

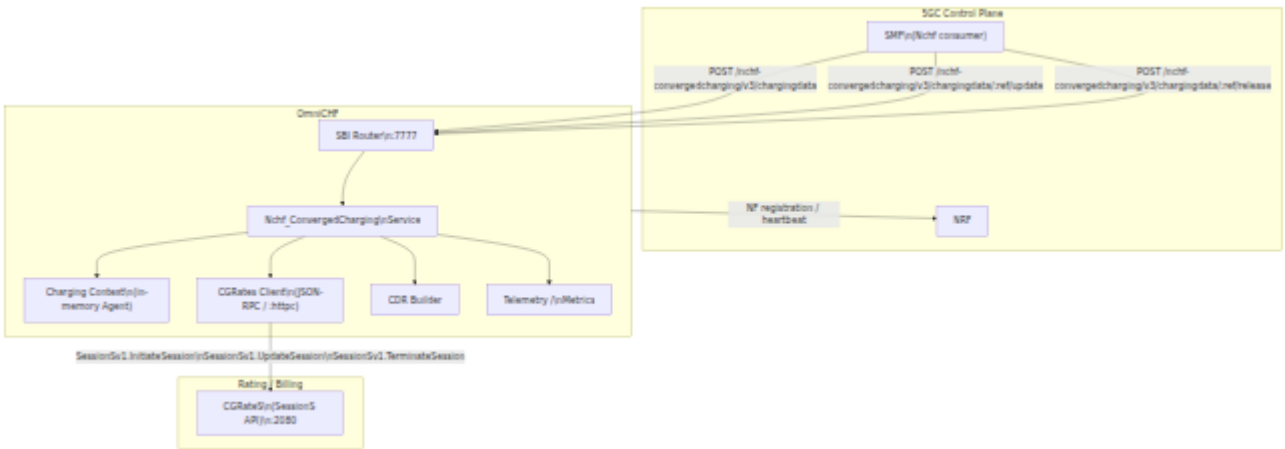
OmniCHF دليل عمليات

جدول المحتويات

1. نظرة عامة على المكونات
2. والمراجع الفنية GPP دور 3
3. SBI نقاط نهاية
4. مرجع التكوين
5. الإجراءات الرئيسية
6. المراقبة
7. القيود المعروفة
8. استكشاف الأخطاء وإصلاحها

نظرة عامة على المكونات

شحنًا CHF توفر 32.291 GPP TS المعرفة في 3 (CHF) ووظيفة الشحن OmniCHF يطبق Nchf_ConvergedCharging عبر خدمة 5G PDU متقاربًا عبر الإنترنت وغير المتصل لجلسات لتفويض CGRateS SessionS ل JSON-RPC إلى استدعاءات G يقوم بتحويل طلبات الشحن 5 عند إنهاء الجلسة (CDRs) الائتمان وإدارة الجلسات، وينتج سجلات تفاصيل المكالمات



دورة حياة جلسة الشحن

chargingDataRef (UUID) مع جلسة شحن واحدة، يتم تتبعها بواسطة PDU تتوافق كل جلسة يتم الاحتفاظ بحالة الجلسة في وكيل في الذاكرة ولا يتم الاحتفاظ بها. يؤدي إعادة التشغيل إلى فقدان جميع حالات الجلسات النشطة.

الحالة	الزناد	إجراء التخزين
تم الإنشاء	POST /chargingdata	تم إنشاء السياق، تم استدعاء CGRateS InitiateSession
تم التحديث	POST /chargingdata/:ref/update	تم تحديث السياق (تم تجميع الاستخدام، تم زيادة التسلسل)
تم الإفراج	POST /chargingdata/:ref/release	وتسجيله، تم استدعاء CDR تم بناء CGRateS TerminateSession، تم حذف السياق

والمراجع الفنية GPP دور 3

العنصر	المرجع
تعريف NF CHF	القسم 6.2.16 3GPP TS 23.501
خدمة Nchf_ConvergedCharging	3GPP TS 32.291
إجراء إنشاء بيانات الشحن	القسم 6.1.3.2.1 3GPP TS 32.291
إجراء تحديث بيانات الشحن	القسم 6.1.3.2.2 3GPP TS 32.291
إجراء إفراج بيانات الشحن	القسم 6.1.3.2.3 3GPP TS 32.291
ChargingDataRequest / Response نموذج بيانات	القسم 6.1.6 3GPP TS 32.291
PDU 5G لجلسات CDR تنسيق	3GPP TS 32.290
المشترك SBI إطار عمل	3GPP TS 29.500
NF مع NRF تسجيل	3GPP TS 29.510

SBI نقاط نهاية

مسار القاعدة: /nchf-convergedcharging/v3

الطريقة	المسار	الوصف
POST	/chargingdata	إنشاء جلسة شحن (طلب أولي). يخصص chargingDataRef، CGRateS، يبدأ جلسة ويعيد الوحدات الممنوحة.
POST	/chargingdata/{chargingDataRef}/update	تحديث جلسة شحن (طلب مؤقت). يبلغ عن الاستخدام الحالي. ويطلب ائتمانيًا إضافيًا.
POST	/chargingdata/{chargingDataRef}/release	إفراج عن جلسة شحن (طلب نهائي). يبلغ عن الاستخدام النهائي، ينتج CDR، ينتج CGRateS.

الحقول الرئيسية – ChargingDataRequest

الوصف	مستخدم في	النوع	الحقل
، على سبيل المثال) SUPI (imsi-999700000000001). يستخدم كمعرف حساب CGRateS.	إنشاء، تحديث، إفراج	string	subscriberIdentifier
NF معلومات مستهلك إذا SUPI مصدر احتياطي لـ كان subscriberIdentifier غائبًا.	إنشاء	object	nfConsumerIdentification
PDU: DNN تفاصيل جلسة S-NSSAI، نوع RAT، QoS PDU معرف ونوع جلسة.	إنشاء، تحديث، إفراج	object	pDUSESSIONChargingInformation
حاويات الاستخدام المبلغ عنها. يتم استخدام usedUnitContainer لعنصر الأول لاستخراج الحجم والمدة.	تحديث، إفراج	array	multipleUnitUsage
INITIAL_REQUEST، UPDATE_REQUEST، أو TERMINATION_REQUEST	الكل	string	requestType

الحقول الرئيسية – ChargingDataResponse

الحقل	النوع	موجود	الوصف
<code>invocationSequenceNumber</code>	integer	إنشاء، تحديث	رقم التسلسل لهذه الاستجابة. انظر) ثابت على 1 عند الإنشاء CHF-M1). يتم زيادته في كل تحديث.
<code>invocationResult</code>	object	إنشاء، تحديث	دائمًا {"resultCode": "SUCCESS"} في المسار السعيد.
<code>sessionId</code>	string	إنشاء، تحديث	<code>chargingDataRef</code> (UUID) المخصص لهذه الجلسة.
<code>multipleUnitInformation</code>	array	إنشاء، تحديث	الوحدات الممنوحة. تحتوي على <code>resultCode</code> ، إدخال واحد مع <code>grantedUnit</code> (totalVolume, time)، ثابت على <code>ratingGroup</code> و CHF-L2)، انظر 1،

مرجع التكوين

(عادةً) يتم تعيين جميع المعلمات عبر بيئة التطبيق (`config/runtime.exs`).

```
config :omnichf,  
  sbi_scheme:      "http",  
  sbi_addr:        "127.0.0.14",  
  sbi_port:        7777,  
  nrf_uri:         "http://127.0.0.10:7777",  
  mcc:             "999",  
  mnc:             "70",  
  heartbeat_interval: 10_000,  
  cgrates_enabled: false,  
  cgrates_url:     "http://localhost:2080/jsonrpc",  
  cgrates_tenant:  "cgrates.org",  
  cgrates_timeout: 5000
```

جدول المعلمات

المعلمة	الافتراضي	النوع	
sbi_scheme	"http"	string	ستمع
sbi_addr	"127.0.0.14"	string	كنوان خادم
sbi_port	7777	integer	منفذ خادم
nrf_uri	"http://127.0.0.10:7777"	string	URI مجل وhea
mcc	"999"	string	مول. ملف م إلى ندمية
mnc	"70"	string	مولة. ملف م إلى ندمية
heartbeat_interval	10_000	integer (ms)	للبات hear
cgrates_enabled	false	boolean	كامل CGR fal باءات فدرها نعينه إلى مثيل

المعلمة	الافتراضي	النوع	
<code>cgrates_url</code>	<code>"http://localhost:2080/jsonrpc"</code>	string	عنوان JSON CGRates يكون <code>cgrates_url</code> <code>true</code>
<code>cgrates_tenant</code>	<code>"cgrates.org"</code>	string	متأجر جميع بيانات تُحقل متأجر ن في
<code>cgrates_timeout</code>	<code>5000</code>	integer (ms)	طلب بيانات CGRates ن من صحة <code>min</code> <code>3000</code> صحية

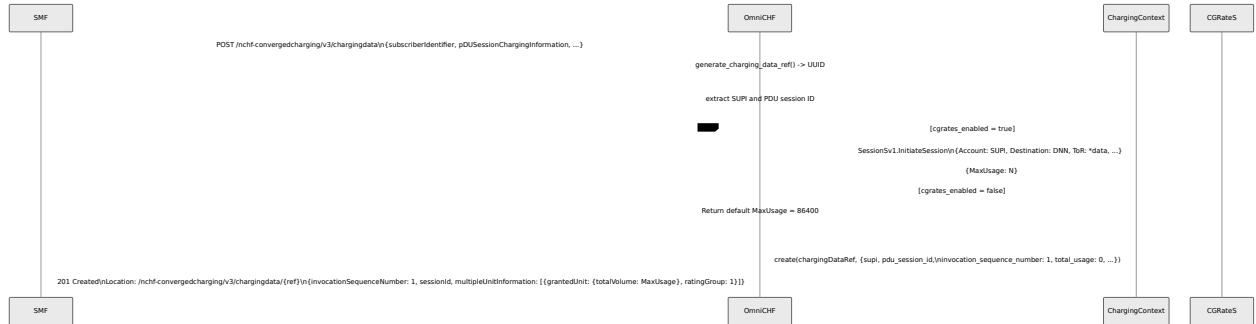
CGRateS ملاحظات تكامل

في وضع التجاوز: يتم قبول OmniCHF يعمل، `false` هو `cgrates_enabled` عندما يكون جميع طلبات الشحن ومنح 86,400 وحدة (وقت أو حجم) دون أي تقييم أو تفويض. هذا مناسب CGRateS. للاختبار في المختبر والتكامل عندما لا يتوفر.

(CHF-M5 انظر القيود) `http`: Erlang المدمج في HTTP عميل CGRateS تستخدم اتصالات HTTP اتصال CGRateS لا يدعم هذا العميل لجميع الاتصالات. تحت الحمل العالي، يفتح كل طلب جديد ويغلقه، مما قد يصبح عنق زجاجة.

الإجراءات الرئيسية

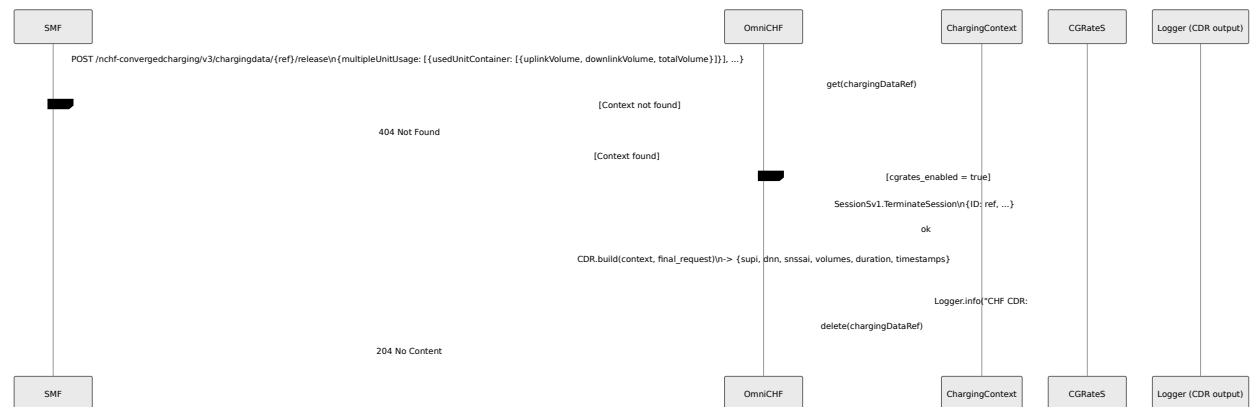
إنشاء جلسة الشحن (الأولية)



تحديث جلسة الشحن (المؤقت)



إفراج عن جلسة الشحن (النهائي)



CGRateS تخطيط أحداث

كما يلي CGRateS SessionS إلى حقول أحداث G بتخطيط حقول الشحن 5 OmniCHF يقوم

حقل CGRateS	المصدر	
Account	subscriberIdentifier (SUPI)	يترجع إلى nfConsum
Subject	subscriberIdentifier (SUPI)	ن الشيء ك
Destination	pduSessionInformation.dnnId أو .dnn	بكة البيانات
ToR	pduSessionInformation.pduType	*data دائمًا
RequestType	requestType	خطيطه إلى
Usage	usedUnitContainer.totalVolume أو مجموع uplink+downlink أو time	صفرية تفوز
OriginID	chargingDataRef	نلسة UUID
OriginHost	ثابت "OmniCHF"	
SUPI	subscriberIdentifier	قل امتداد 5
DNN	pduSessionInformation.dnnId	قل امتداد 5
S-NSSAI_SST	pduSessionInformation.sNSSAI.sst	قل امتداد 5
S-NSSAI_SD	pduSessionInformation.sNSSAI.sd	قل امتداد 5
5QI	pduSessionInformation.qoSInformation.5qi	قل امتداد 5
RATType	pduSessionInformation.ratType	:الافتراضي
PDUSessionID	pduSessionInformation.pduSessionID	قل امتداد 5
PDUSessionType	pduSessionInformation.pduType	:الافتراضي

CDR حقول

تحتوي INFO عند الإفراج عنها ويتم إصدارها إلى سجل التطبيق على مستوى CDRs يتم بناء على CDR خريطة:

الحقل	المصدر
record_type	"G_PDU_SESSION ثابت: 5"
supi	سياق الشحن
dnn	pduSessionInformation.dnnId أو .dnn
snsai	{sst, sd} من pduSessionInformation.sNSSAI
qos_5qi	pduSessionInformation.qoSInformation.5qi
rat_type	pduSessionInformation.ratType
pdu_session_id	سياق الشحن
pdu_session_type	pduSessionInformation.pduType
volume_uplink	usedUnitContainer.uplinkVolume
volume_downlink	usedUnitContainer.downlinkVolume
volume_total	usedUnitContainer.totalVolume (أو uplink+downlink)
duration	usedUnitContainer.time (أو فرق الساعة إذا كانت صفر)
start_time	للجلسة created_at طابع زمني
end_time	الساعة عند وقت الإفراج
charging_data_ref	للجلسة UUID

المراقبة

أحداث القياس

الحدث	القياسات	العلامات	الوصف
<code>[:omnichf, :charging, :initial]</code>	<code>count</code>	<code>supi</code>	يتم إطلاقه في كل طلب إنشاء
<code>[:omnichf, :charging, :update]</code>	<code>count</code>	<code>ref</code>	يتم إطلاقه في كل طلب تحديث
<code>[:omnichf, :charging, :release]</code>	<code>count</code>	<code>ref</code>	يتم إطلاقه في كل طلب إفراج
<code>[:omnichf, :charging, :creates]</code>	<code>count</code>	<code>result (success/failure)</code>	نتيجة عملية الإنشاء
<code>[:omnichf, :charging, :updates]</code>	<code>count</code>	<code>result</code>	نتيجة عملية التحديث
<code>[:omnichf, :charging, :releases]</code>	<code>count</code>	<code>result</code>	نتيجة عملية الإفراج
<code>[:omnichf, :cgrates, :request]</code>	<code>count, duration_ms</code>	<code>operation, result</code>	لكل استدعاء JSON-RPC لـ CGRateS
<code>[:omnichf, :cgrates, :status]</code>	<code>status (1/0)</code>	—	صحة الاتصال بـ CGRateS

الوصف	العلامات	القياسات	الحدث
			:health]
مقياس: الجلسات النشطة للشحن	—	count	[:omnichf, :sessions, :active]
حالة تسجيل NRF	nf_type	status (1/0)	[:omni5g, :nrf, :registration]

Prometheus مقاييس

مقاييس الشحن

المقياس	النوع	العلامات	الوصف
<code>omni_chf.charging.initial.count</code>	counter	supi	إنشاء جلسات الشحن
<code>omni_chf.charging.update.count</code>	counter	ref	تحديث جلسات الشحن
<code>omni_chf.charging.release.count</code>	counter	ref	إفراج عن جلسات الشحن
<code>omni_chf.charging.create.total</code>	counter	result	إجمالي إنشاء جلسات الشحن
<code>omni_chf.charging.update.total</code>	counter	result	إجمالي تحديث جلسات الشحن
<code>omni_chf.charging.release.total</code>	counter	result	إجمالي إفراج عن جلسات الشحن
<code>omni_chf.sessions.active.count</code>	gauge	--	عدد جلسات الشحن النشطة

مقياس CGRateS

المقياس	النوع	العلامات	ف
<code>omni_chf.cgrates.calls.count</code>	counter	<code>method,</code> <code>result</code>	دعوات JSON RPC CGF
<code>omni_chf.cgrates.latency.milliseconds</code>	gauge	--	زمن جواب دعوات CGF
<code>omni_chf.cgrates.health</code>	gauge	--	صحة ال CGF (1= غير نصل
<code>omni_chf.cgrates.requests.total</code>	counter	<code>operation,</code> <code>result</code>	مالي طلبات JSON RPC CGF
<code>omni_chf.cgrates.request.duration_ms</code>	distribution	<code>operation</code>	طلب CGF ملي ثانية سام: ,10 ,50 ,100 ,250 ,500 ,1000 (250

مقياس NRF

المقياس	النوع	العلامات	الوصف
omni_chf.nrf.registration.status	gauge	nf_type	NRF حالة تسجيل (مسجل،=1) (غير مسجل=0)

مقياس BEAM VM

المقياس	النوع	الوصف
<code>beam.memory.total</code>	gauge	بالبايت BEAM إجمالي ذاكرة
<code>beam.memory.processes</code>	gauge	الذاكرة المستخدمة من قبل عمليات Erlang
<code>beam.memory.processes_used</code>	gauge	الذاكرة المستخدمة فعليًا من قبل العمليات
<code>beam.memory.system</code>	gauge	ذاكرة النظام
<code>beam.memory.atom</code>	gauge	إجمالي ذاكرة الذرات
<code>beam.memory.atom_used</code>	gauge	ذاكرة الذرات المستخدمة
<code>beam.memory.binary</code>	gauge	ذاكرة ثنائية
<code>beam.memory.code</code>	gauge	ذاكرة الشيفرة
<code>beam.memory.ets</code>	gauge	ETS ذاكرة جدول
<code>beam.processes.count</code>	gauge	Erlang عدد عمليات
<code>beam.ports.count</code>	gauge	Erlang عدد منافذ
<code>beam.atom.count</code>	gauge	عدد الذرات
<code>beam.vm.uptime</code>	gauge	بالثواني VM وقت تشغيل

أنماط السجل

المستوى	النمط	المعنى
info	CHF Create: ref=<UUID> supi=<SUPI> pdu_session=<N>	تم بدء إنشاء ناجح
info	CHF Update: ref=<UUID>	تم استلام طلب التحديث
info	CHF Release: ref=<UUID>	تم استلام طلب الإفراج
info	CHF CDR: %{\dots}	عند الإفراج CDR تم إصدار
info	Initiating CGRateS session for <ref>, account: <SUPI>	تم إرسال CGRateS InitiateSession
info	CGRateS authorized <N> units for session <ref>	تم منح الائتمان
info	CGRateS session <ref> terminated successfully	تم إنهاء CGRateS TerminateSession بنجاح
warning	CGRateS integration disabled, returning default authorization	وضع التجاوز نشط
warning	CHF Update: unknown ref=<UUID>	تحديث لجلسة غير موجودة
warning	CHF Release: unknown ref=<UUID>	إفراج لجلسة غير موجودة
error	CHF Create failed: <reason>	فشل عملية الإنشاء
error	CHF Update failed: <reason>	فشل عملية التحديث

المستوى	النمط	المعنى
error	CGRateS InitiateSession failed for <ref>: <reason>	عن CGRateS خطأ الإنشاء
error	CGRateS HTTP error <status>: <body>	CGRateS غير 200 من
error	CGRateS HTTP request failed: <reason>	خطأ في الشبكة إلى CGRateS

القيود المعروفة

المعرف	الشدة	الوصف
CHF-M1	متوسط	<p>ثابت على 1 في استجابة الإنشاء <code>invocationSequenceNumber</code> يجب أن يبدأ رقم التسلسل من 1 TS 32.291 (الأولية). وفقًا لـ ويزداد في الاستجابات اللاحقة، وهو ما يحدث في التحديث. المشكلة هي أنه إذا أرسل مستهلك إنشاء مع <code>invocationSequenceNumber</code> في الطلب، فإن تلك القيمة لا يتم فحصها أو التحقق منها.</p>
CHF-M3	متوسط	<p><code>invocationTimeStamp</code> غائب من <code>ChargingDataResponse</code>. هذا الحقل إلزامي في جسم الاستجابة. سيحصل TS 32.291 وفقًا لـ المستهلكون الصارمون الذين يتطلبون هذا الحقل على استجابة غير مكتملة.</p>
CHF-M5	متوسط	<p>بدلاً من Erlang في <code>httpc</code> : HTTP عميل <code>CGRateS</code> يستخدم عميل لجميع الاتصالات؛ يفتح كل استدعاء <code>httpc</code> : لا يدعم Finch. جديد ويغلقه. تحت الحم <code>httpc</code> (العديد من اتصال JSON-RPC جلسات الشحن المتزامنة)، ستزداد زمن استجابة استدعاءات وتصبح تكاليف إعداد الاتصال كبيرة. راقب <code>CGRateS</code> <code>omni_chf.cgrates.request.duration_ms</code>.</p>
CHF-L1	منخفض	<p><code>ChargingDataResponse</code> في <code>triggers</code> لا يتم تضمين حقل لإرسال تحديث مؤقت <code>instruct SMF</code> يمكن أن TS 32.291 وفقًا لـ بدون المحفزات، (QoS حد حجم، حد زمن، تغيير) عند أحداث معينة سياسته المحلية لتحديد متى يجب إرسال التحديثات <code>SMF</code> يستخدم.</p>
CHF-L2	منخفض	<p>ثابت على 1. <code>multipleUnitInformation</code> في <code>ratingGroup</code> عادةً ما تحتوي النشر الفعلي على مجموعات تقييم متعددة لكل جلسة يتم نسب جميع الاستخدامات (واحدة لكل تدفق بيانات خدمة) PDU في <code>ratingGroup</code> إلى مجموعة التقييم 1 بغض النظر عن قيم في الطلب <code>multipleUnitUsage</code>.</p>
CHF-L3	منخفض	<p>TS 32.290، وفقًا لـ GPP في تنسيق 3 <code>chargingId</code> لا يتم إنشاء الذي يتوافق مع معرف <code>chargingId</code> يجب أن تحمل سجلات الشحن UUID يتم استخدام <code>SMF</code> المعين بواسطة PDU جلسة</p>

المعرف	الشدة	الوصف
		بدلاً من ذلك، مما قد يسبب مشاكل في <code>charging_data_ref</code> <code>chargingId</code> الارتباط في أنظمة الفوترة السفلية التي تتوقع تنسيق 3GPP.
CHF-L4	منخفض	<code>chargingID</code> و <code>recordingEntity</code> مفقود الحقول CDR سجل ستحتاج أنظمة الوساطة أو الفوترة. TS 32.290 المطلوبة من قبل السفلية التي تتوقع هذه الحقول إلى تحمل غيابها أو تكوينها لتعتبرها اختيارية.
CHF-L5	منخفض	يتم إصدار CDR، لم يتم تنفيذ الشحن غير المتصل وإخراج ملف لا يوجد إخراج <code>Logger.info</code> فقط إلى سجل التطبيق عبر CDRs ولا نقل إلى بوابة مجال، <code>ASN.1</code> قائمة على الملفات، ولا ترميز CDR الفوترة. من أجل الشحن غير المتصل في الإنتاج، يجب على شاحن CHF لـ CDR جمع خطوط سجل (<code>Fluentd</code> ، <code>Vector</code> مثل) السجل وتحويلها لنظام الفوترة.

استكشاف الأخطاء وإصلاحها

عند التحديث أو الإفراج 404

مع أي جلسة نشطة في الذاكرة. الأسباب `chargingDataRef` لا يتطابق

- بين الإنشاء والتحديث/الإفراج - جميع حالات `OmniCHF` جلسات تم إعادة تشغيل في الذاكرة وتفقد عند إعادة التشغيل.
- غير صحيح في المسار `chargingDataRef` SMF أرسل.
- تم إرسال إفراج سابق عن هذه الجلسة، مما أدى إلى حذف السياق.

`CHF Release: unknown ref=` أو `CHF Update: unknown ref=` تحقق من السجلات لـ للتأكيد.

500 CGRateS عند الإنشاء مع تمكين

تحقق من `CGRateS`. فشل استدعاء

1. يمكن الوصول إليه؟ CGRateS يشير إلى مثل `cgrates_url` هل.
2. صحيح؟ تسبب أسماء المستأجر غير المتطابقة في عودة `cgrates_tenant` هل. باستجابة خطأ CGRateS.
3. (متصل، 0=غير متصل=1) `omni_chf.cgrates.health` تحقق من مقياس.
4. `CGRateS InitiateSession failed for <ref>` راجع السجلات لـ `<reason>`، `{:cgrates_error, message}`: ستكون السبب واحدًا من - `{:http_error, status}`، أو `{:http_error, reason}`.
5. CGRateS مفعلة في تكوين `SessionSv1` CGRateS تحقق من أن خدمة.

ولكن الخدمة تعمل CGRateS فشل التحقق من صحة

مهلة محددة بـ 3 ثوانٍ. (`SessionSv1.GetActiveSessions` عبر) يستخدم التحقق من الصحة بطيئة في الاستجابة، فقد يفشل التحقق من الصحة بينما تكون الخدمة متاحة CGRateS إذا كانت أيضًا تأكد. `min(cgrates_timeout, 3000)` الحد هو - `cgrates_timeout` تقنيًا. تحقق من TLS. ما لم يتم تكوين (HTTPS ليس) HTTP يستخدم `cgrates_url` من أن

أو عدم اكتمالها CDRs عدم ظهور

CDRs لانتقاط. (CHF-L5 انظر القيود) INFO إلى سجل التطبيق على مستوى CDRs يتم كتابة

1. أو أقل `info` تأكد من أن مستوى سجل التطبيق مضبوط على.
2. للمعالجة اللاحقة "CHF CDR:" قم بتصفية خطوط السجل التي تحتوي على.
3. (`chargingID`، `recordingEntity`) (CHF-L4)، تفتقر إلى CDRs لاحظ أن (`ratingGroup: 1`) (CHF-L2) لجميع تدفقات بيانات الخدمة وستحتوي على.

CGRateS ارتفاع زمن استجابة استدعاءات

يزداد الزمن تحت (CHF-M5) بدون تجميع الاتصالات `httpc`: يستخدم CGRateS نظرًا لأن الحمل. لتشخيص

1. لزمن الاستجابة `omni_chf.cgrates.request.duration_ms` تحقق من مخطط. في النسبة المئوية 99.
2. إذا كان الزمن مرتفعًا تحت الحمل المتزامن، قلل من عدد جلسات الشحن المتزامنة. خلف موازن تحميل مع عدة مثيلات OmniCHF أو اعتبر نشر.
3. إلى قيمة أقل من الزمن المتوقع الأسوأ `cgrates_timeout` كحل مؤقت، قم بتعيين البطيئة من حظر مجموعة عمليات CGRateS لمنع استدعاءات CGRateS لاستجابة

Elixir.

عدم انخفاض عدد الجلسات النشطة بعد الإفراج

مرتفعًا بعد أن كان يجب الإفراج عن الجلسات `omni_chf.sessions.active.count` إذا ظل

1. فقد لا يعيد، SMF 404 تحقق من استجابات 404 على استدعاءات الإفراج - إذا تلقى.
2. السياق OmniCHF الجلسة قد أفرج عنها بينما قد لا يزال لدى SMF المحاولة ويعتبر وفقد حالة الجلسة، فإن السياقات قد OmniCHF في الحالة العكسية، إذا أعيد تشغيل. يرسل طلبات تحديث/إفراج تؤدي إلى 404. هذا سلوك SMF اختفت ولكن قد لا يزال متوقع.

NRF عدم الحفاظ على تسجيل

إذا كان يقرأ 0. `omni_chf.nrf.registration.status` تحقق من مقياس

1. لـ `sbi_addr` يمكن الوصول إليه من NRF صحيح وأن `nrf_uri` تحقق من أن OmniCHF.
2. NRF الخاص بـ PLMN يتطابقان مع تكوين `mnc` و `mcc` تحقق من أن.
3. NRF راجع سجلات التطبيق عند بدء التشغيل للبحث عن أخطاء تسجيل.