مرفوض
- عدم وجود استجابة

خطوات التشخيص:

1. **تحقق من عملية API**:

```
# هل التطبيق يعمل؟
systemctl status sms_c

# تحقق من المنافذ المستمعة
netstat -tlnp | grep 8443
```

2. **تحقق من جدار الحماية**:

```
# تحقق من iptables
iptables -L -n | grep 8443

# اختبار الاتصال المحلي
curl -k https://localhost:8443/api/status
```

3. **تحقق من تكوين TLS**:

```
# تحقق من وجود الشهادة
ls -l priv/cert/server.crt priv/cert/server.key

# تحقق من صلاحية الشهادة
openssl x509 -in priv/cert/server.crt -noout -dates
```

الحلول:

التطبيق غير قيد التشغيل:

```
systemctl start sms_c
```

جدار الحماية يحظر:

```
# السماح بمنفذ API
iptables -A INPUT -p tcp --dport 8443 -j ACCEPT
```

مشكلات الشهادة:

إنشاء شهادة جديدة (انظر دليل التكوين).

منفذ خاطئ:

تحقق من التكوين:

```
grep "port:" config/runtime.exs
```

تعيد أخطاء API 500

الأعراض:

- خطأ داخلي في الخادم
- رمز الحالة 500
- خطأ في السجلات

خطوات التشخيص:

1. **تحقق من سجلات التطبيق**:

```
tail -100 /var/log/sms_c/application.log | grep "\[error\]"
```

2. **تحقق من قاعدة البيانات**:

```
mysql -u sms_user -p -e "SELECT 1"
```

3. **تحقق من الموارد**:

```
# الذاكرة
free -h

# وحدة المعالجة المركزية
top -b -n 1

# القرص
df -h
```

الحلول:

قاعدة البيانات غير متاحة:

- بدء قاعدة البيانات
- إصلاح مشكلة الاتصال

نفاد الذاكرة:

- إعادة تشغيل التطبيق
- زيادة ذاكرة النظام
- تحقق من تسرب الذاكرة

خطأ في التطبيق:

- تحقق من الخطأ المحدد في السجلات
- إصلاح مشكلة التكوين
- إعادة تشغيل التطبيق

مشكلات واجهة الويب

لا يمكن الوصول إلى واجهة الويب

الأعراض:

- مهلة الاتصال
- غير موجود 404
- الصفحة لا تحمل

خطوات التشخيص:

1. **تحقق من حالة التطبيق**:

```
systemctl status sms_c
```

2. **تحقق من المنفذ**:

```
netstat -tlnp | grep 80
```

3. **تحق من عنوان URL**:

- هل اسم المضيف صحيح؟
- هل المنفذ صحيح؟
- HTTP مقابل HTTPS؟

الحلول:

منفذ خاطئ:

تحقق من التكوين:

```
grep "control_panel" config/runtime.exs
```

الوصول إلى المنفذ الصحيح (الافتراضي: 80 أو 4000).

التطبيق غير قيد التشغيل:

```
systemctl start sms_c
```

جدار الحماية:

```
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

LiveView لا يتم تحديثه

الأعراض:

- تحميل الصفحة ولكن لا يتم التحديث
- البيانات قديمة
- أخطاء WebSocket في وحدة تحكم المتصفح

خطوات التشخيص:

1. **تحقق من وحدة تحكم المتصفح**:

- افتح أدوات المطور (F12)
- ابحث عن أخطاء WebSocket
- تحقق من علامة الشبكة لطلبات فاشلة

2. **تحقق من تكوين الوكيل**:

إذا كنت تستخدم وكيل عكسي، تأكد من دعم WebSocket:

```
location /live &#123;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
&#125;
```

الحلول:

WebSocket محظور:

- تكوين الوكيل لـ WebSocket
- تحقق من جدار الحماية
- تحقق من ملحقات المتصفح

تحديث الصفحة:

- تحديث صعب (Ctrl+F5)
- مسح ذاكرة التخزين المؤقت للمتصفح

مشكلات موارد النظام

استخدام وحدة المعالجة المركزية مرتفع

الأعراض:

- وحدة المعالجة المركزية باستمرار > 80%
- النظام بطيء
- التطبيق غير مستجيب

خطوات التشخيص:

1. **تحقق من العملية**:

```
top -b -n 1 | grep beam.smp
```

2. **تحقق من المقاييس**:

```
# معدل معالجة الرسائل
rate(sms_c_message_received_count[5m])

# عمليات التوجيه
rate(sms_c_routing_route_matched_count[5m])
```

الحلول:

حركة مرور عالية:

- التوسع أفقيًا (إضافة عقد)
- التوسع عموديًا (إضافة وحدة معالجة مركزية)

توجيه غير فعال:

- تقليل عدد المسارات
- تحسين معايير المسار

الكثير من عمليات البحث عن ENUM:

- تحقق من معدل نجاح التخزين المؤقت
- النظر في تعطيل إذا لم يكن مطلوبًا

استخدام الذاكرة مرتفع

الأعراض:

- 90 < استخدام الذاكرة%
- تعطل التطبيق
- أخطاء نفاد الذاكرة

خطوات التشخيص:

1. **تحقق من الذاكرة**:

```
free -h

ps aux | grep beam.smp
```

2. **تحقق من أحجام التخزين المؤقت**:

```
sms_c_enum_cache_size_size
```

الحلول:

ذاكرة التخزين المؤقت ENUM كبيرة جدًا:

- مسح التخزين المؤقت

- تقليل فترة التخزين المؤقت
- إذا لم يكن مطلوبًا ENUM تعطيل

طابور الدفعة في زيادة:

```
# تحقق من إحصائيات العامل (IEx)
SmsC.Messaging.BatchInsertWorker.stats()
```

إذا كان الطابور كبيرًا، قم بتفريغه يدويًا أو إعادة التشغيل.

إضافة الذاكرة:

- التوسع عموديًا
- إضافة تبادل (مؤقت)

تسرب الذاكرة:

- إعادة تشغيل التطبيق
- الإبلاغ عن المشكلة للتحقيق

للحصول على مساعدة إضافية، استشر:

- الإجراءات اليومية - دليل العمليات
- خيارات التكوين - دليل التكوين
- إعداد المراقبة - دليل المقاييس
- سجلات التطبيق - `/var/log/sms_c/application.log`