دليل تشغيل OmniUPF

جدول المحتويات

1. نظرة عامة
2. فهم بنية مستوى المستخدم في 5G
3. مكونات UPF
4. بروتوكول PFCP وتكامل SMF
5. العمليات الشائعة
6. استكشاف الأخطاء وإصلاحها
7. وثائق إضافية
8. معجم

نظرة عامة

هي وظيفة مستوى مستخدم (eBPF وظيفة مستوى المستخدم المعتمدة على OmniUPF) 5G/LTE عالية الأداء توفر توجيه حزم على مستوى الناقل، وإنفاذ جودة الخدمة (QoS)، وإدارة مرشح حزم بيركلي (Linux eBPF حركة المرور لشبكات الهاتف المحمول. مبنية على تقنية البنية التحتية الأساسية لمعالجة الحزم OmniUPF ومعززة بقدرات إدارة شاملة، توفر (الموسع 5 المطلوبة لشبكاتIG SA 5وG NSA وLTE.

ما هي وظيفة مستوى المستخدم؟

المسؤول عن معالجة GPPهي عنصر الشبكة المعتمد من 3 وظيفة مستوى المستخدم (UPF) 5 الحزم وتوجيهها في شبكاتG وLTE. توفر:

- **توجيه حزم عالي السرعة** بين الأجهزة المحمولة وشبكات البيانات
- **إنفاذ جودة الخدمة (QoS)** لأنواع حركة المرور المختلفة
- **اكتشاف حركة المرور وتوجيهها** بناءً على مرشحات الحزم والقواعد
- **تقرير الاستخدام** للفوترة والتحليلات
- **تخزين الحزم** لسيناريوهات إدارة الحركة والجلسات
- **دعم الاعتراض القانوني** للامتثال التنظيمي

تقوم OmniUPF بتنفيذ الوظائف الكاملة لـ UPF 3 كما هو محدد في 3GPP TS 23.501 (5G) و3GPP TS 23.401 (LTE)، مما يوفر حلاً كاملاً جاهزًا للإنتاج لوظيفة مستوى المستخدم باستخدام تقنية Linux kernel eBPF لتحقيق أقصى أداء.

القدرات الرئيسية لـ OmniUPF

معالجة الحزم:

- معالجة حزم مستوى المستخدم متوافقة بالكامل مع 3GPP
- مسار بيانات معتمد على eBPF لأداء على مستوى النواة
- تغليف وفك تغليف GTP-U (بروتوكول نفق GPRS)
- دعم IPv4 وIPv6 لكل من الشبكات الوصول وبيانات
- XDP (مسار البيانات السريع) لمعالجة ذات زمن تأخير منخفض للغاية
- معالجة حزم متعددة الخيوط

QoS وإدارة حركة المرور:

- قواعد إنفاذ QoS (QER) لإدارة النطاق الترددي
- قواعد اكتشاف الحزم (PDR) لتصنيف حركة المرور
- قواعد إجراء التوجيه (FAR) لقرارات التوجيه
- تصفية تدفق بيانات الخدمة (SDF) للتوجيه الخاص بالتطبيق
- قواعد تقرير الاستخدام (URR) لتتبع الحجم والفوترة

التحكم والإدارة:

- واجهة PFCP (بروتوكول التحكم في توجيه الحزم) إلى SMF/PGW-C
- واجهة API RESTful للمراقبة والتشخيص
- إحصائيات ومقاييس في الوقت الحقيقي
- مراقبة سعة خريطة eBPF
- لوحة تحكم قائمة على الويب

ميزات الأداء:

- معالجة حزم بدون نسخ عبر eBPF
- توجيه حزم على مستوى النواة (بدون عبء على مساحة المستخدم)
- قابلية التوسع متعددة النوى

- دعم التحميل لتسريع الأجهزة
- مُحسّن للنشر السحابي

للحصول على تفاصيل استخدام لوحة التحكم، انظر عمليات واجهة الويب.

فهم بنية مستوى المستخدم

تعد OmniUPF 5G حلاً موحدًا لمستوى المستخدم يوفر توجيه حزم على مستوى الناقل لشبكات 4G LTE/EPC و5G NSA و5G المستقلة (SA). **OmniUPF هو منتج واحد** يمكن أن يعمل في نفس الوقت كـ:

- **UPF (وظيفة مستوى المستخدم)** - 5G/NSA مستوى مستخدم (تحت السيطرة من OmniSMF عبر N4/PFCP)
- **PGW-U (بوابة بيانات المستخدم)** - بوابة EPC 4G إلى الشبكات الخارجية (تحت السيطرة من OmniPGW-C عبر Sxc/PFCP)
- **SGW-U (بوابة الخدمة)** - بوابة الخدمة EPC 4G (من تحت السيطرة OmniSGW-C عبر Sxb/PFCP)

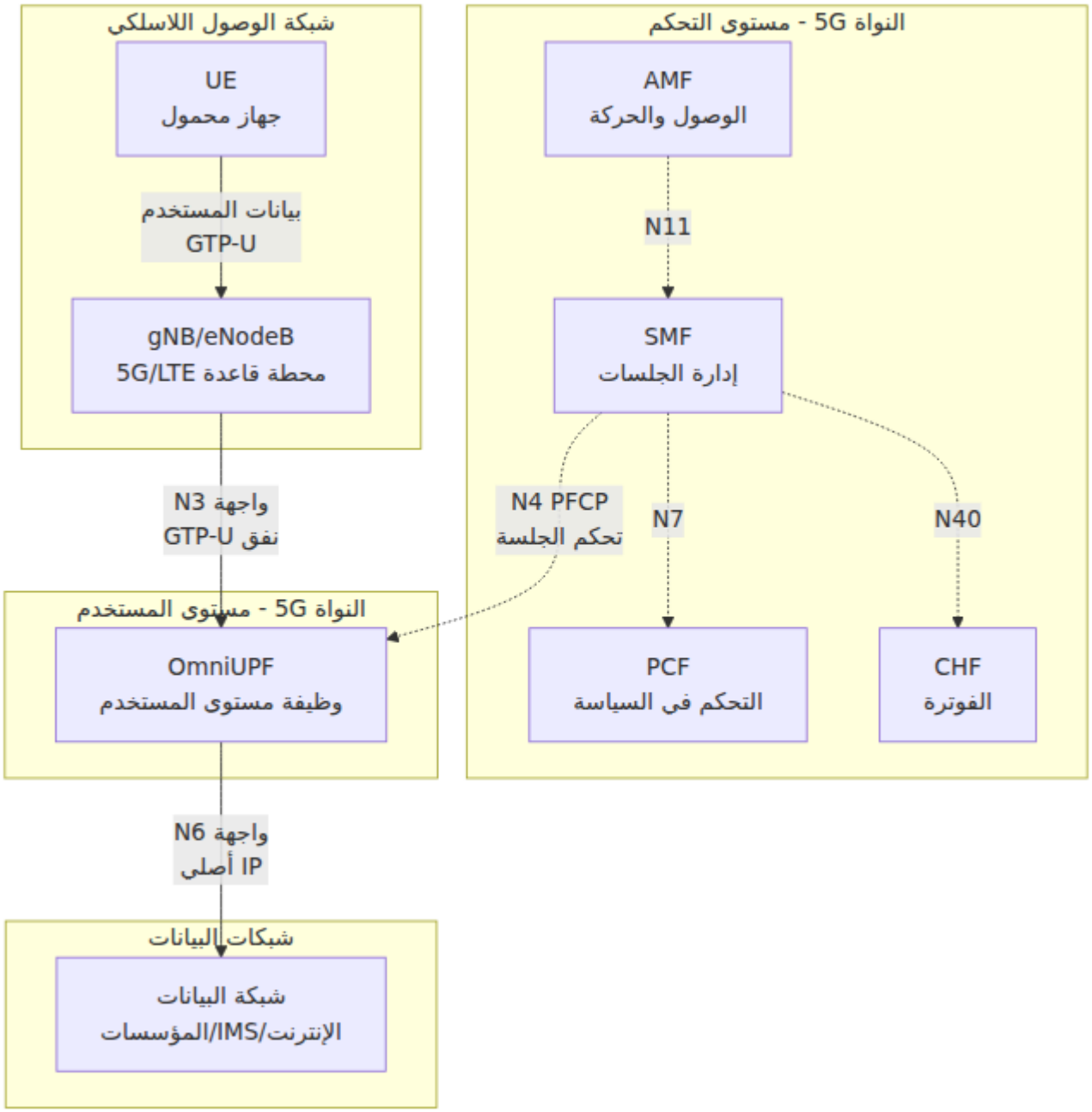
يمكن أن تعمل OmniUPF من هذه الأوضاع **أي مجموعة** في:

- **UPF فقط**: نشر 5G خالص
- **PGW-U + SGW-U**: 4G بوابة مجمعة (نشر EPC نموذجي)
- **UPF + PGW-U + SGW-U**: 4G و5G دعم (سيناريو الهجرة في نفس الوقت)

تستخدم جميع الأوضاع نفس محرك معالجة الحزم المعتمد على eBPF وبروتوكول PFCP، مما يوفر أداءً عاليًا متسقًا سواء كانت تعمل كـ UPF أو PGW-U أو SGW-U أو الثلاثة معًا.

بنية شبكة 5G (وضع SA)

تجلس حل OmniUPF 5G في مستوى البيانات لشبكات 5G، موفرةً طبقة توجيه حزم عالية السرعة تربط الأجهزة المحمولة بشبكات البيانات والخدمات.



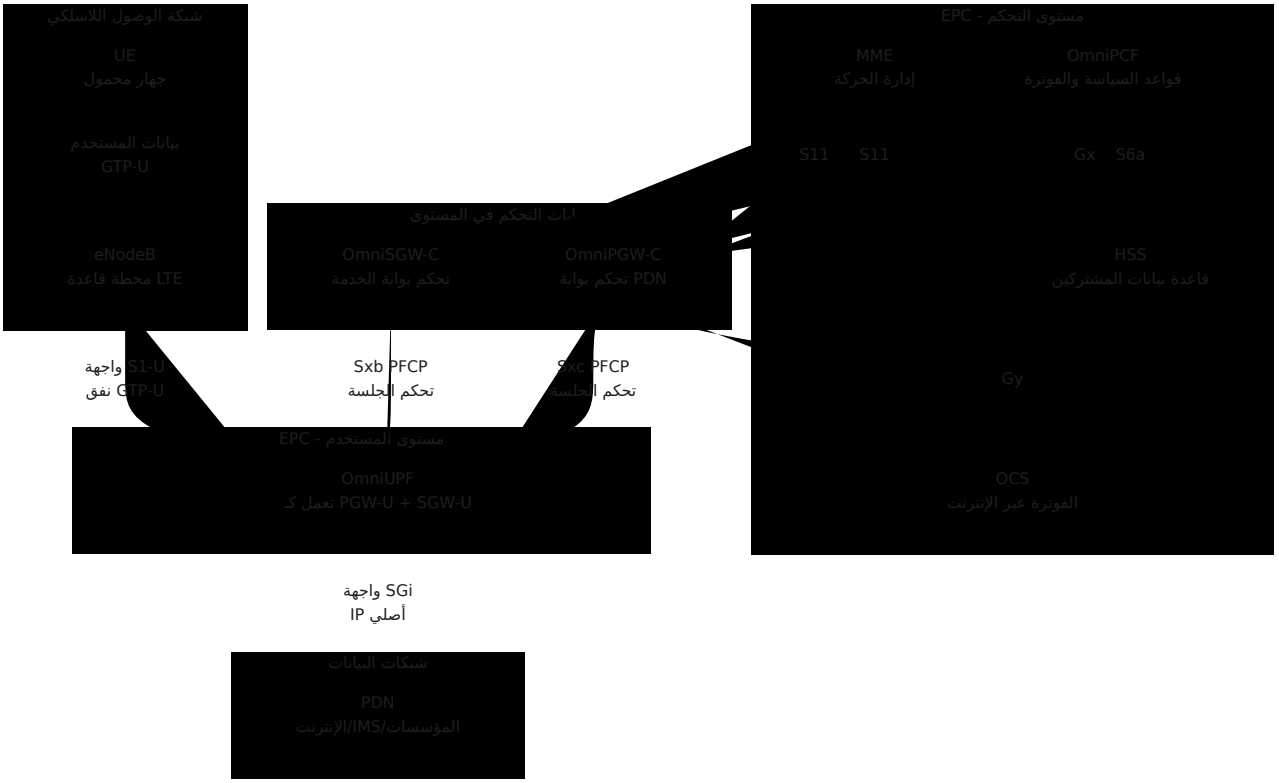
---

4 بنية شبكةG LTE/EPC

تدعم OmniUPF 4 أيضًا نشرات G LTE وEPC (نواة الحزمة المتطورة)، تعمل كـ OmniPGW-U أو OmniSGW-U اعتمادًا على بنية الشبكة.

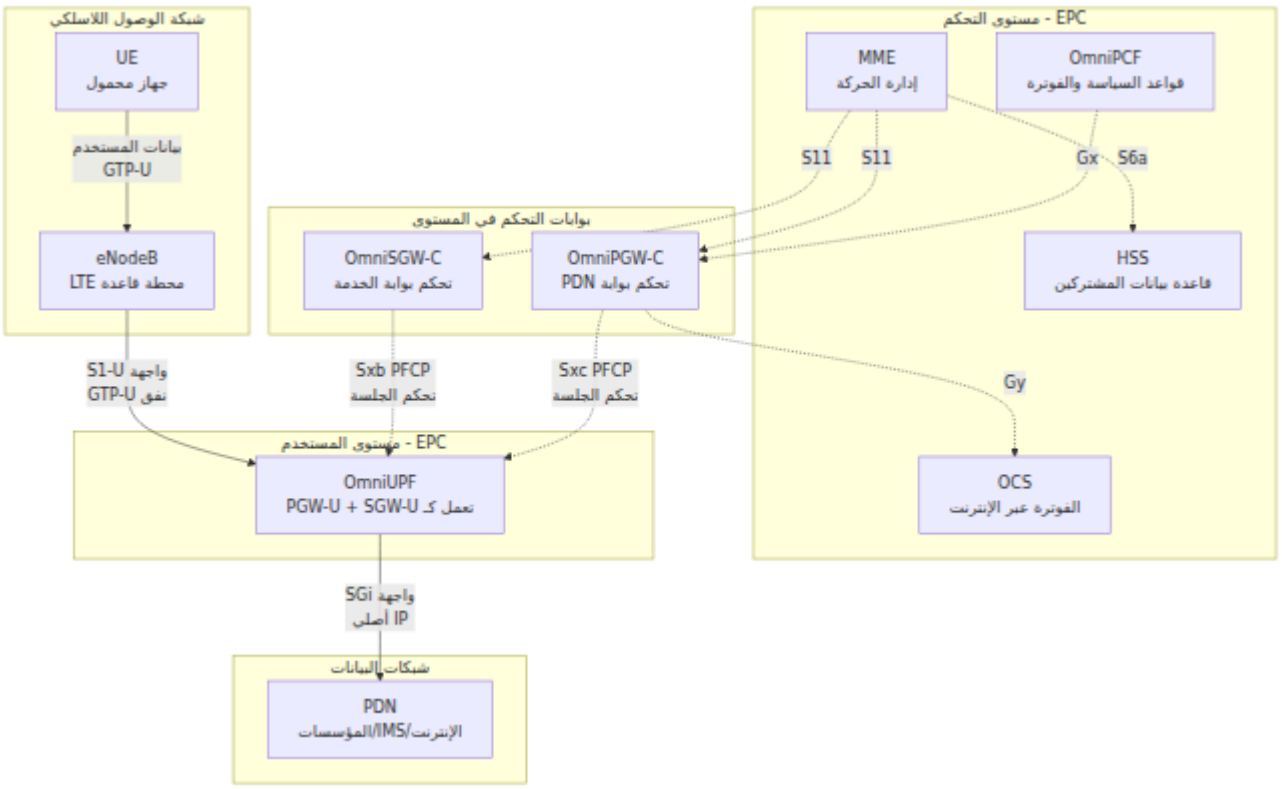
وضع PGW-U/SGW-U المدمج (نموذجي G نشر 4)

في هذا الوضع، تعمل OmniUPF كلاً من SGW-U و PGW-U، تحت السيطرة من وظائف التحكم في مستوى مختلف.



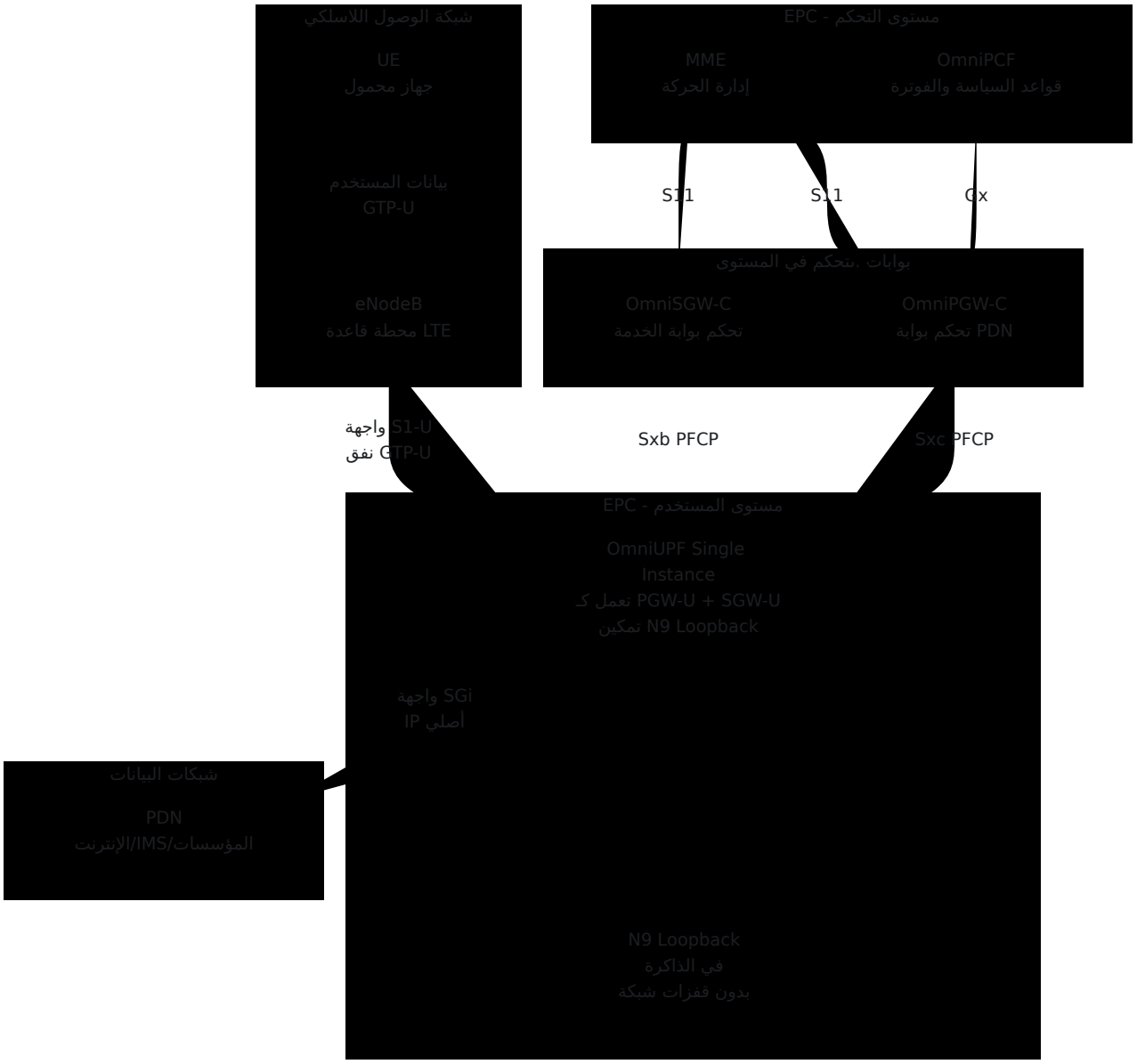
وضع SGW-U و PGW-U (التجوال/مواقع متعددة) المنفصل

في نشرات التجوال أو المواقع المتعددة، يمكن نشر حالتين منفصلتين من OmniUPF - واحدة كـ SGW-U وواحدة كـ PGW-U.



وضع N9 Loopback (حالة SGWU+PGWU واحدة)

تشغيل **كل من أدوار** OmniUPF للنشر المبسط، يمكن لـ **على حالة PGWU و SGWU** بالكامل في eBPF. N9 loopback **واحدة** مع معالجة



الميزات الرئيسية:

- **أقل من ميكروثانية** - معالجة بالكامل في eBPF، لا تلمس **زمن تأخير N9** الشبكة
- **تخفيض CPU** - تمريرة واحدة عبر XDP **بنسبة 40-50%** مقابل حالتين منفصلتين
- **نشر مبسط** - حالة واحدة، ملف تكوين واحد
- **اكتشاف تلقائي** - عندما يكون `n3_address` = `n9_address`، يتم تمكين loopback
- **امتثال كامل لـ 3GPP** - بروتوكولات PFCP و GTP-U القياسية

التكوين:

```
# /etc/omniupf/runtime.exs
xdp_interfaces = "eth0"
n3_address = "10.0.1.10"        # عنوان IP لواجهة S1-U
n9_address = n3_address          # نفس عنوان IP يمكّن N9 loopback
pfcp_address = "10.0.1.10"      # كل من PGWU-C و SGWU-C تتصل هنا
pfcp_port = 8805
```

متى تستخدم:

- نشرات الحوسبة الطرفية (تقليل زمن التأخير)
- بيئات مقيدة من حيث التكلفة (خادم واحد)
- المختبر/الاختبار (إعداد مبسط)
- نشرات صغيرة إلى متوسطة (< 100K مشتركين)

متى لا تستخدم:

- الحاجة إلى تكرار جغرافي (في مواقع مختلفة PGWU و SGWU)
- متطلبات تنظيمية لفصل البوابات
- نطاق ضخم (> 1M مشتركين)

للحصول على تفاصيل كاملة، وأمثلة على التكوين، واستكشاف الأخطاء وإصلاحها، ومقاييس الأداء، انظر **دليل عمليات N9 Loopback**.

كيف تعمل وظائف مستوى المستخدم في الشبكة

تعمل وظيفة مستوى المستخدم (OmniUPF، OmniPGW-U، أو OmniSGW-U) كطائرة توجيه تحت السيطرة من الطائرة التحكم المناسبة:

1. **إنشاء الجلسة**
    - **5G**: يقوم OmniSMF بإنشاء ارتباط PFCP عبر واجهة N4 مع OmniUPF
    - **4G**: يقوم OmniPGW-C أو OmniSGW-C بإنشاء ارتباط PFCP عبر Sxb/Sxc مع OmniPGW-U/OmniSGW-U
    - تنشئ الطائرة التحكم جلسات PFCP لكل جلسة PDU UE (5G) أو سياق PDP (4G)

- تتلقى الطائرة المستخدم PDR و FAR و QER و URR عبر PFCP
- يتم ملء خرائط eBPF بقواعد التوجيه

2. **معالجة حزم الرفع** (UE → شبكة البيانات)

- **5G**: تصل الحزم على واجهة N3 من gNB مع تغليف GTP-U
- **4G**: تصل الحزم على واجهة S1-U (SGW-U) أو واجهة S5/S8 (PGW-U) من eNodeB مع تغليف GTP-U
- الرفع بناءً على TEID تطابق الطائرة المستخدم الحزم مع PDRs
- يطبق برنامج eBPF QER (تحديد المعدل، العلامة)
- تحدد FAR (التوجيه، الإسقاط، التخزين، التكرار) إجراء التوجيه
- تتم إزالة نفق GTP-U، ويتم توجيه الحزم إلى واجهة N6 (5G) أو SGi (4G)
- تتبع URR عدد الحزم والبايتات للفوترة

3. **معالجة حزم النزول** (شبكة البيانات → UE)

- **5G**: تصل الحزم على واجهة N6 كـ IP أصلي
- **4G**: تصل الحزم على واجهة SGi كـ IP أصلي
- تطابق الطائرة المستخدم الحزم مع PDRs النزول بناءً على عنوان IP الخاص بـ UE
- قد تصنف مرشحات SDF أو حركة المرور بشكل إضافي حسب المنفذ البروتوكول أو التطبيق
- تحدد FAR نفق GTP-U ومعلمات التوجيه
- تتم إضافة تغليف GTP-U مع TEID المناسب
- **5G**: يتم توجيه الحزم إلى واجهة N3 نحو gNB
- **4G**: يتم توجيه الحزم إلى S1-U (SGW-U) أو S5/S8 (PGW-U) نحو eNodeB

4. **الحركة والتبديل**

- **5G**: يقوم OmniSMF بتحديث قواعد PDR/FAR خلال سيناريوهات التبديل
- **4G**: يقوم OmniSGW-C/OmniPGW-C بتحديث الق��اعد خلال التب��يل بين eNodeB أو TAU (تحديث منطقة التتبع)
- قد تخزن الطائرة المستخدم الحزم خلال تبديل المسار
- انتقال سلس بين محطات القاعدة دون فقدان الحزم

التكامل مع الطائرة التحكم (4G و5G)

يتكامل OmniUPF 5 عبر واجهات 3GPP القياسية: 3G و4G مع كل من وظائف الطائرة التحكم

واجهات 5G

الواجهة	من → إلى	الغرض	مواصفة 3GPP
N4	OmniSMF ↔ OmniUPF	إنشاء، تعديل، حذف جلسة PFCP	TS 29.244
N3	gNB → OmniUPF	حركة مرور مستوى المستخدم من RAN (GTP-U)	TS 29.281
N6	OmniUPF → شبكة البيانات	حركة مرور مستوى المستخدم إلى DN (IP أصلي)	TS 23.501
N9	OmniUPF ↔ OmniUPF	الاتصال بين UPF للتجوال/الحافة	TS 23.501

واجهات 4G/EPC

الواجهة	من → إلى	الغرض	مواصفة 3GPP
Sxb	OmniSGW-C ↔ OmniUPF (وضع SGW-U)	التحكم في جلسة PFCP للبوابة الخدمية	TS 29.244
Sxc	OmniPGW-C ↔ OmniUPF (وضع PGW-U)	لتحكم في جلسة PFCP للبوابة PDN	TS 29.244
S1-U	eNodeB → OmniUPF (وضع SGW-U)	حركة مرور مستوى المستخدم من RAN (GTP-U)	TS 29.281
S5/S8	OmniUPF (SGW-U) ↔ OmniUPF (PGW-U)	مستوى المستخدم بين البوابات (GTP-U)	TS 29.281
SGi	OmniUPF (وضع PGW-U) → PDN	حركة مرور مستوى المستخدم إلى شبكة البيانات (IP أصلي)	TS 23.401

ملاحظة: جميع واجهات PFCP (N4، Sxb، Sxc) تستخدم نفس بروتوكول PFCP المحدد في TS 29.244. تختلف أسماء الواجهات ولكن البروتوكول وتنسيقات الرسائل متطابقة.

مكونات UPF

مسار بيانات eBPF

مسار بيانات eBPF هو محرك معالجة الحزم الأساسي الذي يعمل في نواة Linux لتحقيق أقصى أداء.

الوظائف الأساسية:

- **معالجة GTP-U**: تغليف وفك تغليف أنفاق GTP-U
- **تصنيف الحزم**: مطابقة الحزم مع قواعد PDR باستخدام TEID، عنوان IP الخاص بـ UE، أو مرشحات SDF
- **إنفاذ QoS**: تطبيق تحديد المعدل وعلامة الحزم وفقًا لقواعد QER
- **قرارات التوجيه**: تنفيذ إجراءات FAR (التوجيه، الإسقاط، التخزين، التكرار، الإخطار)
- **تتبع الاستخدام**: زيادة عدادات URR للفوترة المعتمدة على الحجم

خرائط eBPF: يستخدم مسار البيانات خرائط eBPF (جداول التجزئة في ذاكرة النواة) لتخزين القواعد:

اسم الخريطة	الغرض	المفتاح	القيمة
`uplink_pdr_map`	PDRs الرفع	TEID (32 بت)	معلومات PDR (FAR ID، QER ID، URR IDs)
`downlink_pdr_map`	PDRs النزول (IPv4)	عنوان IP الخاص بـ UE	معلومات PDR
`downlink_pdr_map_ip6`	PDRs النزول (IPv6)	عنوان IPv6 الخاص بـ UE	معلومات PDR
`far_map`	قواعد التوجيه	FAR ID	معلمات التوجيه (الإجراء، معلومات النفق)
`qer_map`	قواعد QoS	QER ID	معلمات QoS (MBR، GBR، العلامة)
`urr_map`	تتبع الاستخدام	URR ID	عدادات الحجم (الرفع، النزول، الإجمالي)
`sdf_filter_map`	مرشحات SDF	PDR ID	مرشحات التطبيقات (المنافذ، البروتوكولات)

خصائص الأداء:

- **بدون نسخ**: تتم معالجة الحزم بالكامل في مساحة النواة
- **دعم XDP**: التعلق على مستوى برنامج تشغيل الشبكة لتحقيق زمن تأخير أقل من ميكروثانية
- **متعددة النوى**: تتوسع عبر نوى CPU مع دعم خريطة لكل CPU
- **السعة**: ملايين من PDRs/FARs في خرائط eBPF (محدودة بذاكرة النواة)

للحصول على مراقبة السعة، انظر إدارة السعة.

معالج واجهة PFCP

ت implements واجهة **PFCP** 3GPP TS 29.244 للتواصل مع SMF أو PGW-C.

الوظائف الأساسية:

- **إدارة الارتباط**: نبض وإعداد/إصدار الارتباط PFCP
- **دورة حياة الجلسة**: إنشاء، تعديل، وحذف جلسات PFCP
- **تثبيت القواعد**: ترجمة عناصر PFCP إلى إدخالات خريطة eBPF
- **تقرير الأحداث**: إبلاغ SMF بعوامل الاستخدام، الأخطاء، أو أحداث الجلسة

دعم رسائل PFCP:

نوع الرسالة	الاتجاه	الغرض
إعداد الارتباط	SMF → UPF	إنشاء ارتباط التحكم PFCP
إصدار الارتباط	SMF → UPF	إنهاء ارتباط PFCP
نبض	ثنائي الاتجاه	الحفاظ على الارتباط نشطًا
إنشاء الجلسة	SMF → UPF	إنشاء جلسة PDU جديدة مع PDR/FAR/QER/URR
تعديل الجلسة	SMF → UPF	تحديث القواعد للحركة، تغييرات QoS
حذف الجلسة	SMF → UPF	إزالة الجلسة وجميع القواعد المرتبطة بها
تقرير الجلسة	UPF → SMF	تقرير الاستخدام، الأخطاء، أو الأحداث

عناصر المعلومات (IE) المدعومة:

- إنشاء PDR، FAR، QER، URR
- تحديث PDR، FAR، QER، URR
- إزالة PDR، FAR، QER، URR
- معلومات اكتشاف الحزم (مرشح SDF، F-TEID، UE الخاص بـ IP عنوان)
- معلمات التوجيه (مثيل الشبكة، إنشاء رأس خارجي)
- معلمات QoS (MBR، GBR، QFI)

- مشغلات تقرير الاستخدام (عتبة الحجم، عتبة الوقت)

خادم API REST

يوفر **API REST** وصولًا برمجيًا إلى حالة UPF وعملياته.

الوظائف الأساسية:

- **مراقبة الجلسات**: استعلام عن جلسات PFCP النشطة والارتباطات
- **فحص القواعد**: عرض تكوينات PDR، FAR، QER، URR
- **الإحصائيات**: استرداد عدادات الحزم، إحصائيات المسار، إحصائيات XDP
- **إدارة التخزين المؤقت**: عرض والتحكم في تخزين الحزم
- **معلومات الخريطة**: مراقبة استخدام خريطة eBPF والسعة

نقاط نهاية API: (نقطة نهاية إجمالية 34)

الفئة	نقاط النهاية	الوصف
الصحة	`/health`	فحص الصحة والحالة
التكوين	`/config`	تكوين UPF
الجلسات	`/pfcp_sessions`, `/pfcp_associations`	بيانات جلسة/ارتباط PFCP
PDRs	`/uplink_pdr_map`, `/downlink_pdr_map`, `/downlink_pdr_map_ip6`, `/uplink_pdr_map_ip6`	قواعد اكتشاف الحزم
FARs	`/far_map`	قواعد إجراء التوجيه
QERs	`/qer_map`	قواعد إنفاذ QoS
URRs	`/urr_map`	قواعد تقرير الاستخدام
Buffers	`/buffer`	حالة التخزين المؤقت للحزم والتحكم
الإحصائيات	`/packet_stats`, `/route_stats`, `/xdp_stats`, `/n3n6_stats`	مقاييس الأداء
السعة	`/map_info`	سعة واستخدام خريطة eBPF
Dataplane	`/dataplane_config`	عناوين واجهات N3/N9

للحصول على تفاصيل API واستخدامه، انظر **دليل المراقبة**.

لوحة التحكم على الويب

توفر UPF **لوحة التحكم على الويب** لوحة معلومات في الوقت الحقيقي لمراقبة وإدارة.

الميزات:

- **عرض الجلسات**: تصفح جلسات PFCP النشطة مع عنوان IP الخاص بـ UE، TEID، وعدد القواعد
- **إدارة القواعد**: عرض وإدارة PDRs، FARs، QERs، وURRs عبر جميع الجلسات
- **مراقبة التخزين المؤقت**: تتبع الحزم المخزنة والتحكم في التخزين المؤقت لكل FAR
- **لوحة إحصائيات**: إحصائيات الحزم، المسار، XDP، وإحصائيات واجهة في N3/N6 الوقت الحقيقي
- **مراقبة السعة**: استخدام خريطة eBPF مع مؤشرات سعة ملونة
- **عرض التكوين**: عرض تكوين UPF وعناوين dataplane
- **عارض السجلات**: بث السجلات الحية لاستكشاف الأخطاء وإصلاحها

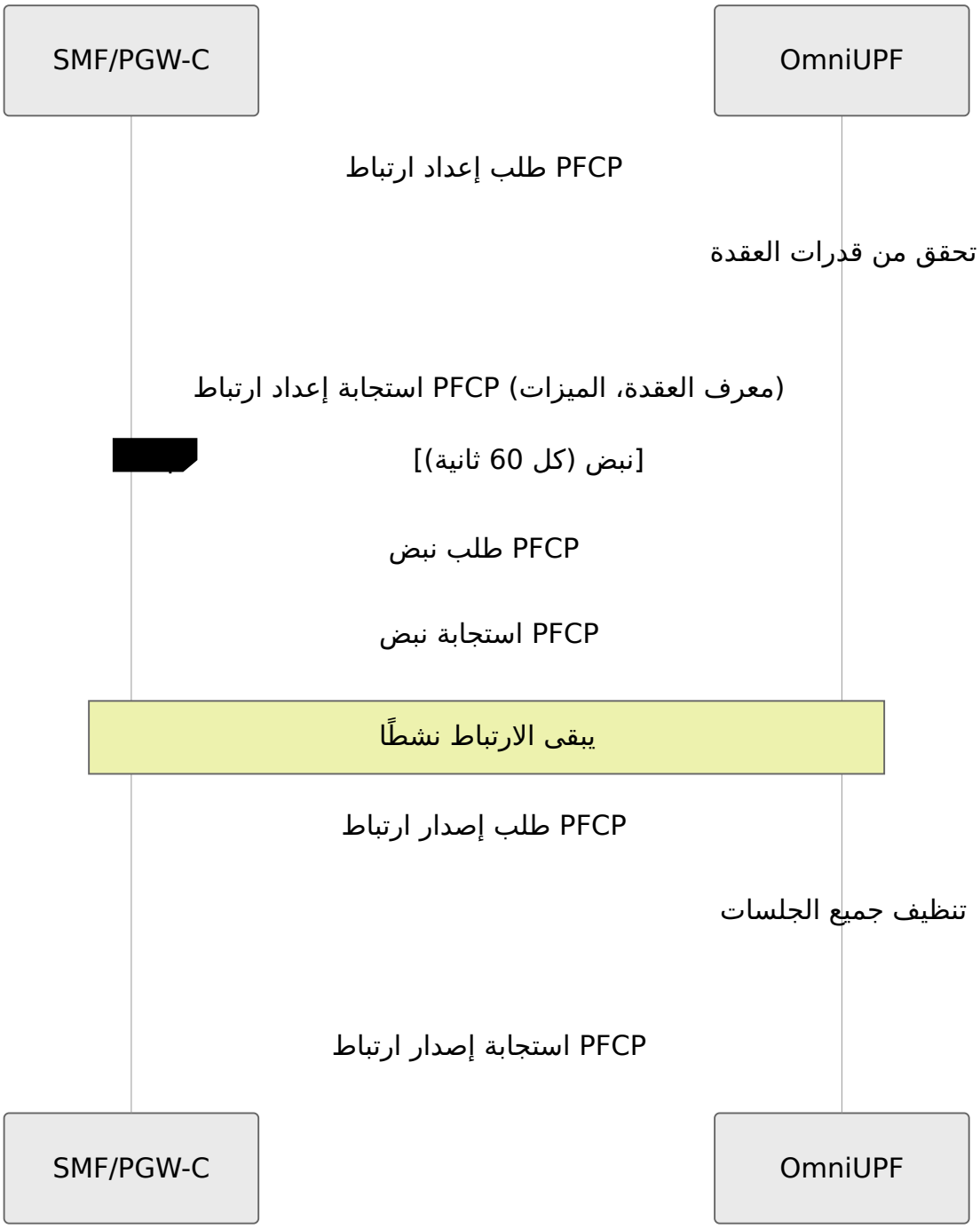
للحصول على عمليات واجهة المستخدم التفصيلية، انظر **دليل عمليات واجهة الويب**.

SMF وتكامل بروتوكول PFCP

ارتباط PFCP

قبل إنشاء الجلسات، يجب على SMF إنشاء ارتباط PFCP مع UPF.

دورة حياة الارتباط:



النقاط الرئيسية:

- يقوم كل SMF بإنشاء ارتباط واحد مع UPF
- يتتبع UPF الارتباط بواسطة معرف العقدة (عنوان IP أو FQDN)
- تحافظ رسائل النبض على نشاط الارتباط
- يتم حذف جميع الجلسات تحت ارتباط إذا تم إصدار الارتباط

لرؤية الارتباطات، انظر عرض الجلسات.

وتنظيف الجلسات اليتيمة SMF اكتشاف إعادة تشغيل

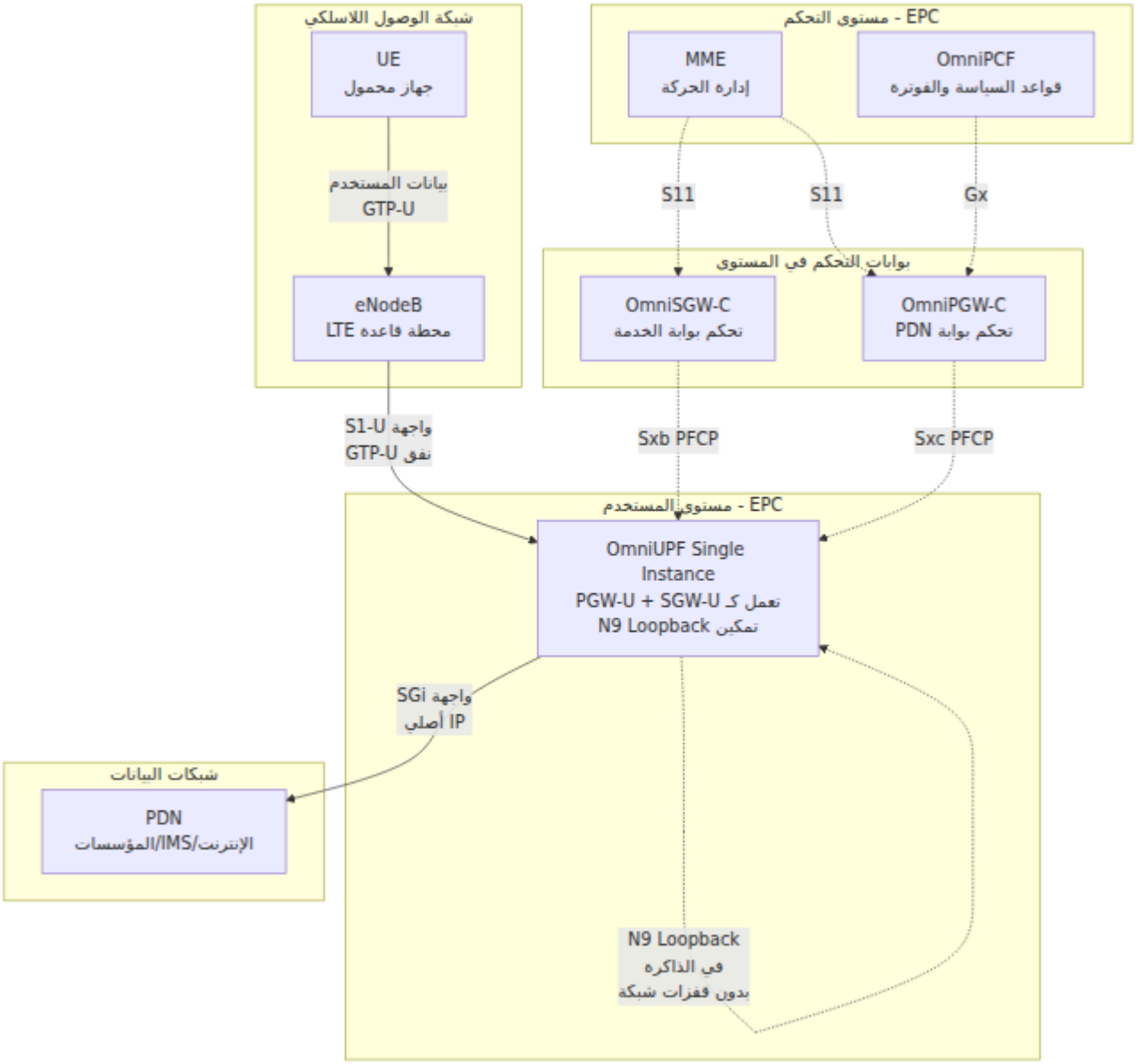
تكتشف OmniUPF تلقائيًا عندما يتم إعادة تشغيل SMF وتقوم بتنظيف الجلسات اليتيمة وفقًا لمواصفات 3GPP TS 29.244.

كيف يعمل:

عندما يقوم SMF بإنشاء ارتباط PFCP، يشير إلى متى بدأ. تخزن SMF فإنه يوفر **ختم الاسترداد** هذا الختم لكل ارتباط. إذا أعيد تشغيل OmniUPF:

1. يفقد SMF جميع حالة الجلسة في الذاكرة
2. يعيد SMF إنشاء ارتباط PFCP مع UPF
3. يرسل SMF (مختلف عن السابق) **ختم استرداد جديد**
4. تكتشف UPF تغيير الختم = تم إعادة تشغيل SMF
5. تحذف UPF من حالة SMF القديمة تلقائيًا **جميع الجلسات اليتيمة**
6. ينشئ SMF جلسات جديدة للمشتركين النشطين

تدفق اكتشاف إعادة التشغيل:



مثال السجل:

عندما يتم إعادة تشغيل SMF، سترى:

```
WARN: الارتباط مع NodeID: smf-1 موجود بالفعل 192.168.1.10 :والعنوان
WARN: تغير ختم استرداد SMF (15-01-2025T10:00:00Z :القديم، الجديد:
15-01-2025T10:30:15Z) - تم إعادة تشغيل SMF، حذف 245 جلسة يتيمة
INFO: 2 حذف الجلسة اليتيمة (LocalSEID) بسبب إعادة تشغيل SMF
INFO: 3 حذف الجلسة اليتيمة (LocalSEID) بسبب إعادة تشغيل SMF
...
INFO: 246 حذف الجلسة اليتيمة (LocalSEID) بسبب إعادة تشغيل SMF
```

ملاحظات هامة:

1. **العزل**: يتم حذف جلسات SMF المعاد تشغيلها فقط. لا تتأثر ارتباطات SMF الأخرى وجلساتها **بأي شكل**.

2. **مقارنة الختم**: إذا كان ختم الاسترداد **مطابقًا**، يتم **الاحتفاظ** بالجلسات (تم إعادة الاتصال بـ SMF دون إعادة التشغيل).

3. **امتثال 3GPP**: هذا السلوك مطلوب بموجب 3GPP TS 29.244 5.22.2 القسم:

> "منذ آخر إعداد ارتباط، يجب أن تعتبر CP إذا تغير ختم وقت الاسترداد لوظيفة PFCP قد أعيد تشغيلها ويجب أن تحذف جميع جلسات CP أن وظيفة UP المرتبطة بتلك الوظيفة CP."

للاستكشاف الأخطاء في الجلسات اليتيمة، انظر اكتشاف الجلسات اليتيمة.

معالجة إشارات خطأ GTP-U

تتعامل OmniUPF مع رسائل إشارات خطأ GTP-U من الأقران السفليين (PGW-U، SGW-U، eNodeB، gNodeB) وفقًا لمواصفات 3GPP TS 29.281.

ما هي إشارات الخطأ:

عندما تقوم OmniUPF بتوجيه حزمة GTP-U إلى نظير بعيد (على سبيل المثال، PGW-U في نشر SGW-U)، وهذا قد يرسل النظير **إشارة خطأ** إذا لم يتعرف على TEID (معرف نقطة النفق). يشير إلى:

- أن النظير البعيد قد أعيد تشغيله وفقد حالة النفق
- أن النفق لم يتم إنشاؤه أبدًا على الجانب البعيد (عدم تطابق التكوين)
- أن النفق قد تم حذفه بالفعل على الجانب البعيد

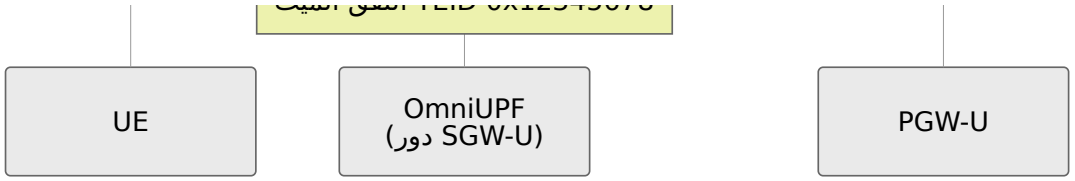
كيف يعمل:

1. **UPF** ترسل حزمة → **توجيه الحزمة** GTP-U مع TEID X إلى النظير البعيد (المنفذ 2152)
2. **النظير البعيد لا يتعرف على TEID X** → يبحث عن TEID في جدول الأنفاق الخاص به، غير موجود
3. رسالة → **النظير البعيد يرسل إشارة خطأ** GTP-U مع 26 من النوع IE تحتوي على TEID الخاطئ
4. **UPF** تقوم بتحليل الرسالة لاستخراج → **تتلقى إشارة خ��أ** TEID X

5. **UPF** **تجد الجلسات المتأثرة** → تبحث عن جميع الجلسات للـ FARs التي توجه إلى TEID X
6. **UPF** **تحذف الجلسات** → تزيل الجلسات من خرائط eBPF وحالة PFCP
7. **UPF** **تحديث المقاييس** → تزيد عدادات Prometheus للمراقبة

تدفق إشارة الخطأ:

UE
OmniUPF
(دور SGW-U)
PGW-U
لديها نفق نشط PGW-U
TEID: 0x12345678
حزمة بيانات الرفع
GTP-U G-PDU
(TEID: 0x12345678)
يتم التوجيه بشكل طبيعي
⚠ أعيد تشغيله PGW-U
!يفقد كل حالة النفق
حزمة بيانات الرفع
GTP-U G-PDU
(TEID: 0x12345678)
البحث عن TEID 0x12345678
غير موجود
⚠ إشارة خطأ GTP-U
(TEID 0 :الخاطئx12345678)
تحليل إشارة الخطأ
استخراج TEID: 0x12345678
العثور على الجلسات مع FAR
توج�� إلى TEID 0x12345678
حذف الجلسات المتأثرة
- إزالة من خرائط eBPF
- مسح حالة PFCP
تحديث المقاييس:
error_indications_received++
sessions_deleted++
توقف عن التوجيه إلى
النفق الميت TEID 0x12345678



تنسيق الحزمة (3GPP TS 29.281 القسم 7.3.1):

```
إشارة خطأ GTP-U:
┌──────────────────────────────────────────────┐
│ رأس GTP-U (12 بايت)                          │
├──────────────────────────────────────────────┤
│ الإصدار، PT، 0 | العلامات x32                │
│ 26 | نوع الرسالة (0x1A)                      │
│ 9 بايت | الطول                               │
│ TEID | 0 (دائمًا)                            │
│ متغير | رقم التسلسل                          │
│ رقم N-PDU | 0                                │
│ 0 | رأس الامتداد التالي                      │
├──────────────────────────────────────────────┤
│ IE: بيانات TEID I (5 بايت)                   │
├──────────────────────────────────────────────┤
│ 16 | النوع (0x10)                            │
│ TEID 4 بايت | الخاطئ                         │
└──────────────────────────────────────────────┘
```

متى تكون هذه الأمور مهمة:

السيناريو 1: إعادة تشغيل PGW-U في بنية GTP S5/S8

- SGW-U (OmniUPF) يوجه حركة S5/S8 إلى PGW-U
- PGW-U يعيد التشغيل ويفقد كل حالة النفق S5/S8
- SGW-U يستمر في توجيه الحزم إلى TEIDs القديمة
- PGW-U يرسل إشارات خطأ
- SGW-U **يتوقف تلقائيًا عن استخدام الأنفاق الميتة**

السيناريو 2: إعادة تشغيل UPF النظير في بنية N9

- UPF-1 (OmniUPF) يوجه حركة N9 إلى UPF-2
- UPF-2 يعيد التشغيل
- UPF-1 يتلقى إشارات خطأ

- ينظف الجلسات UPF-1

مثال السجل:

عند تلقي إشارة خطأ:

```
WARN: تلقي إشارة خطأ GTP-U لـ 192.168.50.10:2152 من TEID
0x12345678 - النظير البعيد لا يتعرف على هذا TEID
WARN: وجدت جلسة LocalSEID=42 مع FAR GlobalId=1 توجه إلى TEID
0 الخاطئx12345678 192.168.50.10 من النظير
INFO: حذف الجلسة LocalSEID=42 بسبب إشارة خطأ GTP-U لـ TEID
0x12345678 192.168.50.10 من
WARN: بسبب إشارة خطأ (ات) تم حذف 1 جلسة GTP-U لـ TEID 0x12345678 من
192.168.50.10 النظير
```

مقاييس Prometheus:

راقب نشاط إشارة الخطأ مع تفاصيل لكل نظير ولكل عقدة:

```
# إجمالي إشارات الخطأ المستلمة من الأقران
upf_buffer_listener_error_indications_received_total{node_id="pgw-u-
1",peer_address="192.168.50.10"}

# الجلسات المحذوفة بسبب إشارات الخطأ
upf_buffer_listener_error_indication_sessions_deleted_total{node_id="
u-1",peer_address="192.168.50.10"}

# إشارات الخطأ المرسلة (لـ TEIDs الواردة غير المعروفة)
upf_buffer_listener_error_indications_sent_total{node_id="enodeb-
1",peer_address="10.60.0.1"}
```

علامات المقاييس:

- `node_id`: معرف عقدة PFCP (أو "غير معروف" إذا لم يكن هناك ارتباط من الارتباط)
- `peer_address`: عنوان IP للنظير البعيد

تساعد هذه المقاييس في تحديد الأقران المشكوك فيها وتتبع أنماط إشارات الخطأ لكل عقدة من الطائرة التحكم.

ملاحظات هامة:

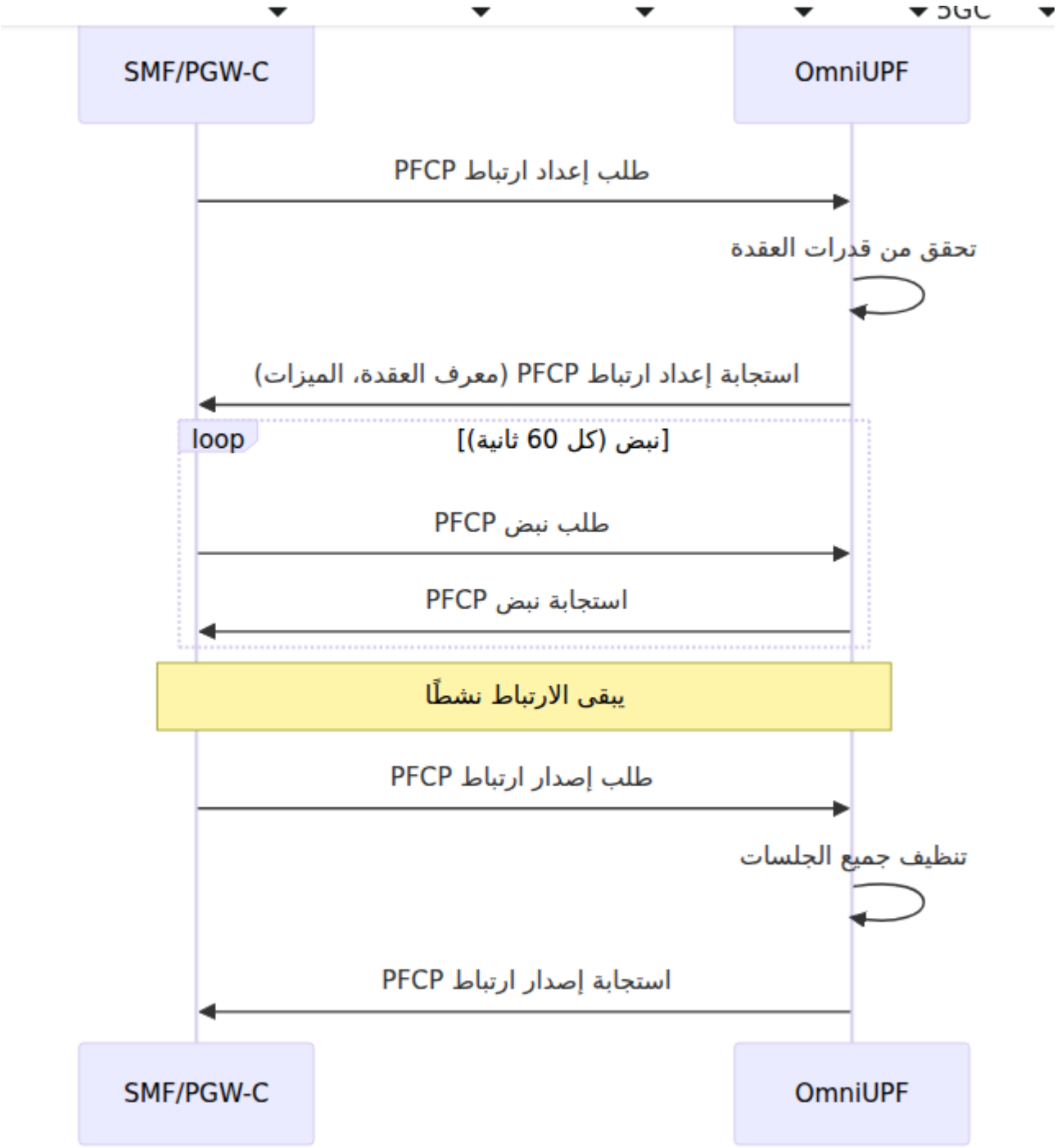
1. **التنظيف التلقائي**: لا حاجة لتدخل المشغل - يتم حذف الجلسات تلقائيًا
2. **مطابقة TEID**: يتم حذف الجلسات فقط التي تحتوي على FARs توجه إلى TEID الخاطئ بالضبط
3. **العزل لكل نظير**: تؤثر إشارات الخطأ من نظير واحد فقط على الجلسات التي توجه إلى ذلك النظير
4. **جلسات متعددة**: إذا كانت هناك جلسات متعددة توجه إلى نفس TEID الميت، يتم **حذف جميعها**
5. **تكامل مع ختم الاسترداد**:
    - اكتشاف ختم الاسترداد = استباقي (يكتشف إعادة التشغيل أثناء إعداد الارتباط)
    - معالجة إشارة الخطأ = تفاعلي (يكتشف الأنفاق الميتة عندما تتدفق حركة المرور)
6. **معالجة الحزم غير الصالحة**: يتم تسجيل إشارات الخطأ غير الصالحة وتجاهلها (لا يتم حذف أي جلسات)

للاستكشاف الأخطاء في إشارات الخطأ، انظر استكشاف أخطاء إشارات GTP-U.

إنشاء جلسة PFCP

عندما يقوم UE بإنشاء جلسة PDU (5G) أو سياق PDP (LTE)، يقوم SMF بإنشاء جلسة PFCP في UPF.

تدفق إنشاء الجلسة:



محتويات الجلسة النموذجية:

- **PDR** مطابقة على: **الرفع** TEID N3، التوجيه عبر FAR إلى N6
- **PDR** مطابقة على عنوان: **النزول** IP الخاص بـ UE، التوجيه عبر FAR إلى N3 مع تغليف GTP-U
- **FAR**: معلمات التوجيه (إنشاء رأس خارجي، مثيل الشبكة)
- **QER**: حدود QoS (MBR، GBR) وعلامة الحزم (QFI)
- **URR**: تقرير الحجم للفوترة (اختياري)

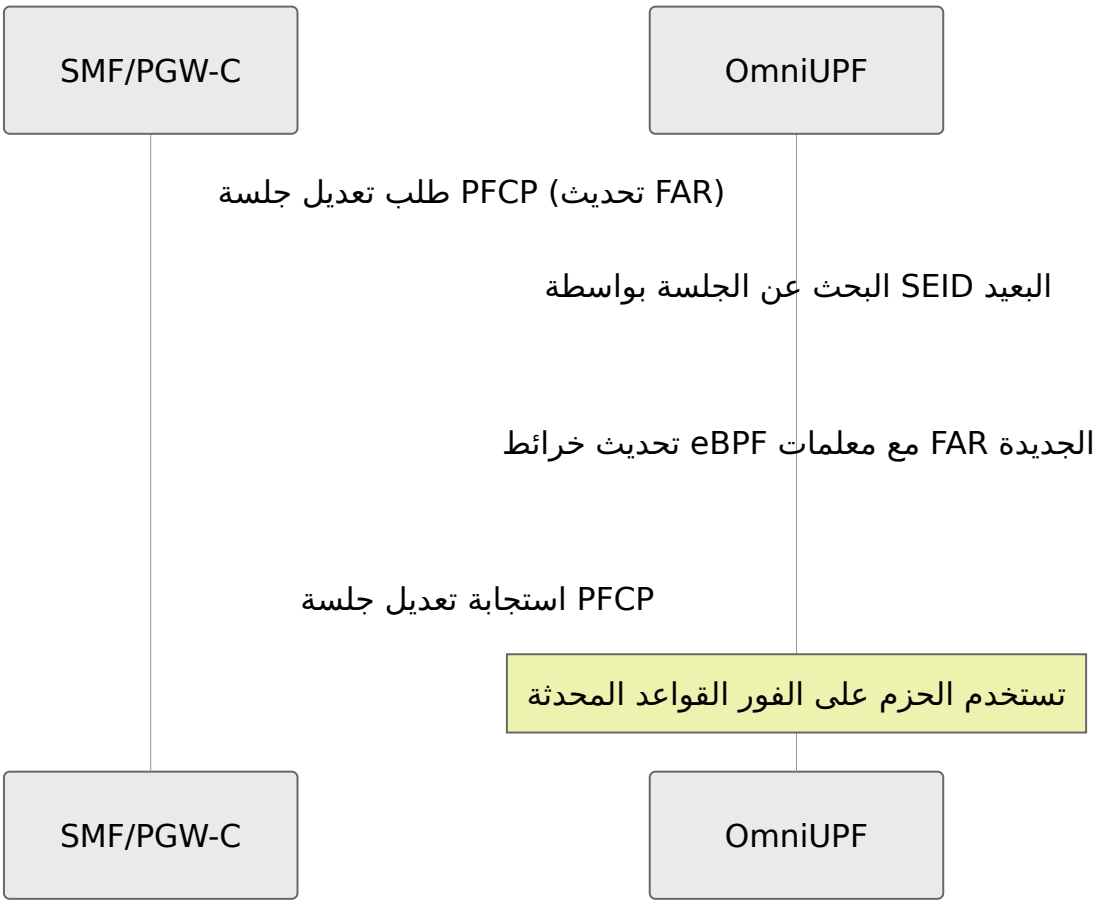
تعديل جلسة PFCP

يمكن لـ SMF تعديل الجلسات لأحداث الحركة (التبديل)، تغييرات QoS، أو تحديثات الخدمة.

السيناريوهات الشائعة للتعديل:

1. **التبديل (مبني على N2)**
    - تحديث FAR لـ gNB (F-TEID) الرفع مع نقطة النفق الجديدة
    - تخزين الحزم مؤقتًا خلال تبديل المسار
    - تفريغ التخزين المؤقت إلى المسار الجديد عند الاستعداد
2. **تغيير QoS**
    - تحديث QER بقيم جديدة لـ MBR/GBR
    - قد تضيف/تزيل مرشحات SDF في PDR لتحديد QoS الخاص بالتطبيق
3. **تحديث الخدمة**
    - إضافة PDRs إضافية ا��جديدة لتدفقات حركة المرور
    - تعديل FARs لتغييرات التوجيه

تدفق تعديل الجلسة:

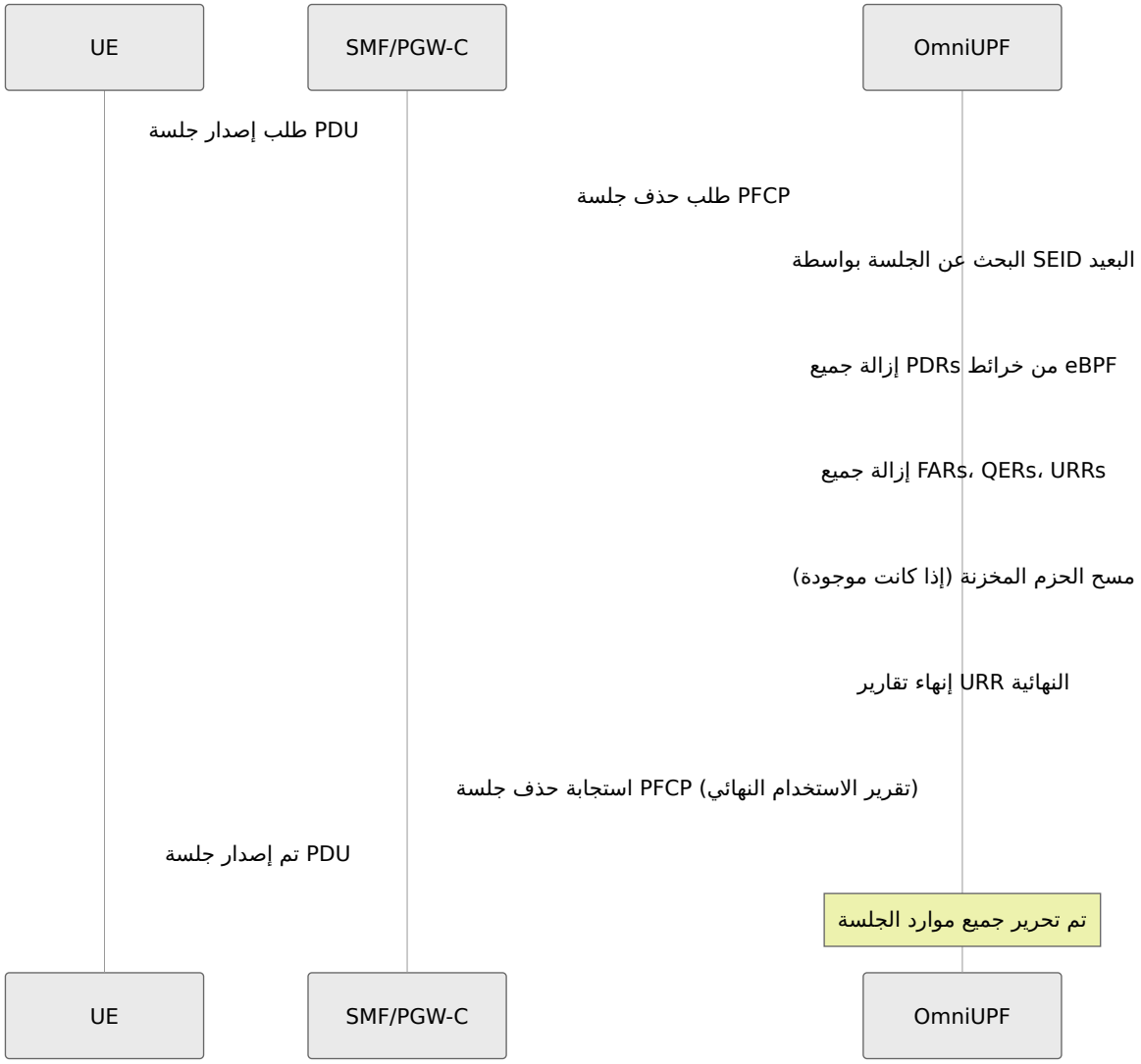


لإدارة القواعد، انظر دليل إدارة القواعد.

حذف جلسة PFCP

عندما يتم إصدار جلسة PDU، يقوم SMF بحذف جلسة PFCP في UPF.

تدفق حذف الجلسة:



التنظيف المنفذ:

- تتم إزالة جميع PDRs (الرفع والنزول)
- تتم إزالة جميع URRs و QERs و FARs
- يتم مسح التخزين المؤقت للحزم
- يتم إرسال تقرير الاستخدام النهائي إلى SMF للفوترة

العمليات الشائعة

توفر OmniUPF قدرات تشغيل شاملة من خلال لوحة التحكم المستندة إلى الويب و API REST. تغطي هذه القسم المهام التشغيلية الشائعة وأهميتها.

مراقبة الجلسات

فهم جلسات PFCP:

تمثل جلسات PFCP جلسات PDU النشطة لـ UE (5G) أو سياقات PDP (LTE). تحتوي كل جلسة على:

- SEIDs (معرفات نقطة الجلسة) المحلي والبعيد
- PDRs لتصنيف الحزم
- FARs لقرارات التوجيه
- QERs لإنفاذ QoS (اختياري)
- URRs لتتبع الاستخدام (اختياري)

العمليات الرئيسية للجلسة:

- **عرض جميع الجلسات** مع عناوين IP الخاصة بـ UE، TEIDs، وعدد القواعد
- **تصفية الجلسات** حسب عنوان IP أو TEID
- **فحص تفاصيل الجلسة** بما في ذلك تكوينات PDR/FAR/QER/URR الكاملة
- **مراقبة عدد الجلسات** لكل ارتباط PFCP

للحصول على إجراءات الجلسة التفصيلية، انظر عرض الجلسات.

إدارة القواعد

قواعد اكتشاف الحزم (PDR):

تحدد PDRs الحزم التي تطابق تدفقات حركة المرور المحددة. يمكن للمشغلين:

- **عرض PDRs الرفع** المفاتيح بواسطة TEID من واجهة N3
- **عرض PDRs النزول** المفاتيح بواسطة عنوان IP الخاص بـ UE
- **فحص مرشحات SDF** لتصنيف حركة المرور الخاصة بالتطبيق
- **مراقبة عدد PDRs** واستخدام السعة

قواعد إجراء التوجيه (FAR):

تحدد FARs ما يجب فعله مع الحزم المطابقة. يمكن للمشغلين:

- **عرض إجراءات FAR** (التوجيه، الإسقاط، التخزين، التكرار، الإخطار)
- **فحص معلمات التوجيه** (إنشاء رأس خارجي، الوجهة)
- **مراقبة حالة التخزين المؤقت** لكل FAR
- **تبديل التخزين المؤقت** لقواعد FAR معينة أثناء استكشاف الأخطاء وإصلاحها

قواعد إنفاذ QoS (QER):

تطبق QERs حدود النطاق الترددي وعلامة الحزم. يمكن للمشغلين:

- **عرض معلمات QoS** (علامات الحزم، GBR، MBR)
- **مراقبة QERs النشطة** لكل جلسة
- **فحص علامات QFI** لتدفقات QoS 5G في

قواعد تقرير الاستخدام (URR):

تتبع URRs أحجام البيانات للفوترة. يمكن للمشغلين:

- **عرض عدادات الحجم** (الرفع، النزول، الإجمالي)
- **مراقبة عتبات الاستخدام** ومشغلات التقرير
- **فحص URRs النشطة** عبر جميع الجلسات

لإدارة القواعد، انظر دليل إدارة القواعد.

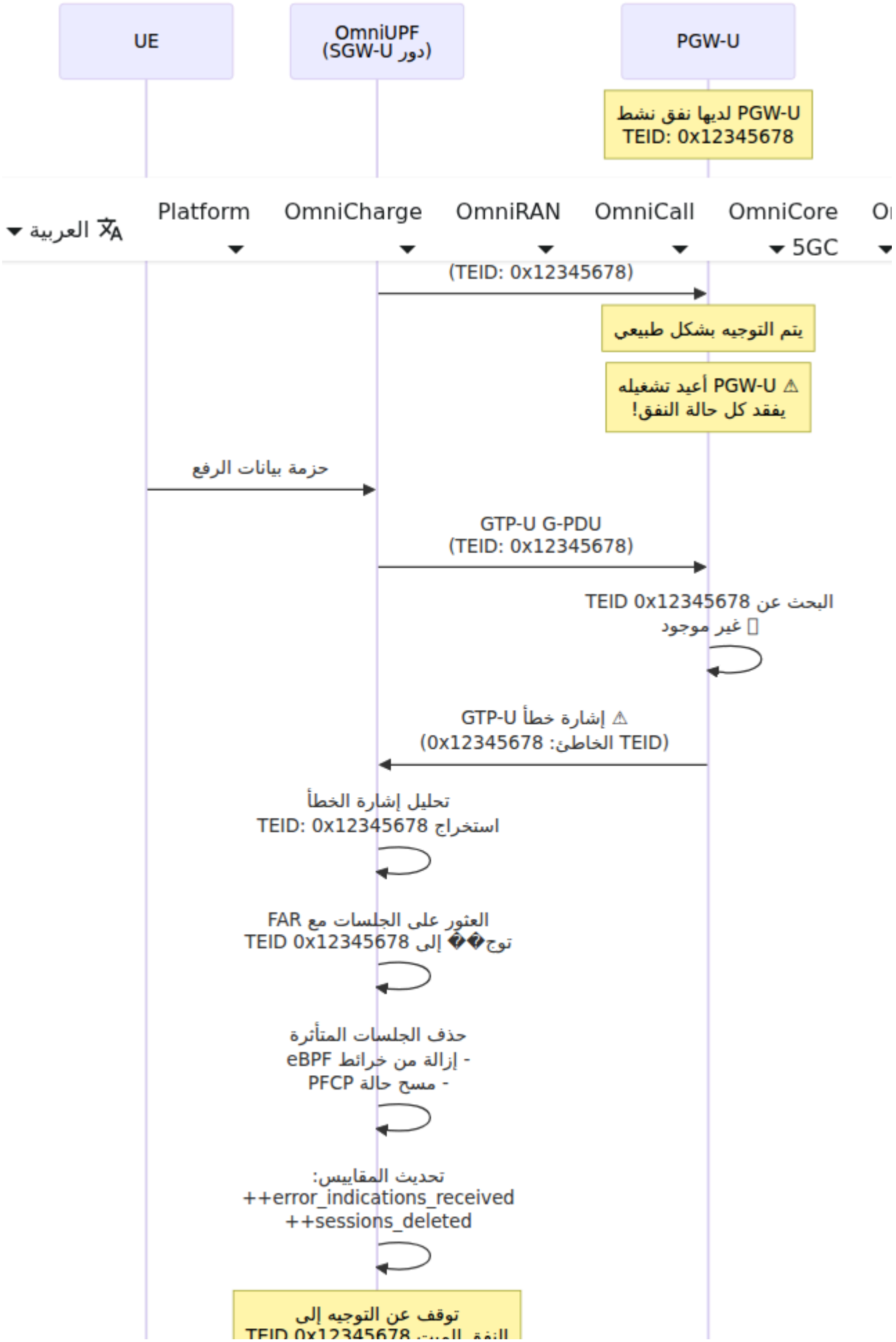
التخزين المؤقت للحزم

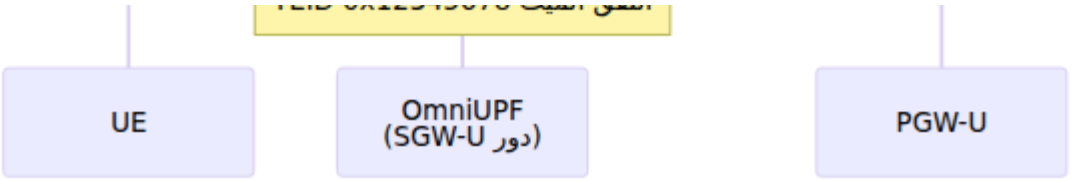
لماذا يعد التخزين المؤقت أمرًا حيويًا لـ UPF

يعد التخزين المؤقت للحزم واحدة من أهم وظائف UPF لأنه يمنع فقدان الحزم أثناء أحداث الحركة وإعادة تكوين الجلسات. بدون التخزين المؤقت، سيواجه المستخدمون المحمولون فقدانًا في الاتصالات، وانقطاعات في التنزيلات، وفشل في الاتصالات في الوقت الحقيقي في كل مرة ينتقلون فيها بين أبراج الخلايا أو عندما تتغير ظروف الشبكة.

المشكلة: فقدان الحزم أثناء الحركة

في الشبكات المحمولة، يتحرك المستخدمون باستمرار. عندما ينتقل جهاز من برج خلية إلى آخر (التبديل)، أو عندما يحتاج الشبكة إلى إعادة تكوين مسار البيانات، هناك نافذة حرجة حيث تكون الحزم في الطيران ولكن المسار الجديد ليس جاهزًا بعد:


UE
OmniUPF
(SGW-U دور)
PGW-U
PGW-U لديها نفق نشط
TEID: 0x12345678
العربية
Platform
OmniCharge
OmniRAN
OmniCall
OmniCore
5GC
(TEID: 0x12345678)
يتم التوجيه بشكل طبيعي
⚠ PGW-U أعيد تشغيله
يفقد كل حالة النفق!
حزمة بيانات الرفع
GTP-U G-PDU
(TEID: 0x12345678)
البحث عن TEID 0x12345678
غير موجود
⚠ إشارة خطأ GTP-U
(TEID الخاطئ: 0x12345678)
تحليل إشارة الخطأ
استخراج TEID: 0x12345678
العثور على الجلسات مع FAR
توج�ه إلى TEID 0x12345678
حذف الجلسات المتأثرة
- إزالة من خرائط eBPF
- مسح حالة PFCP
تحديث المقاييس:
++error_indications_received
++sessions_deleted
توقف عن التوجيه إلى



بدون التخزين المؤقت: ستفقد الحزم التي تصل خلال هذه النافذة الحرجة **، مما يتسبب في:

- **توقف اتصالات TCP** (توقف تصفح الويب، انقطاعات التنزيل) أو إعادة تعيينها
- **تجميد مكالمات الفيديو** (فشل مكالمات Zoom، Teams، WhatsApp) أو انقطاعها
- **انقطاع جلسات الألعاب** (فشل الألعاب عبر الإنترنت، التطبيقات في الوقت الحقيقي)
- **فقدان مكالمات VoIP** (انقطاعات المكالمات) تمامًا
- **فشل التنزيلات** وضرورة إعادة التشغيل

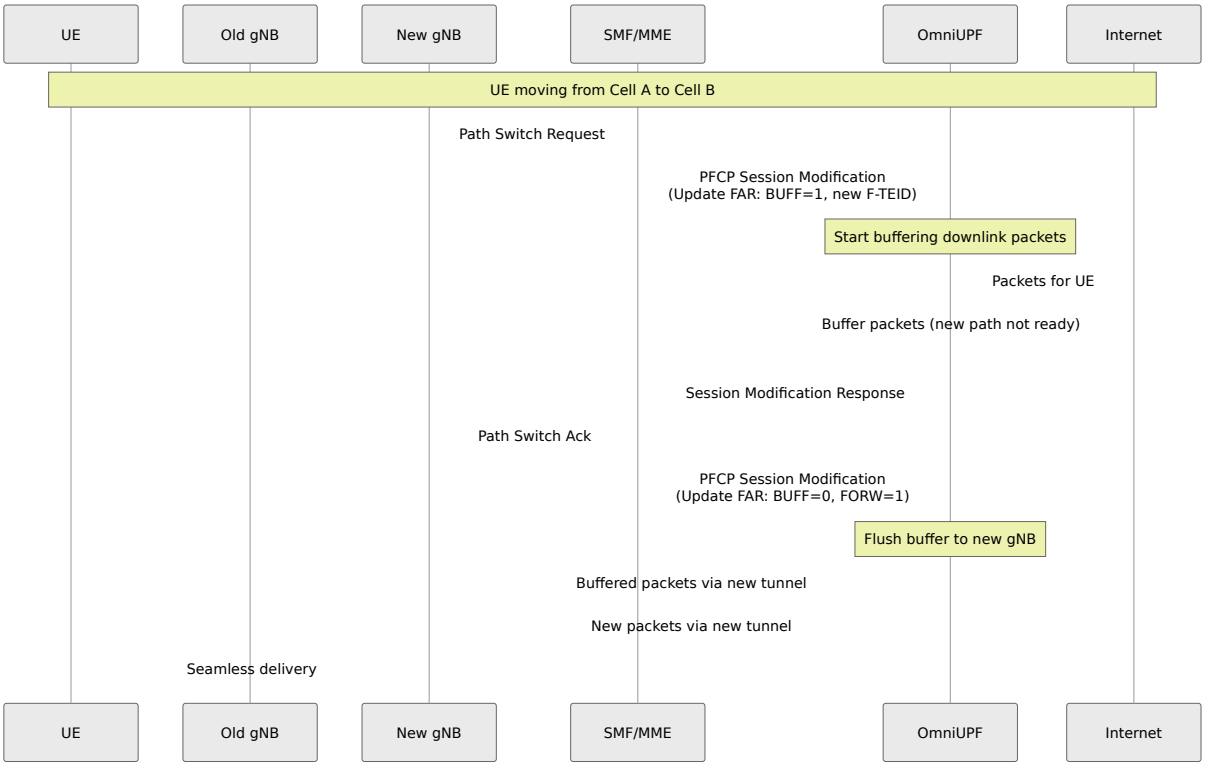
مع التخزين المؤقت: يحتفظ OmniUPF بالحزم مؤقتًا حتى يتم إنشاء المسار الجديد، ثم يوجهها بسلاسة. يختبر المستخدم **عدم وجود انقطاع**.

متى يحدث التخزين المؤقت

يخزن OmniUPF الحزم في هذه السيناريوهات الحرجة:

1. التبديل المبني على N2 (5G) / التبديل المبني على X2 (4G)

عندما ينتقل UE بين أبراج الخلايا:



الجدول الزمني:

- **T+0ms**: لا يزال المسار القديم نشطًا
- **T+10ms**: يخبر SMF UPF بالتخزين المؤقت (المسار القديم يغلق، المسار الجديد ليس جاهزًا)
- **T+10-50ms**: تصل الحزم ولكن لا يمكن **نافذة التخزين المؤقت الحرجة** توجيهها
- **T+50ms**: يخبر SMF UPF بالتوجيه المسار الجديد جاهز،
- **T+50ms+**: يقوم UPF بتفريغ الحزم المخزنة إلى المسار الجديد، ثم يوجه الحزم الجديدة على الفور

40~ من الحزم (آلاف الحزم المحتملة) ستفقد **بدون التخزين المؤقت**: ms **مع التخزين**.** عدم فقدان الحزم، انتقال سلس **المؤقت**:.

---

2. تعديل الجلسة (تغيير QoS، تحديث المسار)

عندما تحتاج الشبكة إلى تغيير معلمات الجلسة:

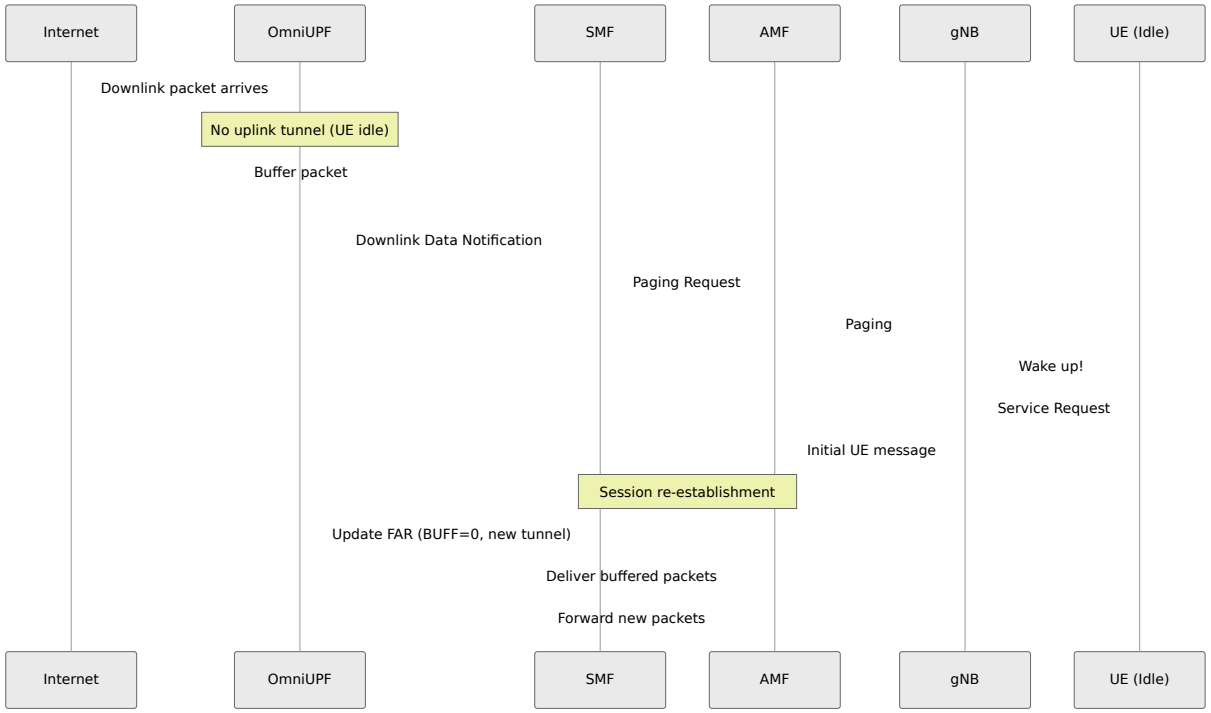
- **ترقية/خفض QoS**: ينتقل المستخدم من تغطية 4G (وضع NSA) إلى 5G
- **تغيير السياسة**: يدخل المستخدم المؤسسي الحرم الجامعي (تغييرات توجيه الحركة)
- **تحسين الشبكة**: يعيد توجيه الشبكة الأساسية الحركة إلى UPF أقرب (تحديث ULCL)

خلال التعديل، قد تحتاج الطائرة التحكم إلى تحديث قواعد متعددة بشكل ذري. يضمن التخزين المؤقت عدم توجيه الحزم مع مجموعات قواعد جزئية/غير متسقة.

---

3. إشعار بيانات النزول (استعادة وضع الخمول)

في وضع الخمول (إيقاف الشاشة، توفير البطارية) وتصل بيانات النزول UE عندما يكون:



بدون التخزين المؤقت: سيتم فقدان الحزمة الأولية التي أثارت الإشعار **، مما يتطلب من المرسل إعادة الإرسال (يضيف زمن التأخير). **مع التخزين المؤقت**: يتم تسليم الحزمة التي أيقظت UE على الفور عند إعادة الاتصال بـ UE.

---

4. التبديل بين RAT (4G ↔ 5G)

عندما ينتقل UE بين 4G و 5G:

- تتغير البنية (eNodeB ↔ gNB)
- تتغير نقاط النفق (مختلفة TEID تخصيص)
- يضمن التخزين المؤقت انتقالًا سلسًا بين أنواع RAT

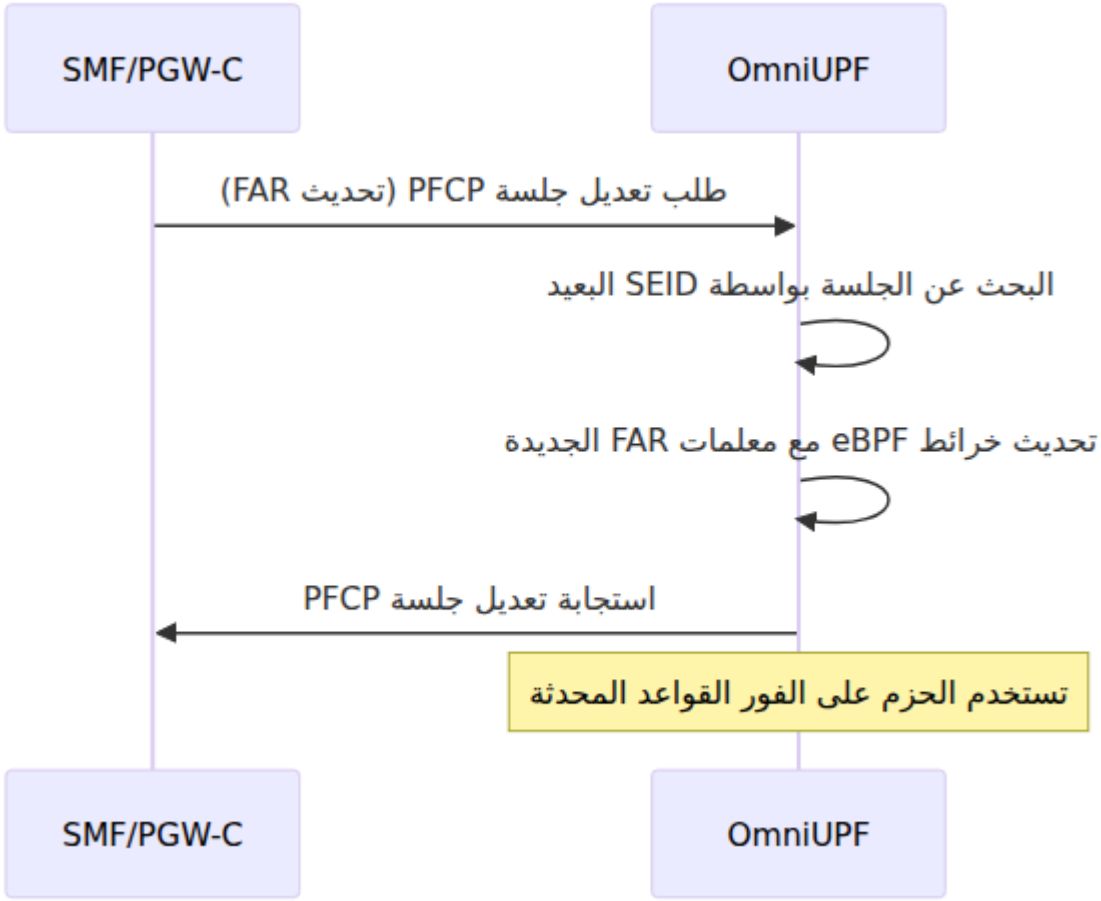
كيف يعمل التخزين المؤقت في OmniUPF

الآلية الفنية:

تستخدم OmniUPF **بنية تخزين مؤقت من مرحلتين**:

1. **مرحلة eBPF (النواة)**: تكشف عن الحزم التي تتطلب التخزين المؤقت بناءً على أعلام إجراء FAR
2. **مرحلة مساحة المستخدم**: تخزن وتدير الحزم المخزنة في الذاكرة

عملية التخزين المؤقت:



تفاصيل رئيسية:

- **منفذ التخزين المؤقت**: منفذ UDP 22152 (الحزم المرسلة من مساحة eBPF إلى المستخدم)
- **التغليف**: يتم تغليف الحزم في GTP-U مع FAR ID كـ TEID
- **التخزين**: تخزين الحزم في الذاكرة لكل FAR، مع بيانات التعريف (الطابع الزمني، الاتجاه، حجم الحزمة)
- **الحدود**:
  - حد لكل FAR: 10,000 حزمة (افتراضي)

    - حد عالمي: 100,000 حزمة عبر جميع FARs
    - TTL: 30 ثانية (افتراضي) - يتم حذف الحزم التي تتجاوز TTL
- **التنظيف**: تقوم عملية خلفية بإزالة الحزم المنتهية كل 60 ثانية

دورة حياة التخزين المؤقت:

1. **تمكين التخزين المؤقت**: يحدد SMF إجراء FAR BUFF=1 (البند 2) عبر تعديل جلسة PFCP
2. **تخزين الحزم**: تكشف eBPF عن علم BUFF، 22152 تغلف الحزم، ترسل إلى المنفذ
3. **تخزين مساحة المستخدم**: يقوم مدير التخزين المؤقت بتخزين الحزم مع FAR ID، الطابع الزمني، الاتجاه
4. **تعطيل التخز��ن المؤقت**: يحدد SMF إجراء FAR FORW=1، BUFF=0 مع معلمات التوجيه الجديدة
5. **تفريغ التخزين المؤقت**: يقوم مساحة المستخدم بإعادة تشغيل الحزم المخزنة باستخدام قواعد FAR (نقطة النفق الجديدة) الجديدة
6. **استئناف الوضع الطبيعي**: يتم توجيه الحزم الجديدة على الفور عبر المسار الجديد

لماذا يهم هذا لتجربة المستخدم

التأثير في العالم الحقيقي:

السيناريو	بدون التخزين المؤقت	مع التخزين المؤقت
مكالمة فيديو أثناء التبديل	تتجمد المكالمة لمدة 1-2 ثانية، قد تنقطع	سلس، لا يوجد انقطاع
تنزيل ملف عند حافة الخلية	يفشل التنزيل، يجب إعادة البدء	يستمر التنزيل دون انقطاع
لعب ألعاب عبر الإنترنت أثناء الحركة	تنقطع الاتصال، يتم طردك من اللعبة	لعبة سلسة، لا انقطاعات
مكالمة VoIP في السيارة	تنقطع المكالمة في كل تبديل	واضحة تمامًا، لا انقطاعات
بث فيديو على القطار	يتوقف الفيديو، تنخفض الجودة	تشغيل سلس
نقطة ساخنة محمولة للكمبيوتر المحمول	تتوقف جلسة SSH، تفشل مكالمة الفيديو	يتم الحفاظ على جميع الاتصالات

فوائد مشغل الشبكة:

- **تقليل معدل انقطاع المكالمات (CDR)**: KPI حيوي لجودة الشبكة
- **زيادة رضا العملاء**: لا يلاحظ المستخدمون التبديلات
- **خفض تكاليف الدعم**: عدد أقل من الشكاوى حول انقطاعات الاتصال
- **ميزة تنافسية**: تسويق "أفضل شبكة للتغطية"

عمليات إدارة التخزين المؤقت

يمكن للمشغلين مراقبة والتحكم في التخزين المؤقت عبر واجهة الويب و API:

المراقبة:

- **عرض الحزم المخزنة** لكل FAR ID (العدد، البايتات، العمر)
- **تتبع استخدام التخزين المؤقت** لكل FAR، (عالمي) مقابل الحدود
- **تنبيه عند تجاوز التخزين المؤقت** أو مدة التخزين المؤقت المفرطة

- **تحديد التخزين المؤقت العالق** (حزم مخزنة > عتبة TTL)

عمليات التحكم:

- **تفريغ التخزين المؤقت**: تشغيل يدوي لتفريغ التخزين المؤقت (استكشاف الأخطاء وإصلاحها)
- **مسح التخزين المؤقت**: حذف الحزم المخزنة (تنظيف التخزين المؤقت العالق)
- **تعديل TTL**: تغيير وقت انتهاء صلاحية الحزم
- **تعديل الحدود**: زيادة سعة التخزين المؤقت لكل FAR أو عالمي

استكشاف الأخطاء:

- **التخزين المؤقت لا يتفريغ**: تحقق مما إذا كان SMF قد أرسل تحديث FAR لتعطيل التخزين المؤقت
- **تجاوز التخزين المؤقت**: زيادة الحدود أو التحقيق في سبب مدة التخزين المؤقت المفرطة
- **حزم قديمة في التخزين المؤقت**: قد يكون TTL مرتفعًا جدًا، أو تأخر تحديث FAR
- **تخزين مؤقت مفرط**: قد يشير إلى مشكلات في الحركة أو مشاكل في SMF

للحصول على عمليات التخزين المؤقت التفصيلية، انظر دليل إدارة التخزين المؤقت.

تكوين التخزين المؤقت

قم بتكوين سلوك التخزين المؤقت في `/etc/omniupf/runtime.exs`:

```
# إعدادات التخزين المؤقت
buffer_port = 22152                # منفذ UDP للحزم المخزنة (افتراضي)
```

التوصيات:

- **الشبكات عالية الحركة** (الطرق السريعة، القطارات): زيادة `buffer_max_packets` إلى 20,000+
- **المناطق الحضرية الكثيفة** (التبديلات المتكررة): تقليل `buffer_packet_ttl` إلى 15s
- **التطبيقات ذات زمن التأخير المنخفض**: تعيين `buffer_packet_ttl` إلى 10s لمنع البيانات القديمة

- **شبكات IoT**: تقليل الحدود (حركة مرور أقل أثناء التبديل IoT تولد أجهزة)

للحصول على خيارات التكوين الكاملة، انظر دليل التكوين.

الإحصائيات والمراقبة

إحصائيات الحزم:

مقاييس معالجة الحزم في الوقت الحقيقي بما في ذلك:

- **حزم RX**: إجمالي الحزم المستلمة من جميع الواجهات
- **حزم TX**: إجمالي الحزم المرسلة إلى جميع الواجهات
- **حزم مفقودة**: الحزم التي تم إسقاطها بسبب الأخطاء أو السياسة
- **حزم GTP-U**: عدادات الحزم الموجهة

إحصائيات المسار:

مقاييس التوجيه لكل مسار:

- **ضربات المسار**: الحزم المطابقة لكل مسار
- **عدادات التوجيه**: النجاح/الفشل لكل وجهة
- **عدادات الأخطاء**: TEIDs غير صالحة، عناوين IP غير معروفة

إحصائيات XDP:

مقاييس أداء eXpress Data Path:

- **XDP المعالجة**: الحزم التي تم التعامل معها في طبقة XDP
- **XDP المارة**: الحزم المرسلة إلى كومة الشبكة
- **XDP المفقودة**: الحزم التي تم إسقاطها في طبقة XDP
- **XDP الملغاة**: أخطاء المعالجة

إحصائيات واجهة N3/N6:

عدادات حركة المرور لكل واجهة:

- **N3 RX/TX**: حركة المرور إلى/من RAN (gNB/eNodeB)
- **N6 RX/TX**: حركة المرور إلى/من شبكة البيانات

- **إجمالي عدادات الحزم**: إحصائيات واجهة مجمعة

للحصول على تفاصيل المراقبة، انظر دليل المراقبة.

إدارة السعة

مراقبة سعة خريطة eBPF:

يعتمد أداء UPF على سعة خريطة eBPF. يمكن للمشغلين:

- **مراقبة استخدام الخريطة** مع مؤشرات النسبة في الوقت الحقيقي
- **عرض حدود السعة** لكل خريطة eBPF
- **تنبيهات ملونة**:
  - أخضر (<50%): تشغيل طبيعي
  - أصفر (50-70%): حذر
  - كهرماني (70-90%): تحذير
  - أحمر (>90%): حرجة

الخرائط الحرجة للمراقبة:

- `uplink_pdr_map`: تصنيف حركة المرور الرفع
- `downlink_pdr_map`: تصنيف حركة المرور النزول IPv4
- `far_map`: قواعد التوجيه
- `qer_map`: قواعد QoS
- `urr_map`: تتبع الاستخدام

تخطيط السعة:

- إدخال خريطة واحدة (حجم المفتاح + حجم القيمة) تستهلك كل PDR
- حد ذاكرة النواة) يتم تكوين سعة الخريطة عند بدء UPF)
- يؤدي تجاوز السعة إلى فشل إنشاء الجلسات

للحصول على مراقبة السعة، انظر إدارة السعة.

إدارة التكوين

تكوين UPF:

عرض والتحقق من معلمات التشغيل UPF:

- **واجهة N3**: عنوان IP للاتصال بـ RAN (GTP-U)
- **واجهة N6**: عنوان IP للاتصال بشبكة البيانات
- **واجهة N9**: عنوان IP للاتصال بين UPF (اختياري)
- **واجهة PFCP**: عنوان IP للاتصال بـ SMF
- **منفذ API**: منفذ API REST للاستماع
- **نقطة نهاية المقاييس**: منفذ مقاييس Prometheus

عرض تكوين dataplane:

معلمات مسار البيانات النشطة:

- **العنوان النشط N3**: ربط واجهة N3 في الوقت الحقيقي
- **العنوان النشط N9**: ربط واجهة N9 (إذا تم تمكينه) في الوقت الحقيقي

للحصول على عرض التكوين، انظر عرض التكوين.

استكشاف الأخطاء وإصلاحها

يغطي هذا القسم المشكلات التشغيلية الشائعة واستراتيجيات الحل.

فشل إنشاء الجلسات

الأعراض: تفشل جلسات إنشاء اتصال بيانات UE في الإنشاء، لا يمكن لـ PFCP

الأسباب الجذرية الشائعة:

1. **لم يتم إنشاء ارتباط PFCP**
    - تحقق من إمكانية وصول SMF إلى واجهة PFCP الخاصة بـ UPF (المنفذ 8805)
    - تحقق من حالة ارتباط PFCP في عرض الجلسات

- تحقق من تكوين معرف العقدة يتطابق بين SMF و UPF

2. **استنفاد سعة خريطة eBPF**
   - تحقق من عرض السعة للخرائط الحمراء (<90%)
   - زيادة أحجام خريطة eBPF في تكوين UPF
   - حذف الجلسات القديمة إذا كانت الخريطة ممتلئة

3. **تكوين PDR/FAR غير صالح**
   - تحقق من أن عنوان IP الخاص بـ UE وصالح فريد
   - تحقق من عدم تعارض تخصيص TEID
   - تأكد من أن FAR تشير إلى مثيلات الشبكة الصالحة

4. **مشكلات تكوين الواجهة**
   - تحقق من أن عنوان واجهة N3 قابل للوصول من gNB
   - تحقق من جداول التوجيه لـ N6 للاتصال بشبكة البيانات
   - تأكد من عدم حظر حركة مرور GTP-U بواسطة جدار الحماية

للحصول على استكشاف الأخطاء التفصيلية، انظر دليل استكشاف الأخطاء.

فقدان الحزم أو مشكلات التوجيه

الأعراض: يمتلك UE اتصالاً ولكن يعاني من فقدان الحزم أو عدم تدفق الحركة

الأسباب الجذرية الشائعة:

1. **تكوين PDR غير صحيح**
   - تحقق من أن TEID الرفع PDR يتطابق مع TEID المعين من gNB
   - تحقق من أن عنوان IP النزول PDR يتطابق مع عنوان IP المعين
   - تحقق من مرشحات SDF لقواعد صارمة للغاية

2. **مشكلات إجراء FAR**
   - تحقق من أن إجراء FAR هو FORWARD (ليس DROP أو BUFFER)
   - تحقق من معلمات إنشاء الرأس الخارجي لـ GTP-U

- تأكد من أن نقطة النهاية الوجهة صحيحة

3. **تجاوز حدود QoS**

- تحقق من إعدادات QER MBR (معدل البت الأقصى)
- تحقق من GBR (معدل البت المضمون) تخصيص
- راقب فقدان الحزم بسبب تحديد المعدل

4. **مشكلات MTU للواجهة**

- تحقق من أن الحمل الزائد لـ GTP-U (بايت 40-50) لا يسبب التجزئة
- تحقق من تكوين MTU لواجهة N3/N6
- راقب رسائل ICMP المطلوبة للت

دليل هندسة OmniUPF

جدول المحتويات

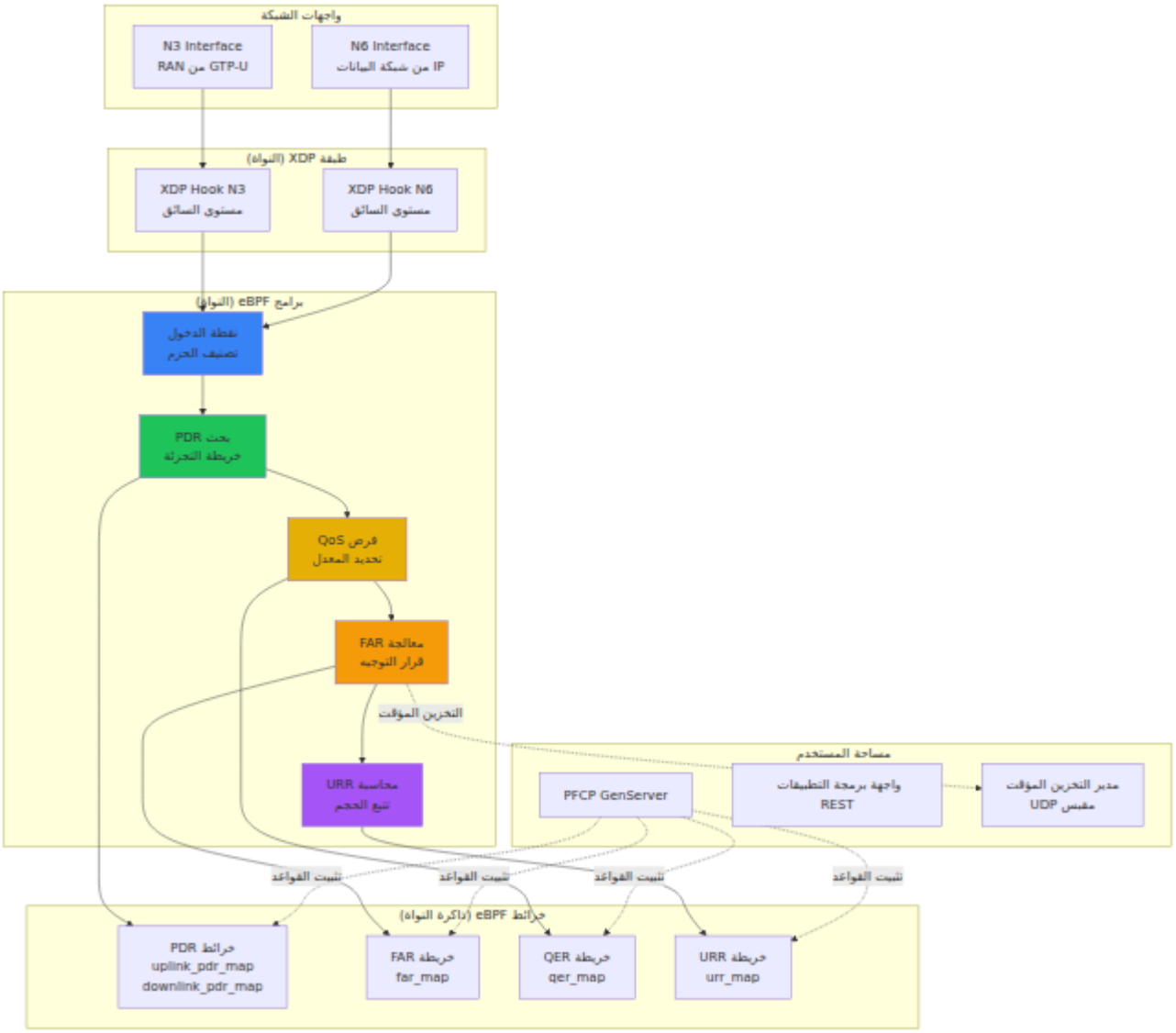
1. نظرة عامة
2. أساسيات تقنية eBPF
3. مسار بيانات XDP
4. خط معالجة الحزم
5. هندسة خريطة eBPF
6. آلية التخزين المؤقت
7. فرض جودة الخدمة (QoS)
8. خصائص الأداء
9. قابلية التوسع والضبط

نظرة عامة

يستفيد OmniUPF من eBPF (مرشح الحزم الممتد من بيركلي) وXDP (مسار البيانات السريع) لمعالجة حزم carrier-grade 5G/LTE. يتم تنفيذ مستوى التحكم في Elixir/OTP باستخدام إدارة جلسات PFCP المعتمدة على GenServer، بينما يتم تشغيل برامج eBPF مباشرة في نواة Linux. هذه الفجوة تلغي عبء معالجة الحزم في مساحة المستخدم وتحقق من خلال ذلك إنتاجية متعددة الجيجابت مع زمن استجابة بالميكروثانية.

يتم تجميع ملف كائن eBPF (`ipentrypoint_bpf.o`) من مصدر C (في دليل `ebpf/xdp/`) وتحميله في وقت التشغيل بواسطة مستوى التحكم في Elixir عبر روابط NIF إلى libbpf.

طبقات الهندسة



مبادئ التصميم الرئيسية

معالجة بدون نسخ:

- تتم معالجة الحزم بالكامل في مساحة النواة
- لا يوجد نسخ للبيانات بين النواة ومساحة المستخدم
- التلاعب المباشر بالحزم باستخدام XDP

هياكل بيانات خالية من القفل:

- جداول تجزئة لكل وحدة معالجة مركزية eBPF تست��دم خرائط
- عمليات ذرية للوصول المتزامن

- لا يوجد عبء قفل/قفل دوار

جاهز لتحميل الأجهزة:

- SmartNIC يدعم تنفيذ XDP وضع تحميل
- XDP متوافق مع بطاقات الشبكة التي تدعم
- العودة إلى الأوضاع الأصلية أو العامة للسائق

أساسيات تقنية eBPF

ما هو eBPF؟

eBPF (تسمح بتشغيل برامج آمنة Linux هو تقنية ثورية في نواة (مرشح الحزم الممتد من بيركلي ومحمية في مساحة النواة دون تغيير كود المصدر للنواة أو تحميل وحدات النواة.

الميزات الرئيسية:

- من أن البرامج لا يمكن أن تتسبب في تعطل النواة eBPF **الأمان**: يتحقق مُحقق
- **الأداء**: تعمل بسرعة النواة الأصلية (بدون عبء تفسير)
- **المرونة**: يمكن تحديثها في وقت التشغيل دون إعادة تشغيل النواة
- **قابلية المراقبة**: تتبع وإحصائيات مدمجة

دورة حياة برنامج eBPF

خرائط eBPF

خرائط eBPF هي هياكل بيانات في النواة مشتركة بين برامج eBPF ومساحة المستخدم.

أنواع الخرائط المستخدمة في OmniUPF:

نوع الخريطة	الوصف	حالة الاستخدام
`BPF_MAP_TYPE_HASH`	جدول تجزئة مع أزواج المفتاح والقيمة	بحث PDR بوا��طة UE IP أو TEID
`BPF_MAP_TYPE_ARRAY`	مصفوفة مؤشرة بواسطة عدد صحيح	بحث FAR وQER وURR بواسطة ID
`BPF_MAP_TYPE_PERCPU_HASH`	جدول تجزئة لكل وحدة معالجة مركزية (خالي من القفل)	بحث PDR عالي الأداء
`BPF_MAP_TYPE_LRU_HASH`	جدول تجزئة LRU (الأقل استخدامًا مؤخرًا)	الإخلاء التلقائي للمدخلات القديمة

عمليات الخريطة:

- **بحث**: O(1) (بحث تجزئة (دون ميكروثانية
- **تحديث**: تحديثات ذرية من مساحة المستخدم
- **حذف**: إزالة فورية للمدخلات
- **تكرار**: عمليات دفعة لتفريغ الخرائط

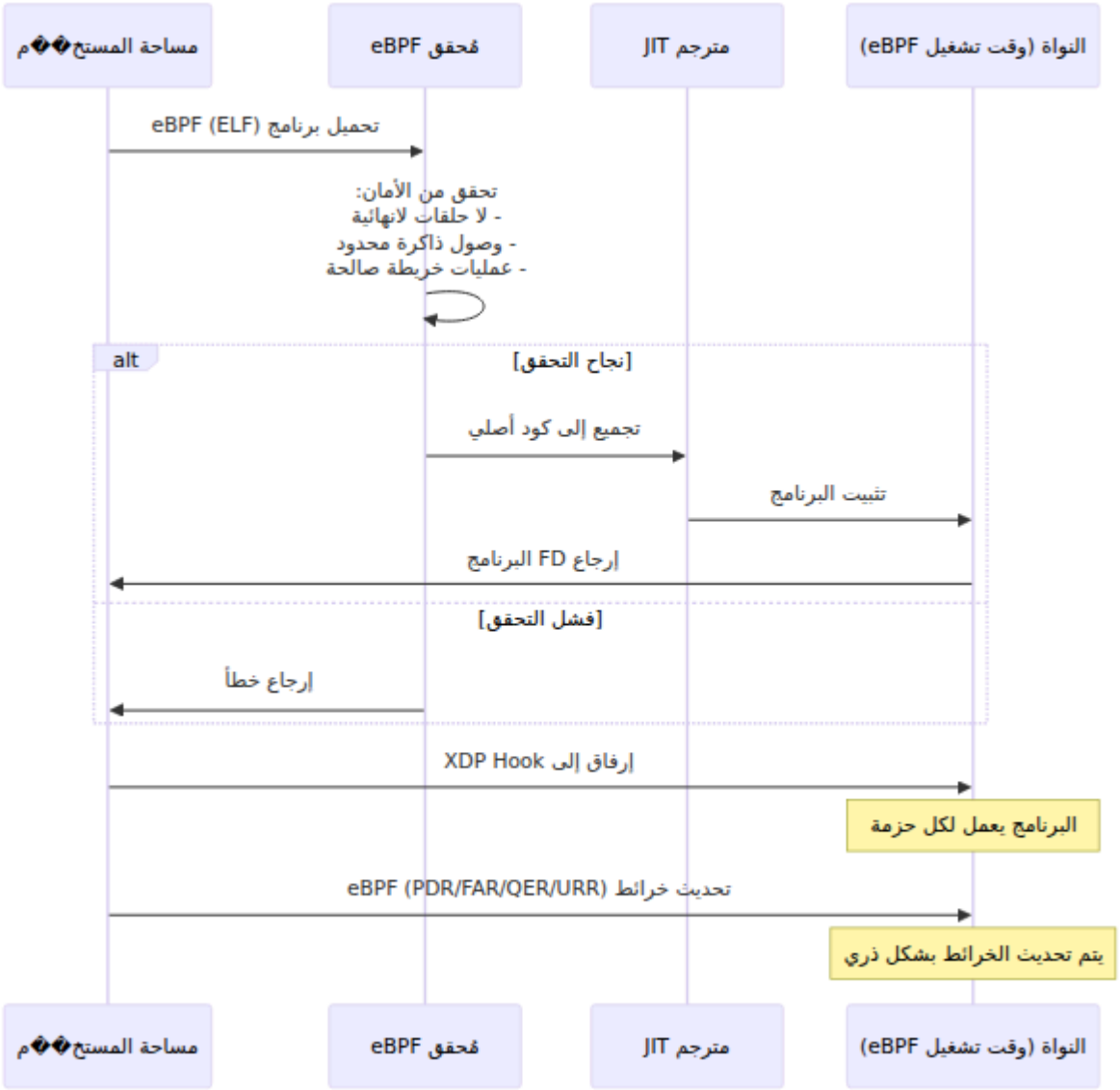
مسار بيانات XDP

نظرة عامة على XDP

XDP (مسار البيانات السريع) هو نقطة ربط في نواة Linux تسمح لبرامج eBPF بمعالجة الحزم في أقرب نقطة ممكنة - مباشرة بعد أن تستلمها سائق الشبكة، قبل كومة الشبكة في النواة.

أوضاع إرفاق XDP

يدعم OmniUPF ثلاثة أوضاع إرفاق XDP، كل منها له خصائص أداء وتوافق مختلفة.



1. وضع تحميل XDP

(أفضل أداء) **تنفيذ الأجهزة**:

- يعمل برنامج eBPF مباشرة على أجهزة SmartNIC
- دون لمس وحدة المعالجة المركزية NIC معالجة الحزم في
- يحقق إنتاجية تزيد عن 100 جيجابت في الثانية
- يتطلب SmartNIC متوافق (Netronome، Mellanox ConnectX-6)

التكوين (في `runtime.exs`):

```
xdp_attach_mode = "offload"
```

القيود:

- يتطلب أجهزة SmartNIC باهظة الثمن
- تعقيد برنامج eBPF محدود
- مدعومة في الأجهزة eBPF ليست جميع ميزات

---

2. وضع XDP الأصلي (الافتراضي للإنتاج)

تنفيذ على مستوى السائق (أداء عالٍ):

- يعمل برنامج eBPF في سياق سائق الشبكة
- تتم معالجة الحزم قبل تخصيص SKB (مخزن المقبس)
- يحقق إنتاجية تتراوح بين 10-40 جيجابت في الثانية لكل نواة
- يتطلب XDP سائقًا يدعم (معظم السائقين الحديثين)

التكوين (في `runtime.exs`):

```
xdp_attach_mode = "native"
```

المزايا:

- أداء عالٍ جدًا (ملايين الحزم في الثانية)
- توافق واسع مع الأجهزة
- مجموعة كاملة من ميزات eBPF

السائقون المدعومون:

- Intel: i40e، ice، ixgbe، igb
- Mellanox: mlx4، mlx5
- Broadcom: bnxt
- Amazon: ena
- معظم بطاقات الشبكة 10G+

---

3. وضع XDP العام

محاكاة برمجية (التوافق):

- يعمل برنامج eBPF بعد أن تخصص النواة SKB
- محاكاة برمجية لسلوك XDP
- يعمل على أي واجهة شبكة
- مفيد للاخ��بار والتطوير

التكوين (في `runtime.exs`):

```
xdp_attach_mode = "generic"
```

حالات الاستخدام:

- التطوير والاختبار
- البيئات الافتراضية (VMs بدون SR-IOV)
- الأجهزة الشبكية القديمة
- اختبار واجهة الحلقة

الأداء: 1-5 جيجابت في الثانية (أبطأ بكثير من الأصلي/التحميل)

رموز إرجاع XDP

ترجع برامج eBPF رموز إجراءات XDP لإخبار النواة بما يجب القيام به مع الحزم:

رمز الإرجاع	المعنى	الاستخدام في OmniUPF
`XDP_PASS`	إرسال الحزمة إلى كومة الشبكة في النواة	التخزين المؤقت (التسليم المحلي)، حركة المرور غير المعروفة، ICMP
`XDP_DROP`	إسقاط الحزمة على الفور	حزم غير صالحة، تحديد المعدل، إسقاط السياسات
`XDP_TX`	إرسال الحزمة مرة أخرى عبر نفس الواجهة	غير مستخدم حاليًا
`XDP_REDIRECT`	إرسال الحزمة إلى واجهة مختلفة	المسار الرئيسي للتوجيه (N3 ↔ N6)
`XDP_ABORTED`	خط�� في المعالجة، إسقاط الحزمة وتسجيلها	أخطاء برنامج eBPF

خط معالجة الحزم

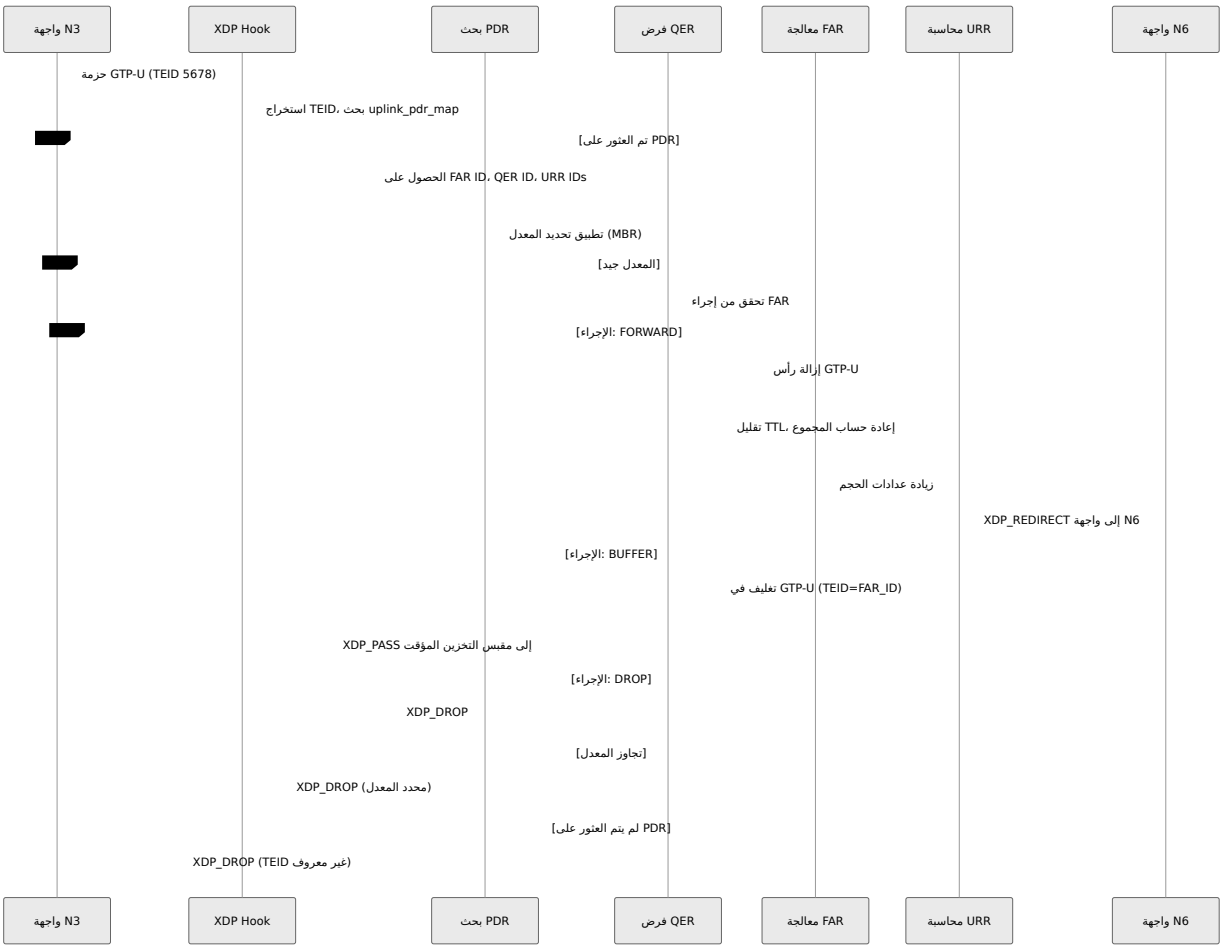
هيكل البرنامج

يستخدم OmniUPF استدعاءات ذيل eBPF لإنشاء خط معالجة حزم معياري.

استدعاءات الذيل:

- تسمح برامج eBPF باستدعاء برامج eBPF أخرى
- تعيد استخدام نفس إطار المكدس (عمق مكدس محدود)
- تمكّن تصميم خط أنابيب معياري
- الحد الأقصى لعمق استدعاء الذيل هو 33

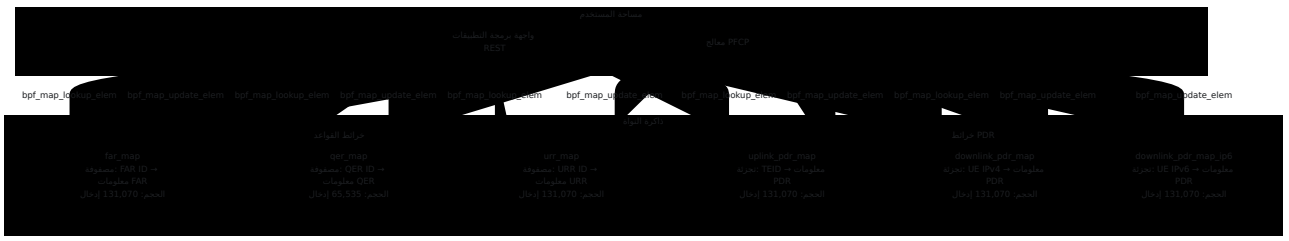
معالجة حزم الرفع



معالجة حزم التنزيل

هندسة خريطة eBPF

تخطيط ذاكرة الخريطة



حجم الخريطة

يحسب OmniUPF تلقائيًا أحجام الخرائط بناءً على تكوين `max_sessions`:

```
خرائط PDR = 2 × max_sessions  (uplink + downlink)
خرائط FAR = 2 × max_sessions  (uplink + downlink)
خرائط QER = 1 × max_sessions  (مشاركة لكل جلسة)
خرائط URR = 3 × max_sessions  (لكل جلسة عدة URRs)
```

مثال (max_sessions = 65,535):

- خرائط PDR: 131,070 إدخال لكل منها
- خريطة FAR: 131,070 إدخال
- خريطة QER: 65,535 إدخال
- خريطة URR: 131,070 إدخال

إجمالي الذاكرة:

```
خرائط PDR: 3 × 131,070 × 212 B = ~83 MB
خريطة FAR:  131,070 × 20 B = ~2.6 MB
خريطة QER:  65,535 × 36 B = ~2.3 MB
خريطة URR:  131,070 × 20 B = ~2.6 MB
91~ :الإجمالي MB ذاكرة النواة
```

آلية التخزين المؤقت

نظرة عامة على التخزين المؤقت

ينفذ OmniUPF تخزين الحزم المؤقتة لسيناريوهات الانتقال عن طريق تغليف الحزم في GTP-U وإرسالها إلى عملية مساحة المستخدم عبر مقبس UDP.

هندسة التخزين المؤقت

Parse error on line 4: ...r/>2. إضافة رأس UDP (22152 المنفذ)
3 ----------
------^ Expecting 'SQE', 'DOUBLECIRCLEEND', 'PE', '-)', 'STADIUMEND',

'SUBROUTINEEND', 'PIPE', 'CYLINDEREND', 'DIAMOND_STOP', 'TAGEND', 'TRAPEND', 'INVTRAPEND', 'UNICODE_TEXT', 'TEXT', 'TAGSTART', got 'PS'

المحاولة مجددا

تفاصيل تغليف التخزين المؤقت

يقوم برنامج eBPF، تم تعيين بت إجراء (FAR 2) ع��دما يتم تمكين التخزين المؤقت:

1. **حساب حجم الحزمة الأصلية**:

```
orig_packet_len = ntohs(ip->tot_len);  // من رأس IP
```

2. **توسيع رأس الحزمة**:

```
// إضافة مساحة لـ: IP الخارجي + UDP + GTP-U
gtp_encap_size = sizeof(struct iphdr) + sizeof(struct
udphdr) + sizeof(struct gtpuhdr);
bpf_xdp_adjust_head(ctx, -gtp_encap_size);
```

3. **بناء رأس IP الخارجي**:

```
ip->saddr = original_sender_ip;  // الحفاظ على المصدر لتجنب
تصفية المارتين
ip->daddr = local_upf_ip;        // IP المحلي حيث يرتبط مستمع
مساحة المستخدم
ip->protocol = IPPROTO_UDP;
ip->ttl = 64;
```

4. **بناء رأس UDP**:

```
udp->source = htons(22152);  // BUFFER_UDP_PORT
udp->dest = htons(22152);
udp->len = htons(sizeof(udphdr) + sizeof(gtpuhdr) +
orig_packet_len);
```

5. **بناء رأس GTP-U**:

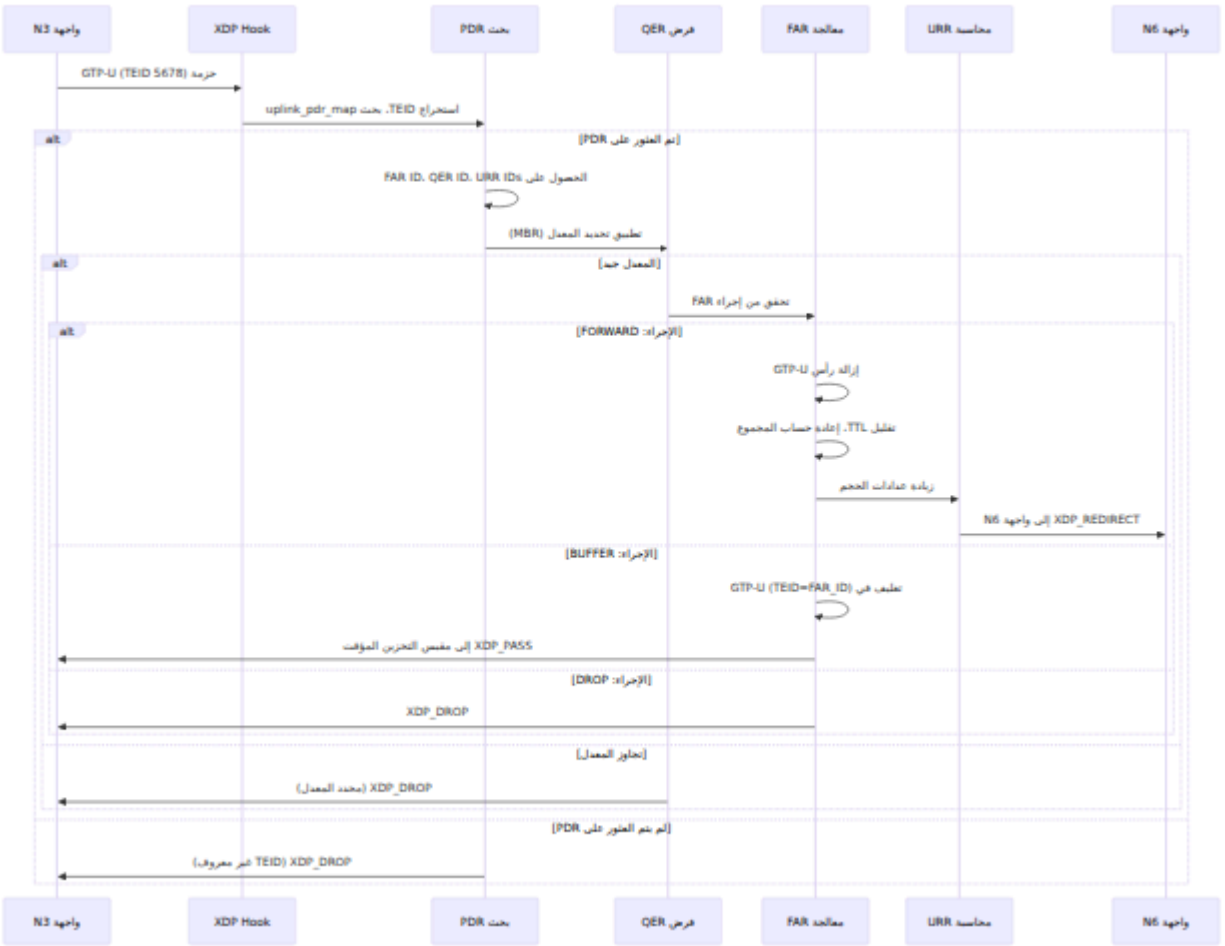
```
gtp->version = 1;
gtp->message_type = GTPU_G_PDU;
gtp->teid = htonl(far_id | (direction << 24));  // ترميز FAR ID والاتجاه
gtp->message_length = htons(orig_packet_len);
```

6. **إرجاع XDP_PASS**:

   - تسلم النواة الحزمة إلى مقبس UDP 22152 المحلي على المنفذ
   - يتلقى مدير التخزين المؤقت في مساحة المستخدم الحزمة ويخزنها

عملية تفريغ التخزين المؤقت

يتم إعادة تشغيل الحزم .BUFFER لمسح علامة FAR بتحديث SMF عندما يكتمل الانتقال، يقوم المؤقتة:



معلمات إدارة التخزين المؤقت

المعلمة	الافتراضي	الوصف
الحد الأقصى لكل FAR	10,000 حزمة	الحد الأقصى للحزم المؤقتة لكل FAR
الحد الأقصى الإجمالي	100,000 حزمة	الحد الأقصى للحزم المؤقتة الإجمالية
الحزمة TTL	30 ثانية	الوقت قبل انتهاء صلاحية الحزم المؤقتة
منفذ التخزين المؤقت	22152	منفذ UDP لتسليم التخزين المؤقت
فترة تنظيف التخزين المؤقت	60 ثانية	كم مرة يتم التحقق من الحزم المنتهية الصلاحية

فرض جودة الخدمة (QoS)

خوارزمية تحديد المعدل

ينفذ OmniUPF لفرض جودة الخدمة **محدد معدل نافذة منزلقة**.

Parse error on line 5: ...= packet_size × 8 × (NSEC_PER_SEC / rate ----------^ Expecting 'SQE', 'DOUBLECIRCLEEND', 'PE', '-)', 'STADIUMEND', 'SUBROUTINEEND', 'PIPE', 'CYLINDEREND', 'DIAMOND_STOP', 'TAGEND', 'TRAPEND', 'INVTRAPEND', 'UNICODE_TEXT', 'TEXT', 'TAGSTART', got 'PS'

المحاولة مجددا

تنفيذ النافذة المنزلقة

الخوارزمية (من `qer.h`):

```c
static __always_inline enum xdp_action limit_rate_sliding_window(
    const __u64 packet_size,
    volatile __u64 *window_start,
    const __u64 rate)
{
    static const __u64 NSEC_PER_SEC = 1000000000ULL;
    static const __u64 window_size = 5000000ULL;  // نافذة 5 مللي ثانية

    // المعدل = 0 يعني غير محدود
    if (rate == 0)
        return XDP_PASS;

    // حساب وقت الإرسال لهذه الحزمة
    __u64 tx_time = packet_size * 8 * (NSEC_PER_SEC / rate);
    __u64 now = bpf_ktime_get_ns();

    // تحقق مما إذا كنا متقدمين على النافذة (سترسل الحزمة في المستقبل)
    __u64 start = *window_start;
    if (start + tx_time > now)
        return XDP_DROP;  // تجاوز حد المعدل

    // إذا مرت النافذة، أعد تعيينها
    if (start + window_size < now) {
        *window_start = now - window_size + tx_time;
        return XDP_PASS;
    }

    // تحديث النافذة لتضمين هذه الحزمة
    *window_start = start + tx_time;
    return XDP_PASS;
}
```

المعلمات الرئيسية:

- **حجم النافذة**: 5 مللي ثانية (5,000,000 نانوثانية)
- **لكل اتجاه**: نوافذ منفصلة للرفع والتنزيل
- **تحديثات ذرية**: تستخدم مؤشرات متقلبة للوصول المتزامن
- **MBR = 0**: يعتبر عرض نطاق غير محدود

مثال حساب QoS

السيناريو: MBR = 100 ميغابت في الثانية، حجم الحزمة = 1500 بايت

1. **وقت الإرسال**:

```
tx_time = 1500 بايت × 8 بت/بايت × 1,000,000,000 نانوثانية/ثانية ÷ 100,000,000 بت في الثانية)
tx_time = 1500 × 8 × 10 = 120,000 نانوثانية = 120 ميكروثانية
```

2. **تحقق المعدل**:
   - يمكن إرسال الحزمة التالية عند `t=0`، إذا تم إرسال الحزمة الأخيرة عند `t=120µs`
   - يتم إسقاطها (مبكر جدًا) إذا وصلت الحزمة عند `t=100µs`
   - يتم إرسالها (تمت ترقية النافذة) إذا وصلت الحزمة عند `t=150µs`

3. **أقصى معدل للحزم**:

```
Max PPS = (100 ميغابت في الثانية ÷ 8) ÷ 1500 بايت = 8,333 حزمة في الثانية
الفجوة بين الحزم = 120 ميكروثانية
```

خصائص الأداء

الإنتاجية

التكوين	الإنتاجية	الحزم في الثانية	زمن الاستجابة
تحميل XDP (SmartNIC)	100 جيجابت في الثانية	148 مليون حزمة في الثانية	< 1 ميكروثانية
XDP الأصلي (10G NIC، نواة واحدة)	10 جيجابت في الثانية	8 مليون حزمة في الثانية	2-5 ميكروثانية
XDP الأصلي (10G NIC، 4 نوى)	40 جيجابت في الثانية	32 مليون حزمة في الثانية	2-5 ميكروثانية
XDP العام	1-5 جيجابت في الثانية	0.8-4 مليون حزمة في الثانية	50-100 ميكروثانية

تحليل زمن الاستجابة

إجمالي زمن معالجة الحزمة (XDP الأصلي):

المرحلة	زمن الاستجابة	التراكمي
NIC RX	0.5 ميكروثانية	0.5 ميكروثانية
استدعاء XDP Hook	0.1 ميكروثانية	0.6 ميكر��ثانية
بحث PDR (تجزئة)	0.3 ميكروثانية	0.9 ميكروثانية
تحقق معدل QER	0.1 ميكروثانية	1.0 ميكروثانية
معالجة FAR	0.5 ميكروثانية	1.5 ميكروثانية
تحديث URR	0.2 ميكروثانية	1.7 ميكروثانية
إزالة التغليف/تغليف GTP-U	0.8 ميكروثانية	2.5 ميكروثانية
XDP_REDIRECT	0.5 ميكروثانية	3.0 ميكروثانية
NIC TX	0.5 ميكروثانية	3.5 ميكروثانية

(XDP 10 الأصلي، IG NIC) **الإجمالي**: ~3.5 ميكروثانية لكل حزمة

استخدام وحدة المعالجة المركزية

قدرة المعالجة لكل نواة:

- (الأصلي XDP) نواة واحدة: 8-10 مليون حزمة في الثانية
- مع تفعيل خيوط المعالجة: 12-15 مليون حزمة في الثانية
- توسيع متعدد النوى: تقريبًا خطي حتى 8 نوى

استخدام وحدة المعالجة المركزية حسب معدل الحزم:

```
نسبة استخدام وحدة المعالجة المركزية ≈ (معدل الحزم / 10,000,000) × 100% لكل نواة
```

مثال: حركة مرور 2 مليون حزمة في الثانية تستخدم ~20% من ��واة واحدة

عرض النطاق الترددي للذاكرة

الوصول إلى خريطة eBPF:

- بحث التجزئة: ~100 نانوثانية (ضربة ذاكرة)
- بحث التجزئة: ~300 نانوثانية (فشل في ذاكرة التخزين المؤقت)
- بحث المصفوفة: ~50 نانوثانية (دائمًا ضربة ذاكرة)

عرض النطاق الترددي المطلوب للذاكرة:

```
عرض النطاق الترددي = معدل الحزم × (متوسط حجم الحزمة + عمليات بحث الخريطة × 64 بايت)
```

مثال: حركة مرور 10 مليون حزمة في الثانية × (1500 بايت + 3 عمليات بحث × 64 بايت) ≈ 160 جيجابت في الثانية من عرض النطاق الترددي للذاكرة

قابلية التوسع والضبط

التوسع الأفقي

عدة مثيلات UPF:

Setting SMF as parent of SMF would create a cycle

المحاولة مجددا

توزيع الجلسات:

- يقوم SMF بتوزيع الجلسات عبر مثيلات UPF
- يتعامل كل UPF مع مجموعة فرعية من جلسات UE
- لا حاجة للتواصل بين UPF (بدون حالة)

التوسع الرأسي

ضبط وحدة المعالجة المركزية:

1. تمكين ارتباط وحدة المعالجة المركزية لمعالجة XDP

2. استخدام RSS (توزيع جانب الاستقبال) لتوزيع قوائم RX
3. تثبيت برامج eBPF على نوى معينة

ضبط NIC:

1. زيادة حجم مخزن حلقة RX
2. تمكين NICs متعددة القوائم (RSS)
3. استخدام موجه التدفق لتوجيه الحركة

ضبط النواة:

```
# زيادة حد الذاكرة المقفلة لخرائط eBPF
ulimit -l unlimited

# تعطيل توازن IRQ للنوى XDP
systemctl stop irqbalance

# تعيين حاكم وحدة المعالجة المركزية إلى الأداء
cpupower frequency-set -g performance

# زيادة أحجام المخازن الشبكية
sysctl -w net.core.rmem_max=134217728
sysctl -w net.core.wmem_max=134217728
```

تخطيط السعة

الصيغة:

```
عدد النوى المطلوبة = (معدل PPS المتوقع ÷ 10,000,000) × 1.5 (50%)
احتياطي)
الذاكرة المطلوبة = (212 × الحد الأقصى للجلسات × 3 B) + 100 MB
(النفقات العامة + خرائط eBPF)
الشبكة المطلوبة = (ذروة الإنتاجية × 2) + 10 جيجابت في الثانية
(احتياطي)
```

مثال (1 مليون جلسة، 20 جيجابت ذروة):

- وحدة المعالجة المركزية: (20 جيجابت ÷ 10 جيجابت لكل نواة) × 1.5 = 3-4 نوى
- الذاكرة: (1M × 212 B × 3) + 100 MB ≈ 750 MB

- الشبكة: (20 جيجابت × 2) + 10 جيجابت = 50 جيجابت واجهات

الوثائق ذات الصلة

- **دليل عمليات UPF** - العمليات العامة لـ UPF والنشر
- **دليل إدارة القواعد** - تفاصيل URR وQER وFAR وPDR
- **دليل المراقبة** - مراقبة الأداء والإحصائيات
- **دليل عمليات واجهة الويب** - استخدام لوحة التحكم
- **دليل استكشاف الأخطاء** - المشكلات الشائعة والتشخيصات
- **دليل الحديقة المسورة** BUFF - إع��دة التوجيه عند نفاد الرصيد باستخدام آلية لاعتراض حزم مساحة المستخدم

دليل تكوين OmniUPF

جدول المحتويات

1. نظرة عامة
2. أوضاع التشغيل
3. أوضاع إرفاق XDP
4. معلمات التكوين
5. ملف التكوين
6. توافق المحاكيات
7. توافق NIC
8. أمثلة التكوين
9. تخطيط سعة الخريطة وحجمها

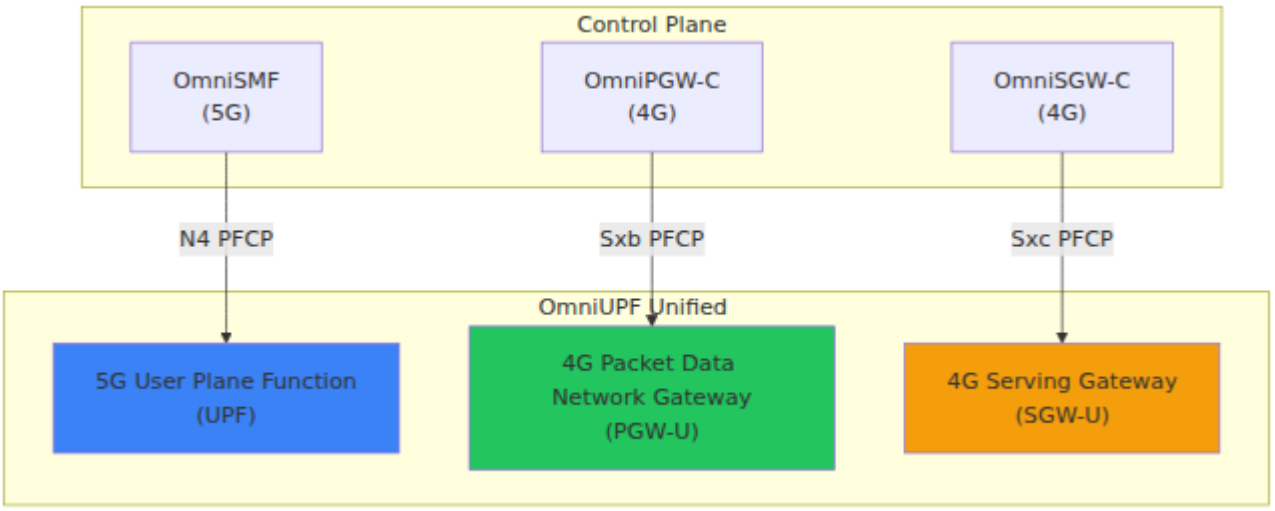
نظرة عامة

OmniUPF هو وظيفة طائرة مستخدم متعددة الاستخدامات يمكن أن تعمل في أوضاع متعددة 4 لدعم كل من الشبكات الأساسية G (EPC) 5وG. يتم تنفيذ مستوى التحكم في Elixir/OTP، ويتم إدارة التكوين من خلال ملف تكوين Elixir (`runtime.exs`).

ويتم الاحتفاظ به `/etc/omniupf/runtime.exs` عند تثبيت الحزمة، يتم وضع ملف التكوين في بعد تغييرات التكوين عبر التحديثات. يجب إعادة تشغيل UPF.

أوضاع التشغيل

يمكن أن تعمل في وقت واحد كـ OmniUPF هو **منصة موحدة**:



تكوين الوضع

الذي ينشئ (SMF أو PGW-C أو SGW-C) يتم **تحديد وضع التشغيل** بواسطة مستوى التحكم للتبديل بين الأوضاع OmniUPF لا يتطلب الأمر تكوينًا محددًا لـ OmniUPF. مع PFCP ارتباطات.

التشغيل المتزامن:

- من عدة مستويات تحكم في وقت واحد PFCP قبول ارتباطات OmniUPF يمكن لـ
- **في نفس** SGW-U و PGW-U و UPF كـ OmniUPF يمكن أن تعمل مثيل واحد من **الوقت**
- يتم عزل الجلسات من مستويات التحكم المختلفة وإدارتها بشكل مستقل

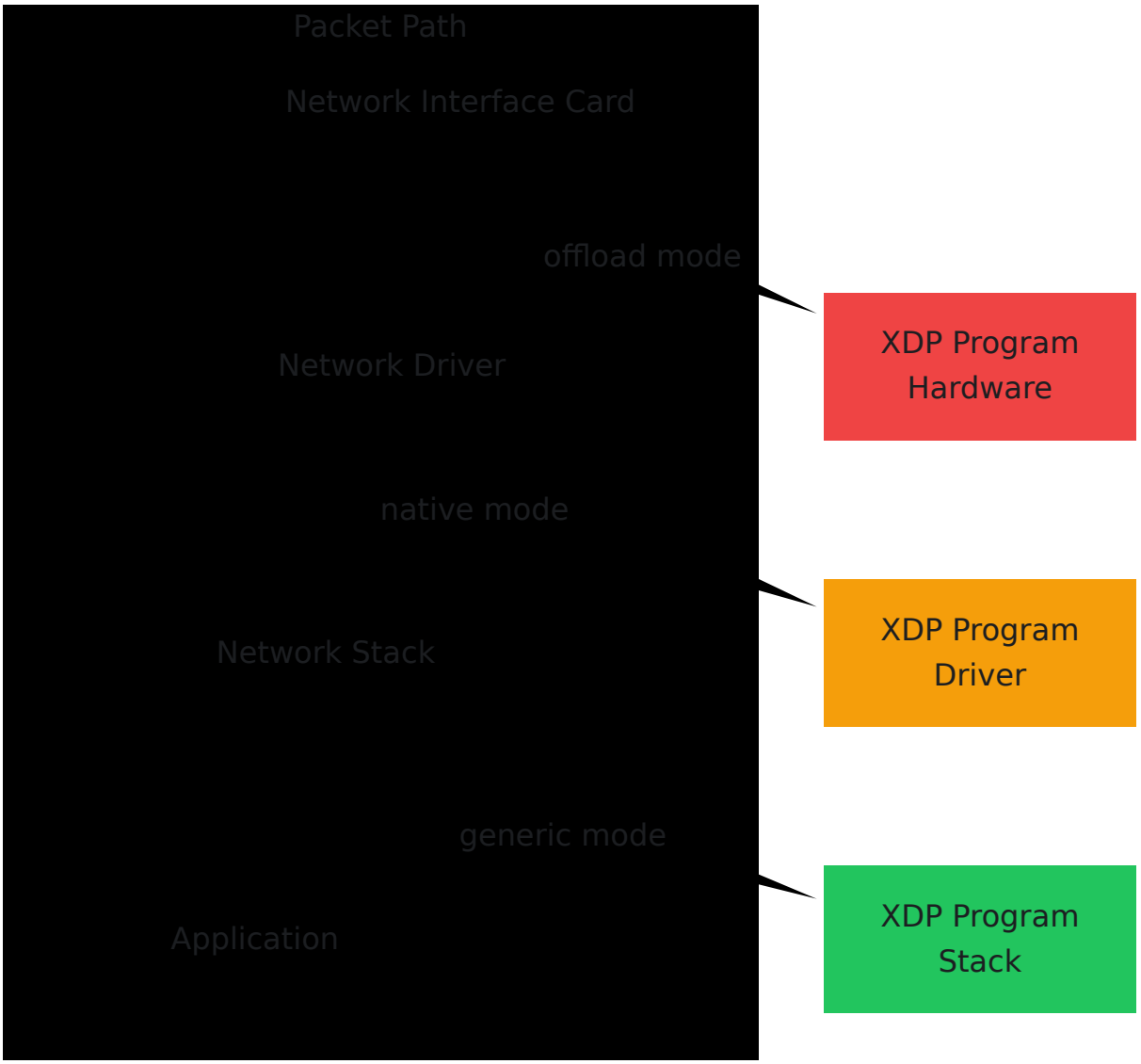
أوضاع إرفاق XDP

لمعالجة الحزم عالية الأداء. يتم دعم ثلاثة (eXpress Data Path) تستخدم OmniUPF XDP أوضاع إرفاق.

والمحاكيات Proxmox التفصيلية، خاصة لـ XDP للحصول على تعليمات إعداد الأخرى، راجع دليل أوضاع XDP.

مقارنة الأوضاع

الوضع	نقطة الإرفاق	الأداء	حالة الاستخدام	متطلبات NIC
عام	مكدس الشبكة (kernel)	~1-2 Mpps	الاختبار، التطوير، التوافق	أي NIC
محلي	برنامج تشغيل الشبكة (kernel)	~5-10 Mpps	الإنتاج (العتاد الفعلي، SR-IOV مع VM)	برنامج تشغيل يدعم XDP
تحميل	عتاد NIC (SmartNIC)	~10-40 Mpps	إنتاج عالي السعة	SmartNIC مع تحميل XDP



الوضع العام (افتراضي)

الوصف: يعمل برنامج XDP في مكدس الشبكة kernel

المزايا:

- يعمل مع **أي** واجهة شبكة
- لا متطلبات خاصة للسائق أو العتاد
- مثالي للاختبار والتطوير
- متوافق مع جميع المحاكيات ومنصات الافتراضية

العيوب:

- أداء أقل (~1-2 Mpps لكل نواة)
- تم تمرير الحزم بالفعل عبر السائق قبل معالجة XDP

التكوين (في `runtime.exs`):

```
xdp_attach_mode = "generic"
```

الأفضل لـ:

- الآلات الافتراضية بدون SR-IOV
- بيئات الاختبار والتحقق
- NICs بدون دعم سائق XDP
- المحاكيات مثل VirtualBox و VMware و Proxmox

الوضع المحلي (موصى به)

الوصف: يعمل برنامج XDP على مستوى برنامج تشغيل الشبكة

المزايا:

- أداء عالي (~5-10 Mpps لكل نواة)
- تتم معالجة الحزم قبل دخول مكدس الشبكة
- زمن وصول أقل بكثير من الوضع العام

- يعمل على العتاد الفعلي و VMs مع SR-IOV

العيوب:

- يتطلب برنامج تشغيل شبكة يدعم XDP
- ليست جميع NICs/تدعم XDP المحلي السائقين

التكوين (في `runtime.exs`):

```
xdp_attach_mode = "native"
```

الأفضل لـ:

- نشرات الإنتاج على العتاد الفعلي
- VMs مع تمرير SR-IOV
- NICs مع برامج تشغيل تدعم XDP (Intel و Mellanox، إلخ.)

المتطلبات:

- برنامج تشغيل شبكة يدعم XDP (راجع توافق NIC)
- نواة Linux 5.15+ مع دعم XDP مفعل

وضع التحميل (أقصى أداء)

الوصف: يعمل برنامج XDP مباشرة على عتاد SmartNIC

المزايا:

- أقصى أداء (~10-40 Mpps)
- صفر عبء على وحدة المعالجة المركزية لمعالجة الحزم
- زمن وصول أقل من ميكروثانية
- يحرر وحدة المعالجة المركزية لمعالجة مستوى التحكم

العيوب:

- يتطلب عتاد SmartNIC باهظ الثمن
- توفر محدود لـ SmartNIC

- إعداد وتكوين معقد

التكوين (في `runtime.exs`):

```
xdp_attach_mode = "offload"
```

الأفضل لـ:

- نشرات إنتاج عالية السعة للغاية
- الحوسبة الطرفية مع متطلبات زمن وصول صارمة
- البيئات التي تكون فيها موارد وحدة المعالجة المركزية محدودة

المتطلبات:

- SmartNIC مع دعم تحميل XDP (Netronome Agilio CX، Mellanox BlueField)
- برامج ثابتة و برامج تشغيل متخصصة

معلمات التكوين

واجهات الشبكة

المعلمة	الوصف	النوع	الافتراضي
`xdp_interfaces`	واجهات الشبكة لحركة مرور N3/N6/N9 (مفصولة بفواصل، أو `"auto"` لجميع غير حلقة العودة)	سلسلة	`"auto"`
`n3_address`	عنوان IPv4 لواجهة N3 (GTP-U من RAN)	IP	`"127.0.0.1"`
`n9_address`	عنوان IPv4 لواجهة N9 (UPF-to-UPF لـ ULCL)	IP	نفس `n3_address`

مثال (في `runtime.exs`):

```
xdp_interfaces = "eth0,eth1"
n3_address = "10.100.50.233"
n9_address = "10.100.50.234"
```

تكوين PFCP

المعلمة	الوصف	النوع	الافتراضي
`pfcp_address`	العنوان المحلي لخادم PFCP (واجهة N4/Sxb/Sxc)	سلسلة IP	`"127.0.0.1"`
`pfcp_port`	منفذ PFCP	عدد صحيح	`8805`
`node_id`	معرف العقدة المحلي لبروتوكول PFCP	سلسلة IP	`"127.0.0.1"`
`heartbeat_interval_ms`	فترة نبض PFCP (بالملي ثانية)	عدد صحيح	`5000`
`heartbeat_timeout_ms`	مهلة نبض PFCP (بالملي ثانية)	عدد صحيح	`5000`
`heartbeat_retries`	عدد محاولات نبض القلب قبل إعلان نظير ميت	عدد صحيح	`3`

مثال (في `runtime.exs`):

```
pfcp_address = "10.100.50.241"
pfcp_port = 8805
node_id = "10.100.50.241"
heartbeat_interval_ms = 10_000
heartbeat_retries = 5
```

API والمراقبة

المعلمة	الوصف	النوع	الافتراضي
`api_port`	المنفذ المحلي لخادم REST API (Phoenix)	عدد صحيح	`8080`
`log_level`	مستوى التسجيل (`:debug`, `:info`, `:warning`, `:error`)	ذرة	`:info`

مثال (في `runtime.exs`):

```
api_port = 8080
log_level = :debug
```

إدارة مسار GTP

المعلمة	الوصف	النوع	الافتراضي
`gtpu_port`	منفذ GTP-U	عدد صحيح	`2152`
`gtp_echo_interval`	فترة الصدى الافتراضية بالثواني	عدد صحيح	`10`
`gtp_echo_retries`	الحد الأقصى لعدد الصدى المفقود قبل فشل المسار	عدد صحيح	`3`
`gtp_peers`	قائمة من أقران GTP لطلبات الصدى	قائمة	`[]`

مثال (في `runtime.exs`):

```
gtp_echo_interval = 15
gtp_echo_retries = 3
gtp_peers = [
  %{address: parse_ip.("10.100.50.50"), echo: true, echo_interval:
10},
  %{address: parse_ip.("10.100.50.60"), echo: false},
]
```

سعة خريطة eBPF

المعلمة	الوصف	النوع	الافتراضي
`max_sessions`	الحد الأقصى لعدد الجلسات المتزامنة	عدد صحيح	`1_000_000`
`far_map_size`	حجم خريطة FAR eBPF	عدد صحيح	`131_070`
`qer_map_size`	حجم خريطة QER eBPF	عدد صحيح	`65_535`
`urr_map_size`	حجم خريطة URR eBPF	عدد صحيح	`65_535`

مثال (في `runtime.exs`):

```
max_sessions = 100_000
far_map_size = 131_070
qer_map_size = 65_535
urr_map_size = 131_070
```

تكوين المخزن المؤقت

المعلمة	الوصف	النوع	الافتراضي
`buffer_port`	منفذ UDP للحزم المخزنة من eBPF	عدد صحيح	`22152`

مثال (في `runtime.exs`):

```
buffer_port = 22152
```

أعلام الميزات

المعلمة	الوصف	النوع	الافتراضي
`feature_ueip`	تمكين تخصيص IP لـ UE بواسطة OmniUPF	منطقي	`false`
`ueip_pool`	مجموعة IP لتخصيص IP لـ UE (تتطلب `feature_ueip`)	سلسلة CIDR	`"10.60.0.0/24"`
`feature_ftup`	تمكين تخصيص F-TEID بواسطة OmniUPF	منطقي	`true`
`teid_pool_start`	بداية نطاق تخصيص TEID	عدد صحيح	`1`
`teid_pool_end`	نهاية نطاق تخصيص TEID	عدد صحيح	`10_000_000`

مثال (تخصيص IP لـ UE) (في `runtime.exs`):

```
feature_ueip = true
ueip_pool = "10.45.0.0/16"
```

مثال (تخصيص F-TEID) (في `runtime.exs`):

```
feature_ftup = true
teid_pool_start = 1
teid_pool_end = 1_000_000
```

تكوين مدير المسار

لتزامن مسار UE مع خادم FRR (Free Range Routing). راجع دليل إدارة المسار للحصول على التفاصيل.

المعلمة	الوصف	النوع	الافتراضي
`route_manager_enabled`	تمكين التزامن التلقائي لمسار UE	منطقي	`true`
`route_manager_type`	نوع خادم التوجيه (`"frr"` أو `"static"`)	سلسلة	`"frr"`

مثال (في `runtime.exs`):

```
route_manager_enabled = true
route_manager_type = "frr"
```

متى يتم التمكين:

- تتطلب إعلان المسار UPF نشرات متعددة لـ
- التكامل مع بروتوكولات التوجيه OSPF أو BGP
- يتطلب تثبيت وتكوين خادم FRRouting

تكوين eBPF / XDP

المعلمة	الوصف	النوع	الافتراضي
`xdp_interfaces`	قائمة مفصولة بفواصل من الواجهات لـ XDP، أو `"auto"` لجميع غير حلقة العودة	سلسلة	`"auto"`
`xdp_attach_mode`	وضع XDP: `"generic"`، `"native"`، أو `"offload"`	سلسلة	`"generic"`
`ebpf_pin_path`	مسار تثبيت نظام الملفات BPF	سلسلة	`"/sys/fs/bpf/upf_pipeline"`
`xdp_obj_path`	المسار إلى كائن eBPF المترجم	سلسلة	`"/etc/omniupf/ipentrypoint_bpf`

مثال (في `runtime.exs`):

```
xdp_interfaces = "auto"              # أو "eth0,eth1" لواجهات
محددة
xdp_attach_mode = "native"
ebpf_pin_path = "/sys/fs/bpf/upf_pipeline"
xdp_obj_path = "/etc/omniupf/ipentrypoint_bpf.o"
```

ملف التكوين

ملف تكوين Elixir (runtime.exs)

الملف: `/etc/omniupf/runtime.exs`

يستخدم ملف التكوين بناء جملة Elixir مع تعيينات متغيرة بسيطة لسهولة القراءة. يتم تطبيق المتغيرات على تكوين التطبيق في أسفل الملف عبر `config :upf_ex, ...`.

```
import Config

parse_ip = fn str ->
  {:ok, addr} = :inet.parse_address(String.to_charlist(str))
  addr
end

#
============================================================================
# PFCP (N4/Sx) Configuration
#
============================================================================
pfcp_address = "10.100.50.241"       # IP address to listen for PFCP
pfcp_port = 8805                      # PFCP port (standard: 8805)
node_id = "10.100.50.241"            # Node ID advertised in Associat
Setup

#
============================================================================
# GTP-U Data Plane Configuration
#
============================================================================
n3_address = "10.100.50.233"         # N3 interface IP (GTP-U toward
n9_address = n3_address               # N9 interface IP (defaults to
gtpu_port = 2152                      # GTP-U port (standard: 2152)
buffer_port = 22152                   # Buffer listener port

#
============================================================================
# eBPF / XDP Configuration
#
============================================================================
xdp_interfaces = "auto"              # "auto" for all non-loopback, c
separated list
xdp_attach_mode = "native"           # XDP mode: "generic", "native",
"offload"
ebpf_pin_path = "/sys/fs/bpf/upf_pipeline"
xdp_obj_path = "/etc/omniupf/ipentrypoint_bpf.o"

#
============================================================================
# Session & Resource Pool Configuration
#
```

```
# ============================================================================
max_sessions = 100_000
teid_pool_start = 1
teid_pool_end = 10_000_000
far_map_size = 131_070
qer_map_size = 65_535
urr_map_size = 65_535

# ============================================================================
# Feature Flags
# ============================================================================
feature_ueip = true
feature_ftup = true
ueip_pool = "10.45.0.0/16"

# ============================================================================
# Route Management
# ============================================================================
route_manager_enabled = true
route_manager_type = "frr"

# ============================================================================
# Heartbeat Configuration
# ============================================================================
heartbeat_interval_ms = 10_000
heartbeat_timeout_ms = 5_000
heartbeat_retries = 5

# ============================================================================
# GTP-U Peer Echo Monitoring
# ============================================================================
gtp_echo_interval = 10
gtp_echo_retries = 3
gtp_peers = [
  %{address: parse_ip.("10.100.50.50"), echo: true, echo_interval: 10
]
```

```
#
# ============================================================================
# HTTP API and Logging
#
# ============================================================================
api_port = 8080
log_level = :info

#
# ============================================================================
# Apply Configuration (do not edit below this line)
#
# ============================================================================
config :upf_ex,
  pfcp_address: parse_ip.(pfcp_address),
  pfcp_port: pfcp_port,
  n3_address: parse_ip.(n3_address),
  n9_address: parse_ip.(n9_address),
  node_id: parse_ip.(node_id),
  api_port: api_port,
  ebpf_pin_path: ebpf_pin_path,
  xdp_obj_path: xdp_obj_path,
  interface_names: String.split(xdp_interfaces, ","),
  xdp_attach_mode: xdp_attach_mode,
  feature_ueip: feature_ueip,
  feature_ftup: feature_ftup,
  ueip_pool: ueip_pool,
  teid_pool_start: teid_pool_start,
  teid_pool_end: teid_pool_end,
  far_map_size: far_map_size,
  qer_map_size: qer_map_size,
  urr_map_size: urr_map_size,
  max_sessions: max_sessions,
  route_manager_enabled: route_manager_enabled,
  route_manager_type: route_manager_type,
  buffer_port: buffer_port,
  gtpu_port: gtpu_port,
  heartbeat_interval_ms: heartbeat_interval_ms,
  heartbeat_timeout_ms: heartbeat_timeout_ms,
  heartbeat_retries: heartbeat_retries,
  gtp_echo_interval: gtp_echo_interval,
  gtp_echo_retries: gtp_echo_retries,
  gtp_peers: gtp_peers
```

```
config :logger, level: log_level
```

إدارة الخدمة

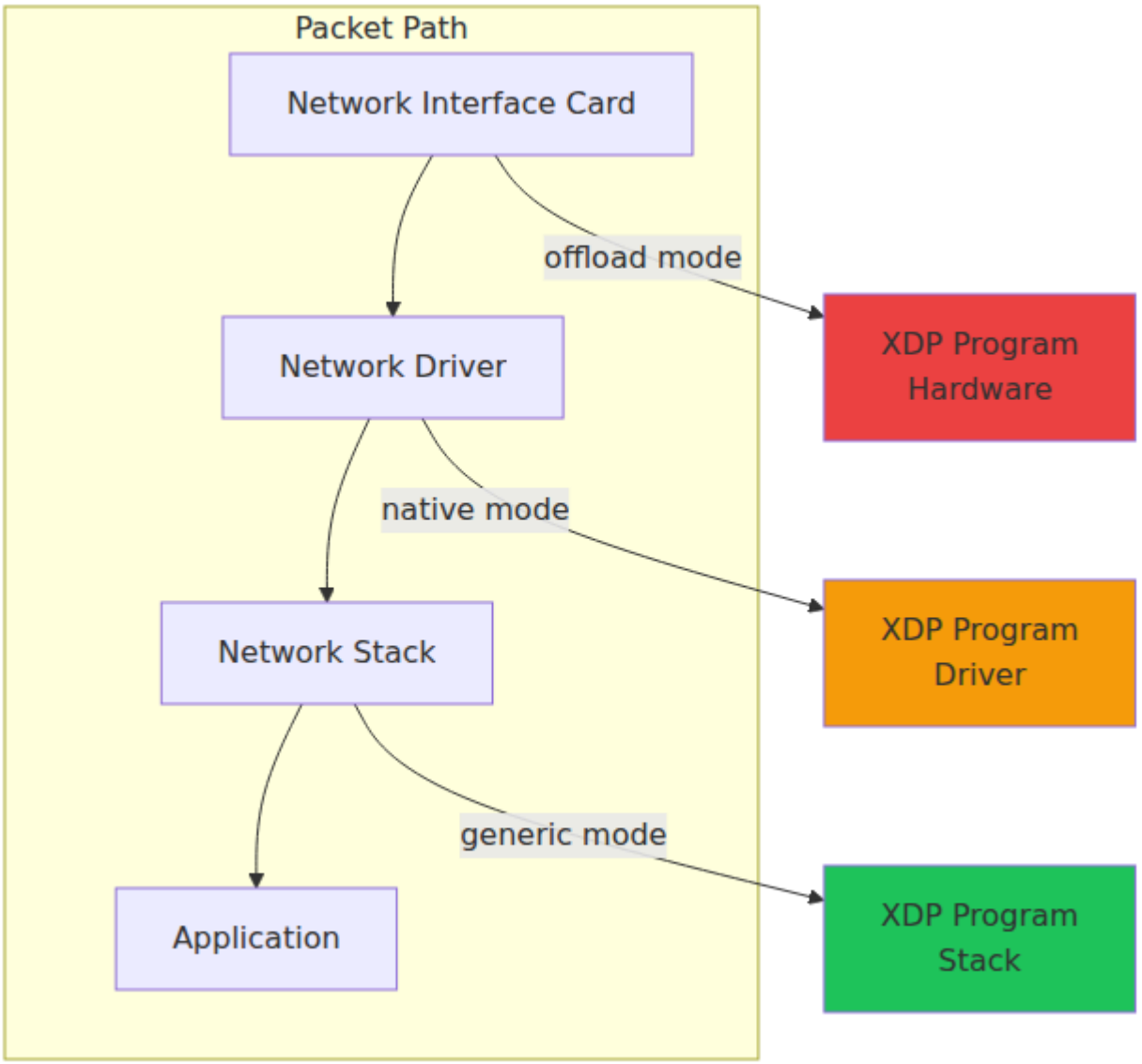
```
# بدء UPF
sudo systemctl start omniupf

# إيقاف UPF
sudo systemctl stop omniupf

# إعادة التشغيل بعد تغييرات التكوين
sudo systemctl restart omniupf

# التحقق من الحالة
sudo systemctl status omniupf

# عرض السجلات
sudo journalctl -u omniupf -f

# أو استخدم الثنائي مباشرة
sudo /opt/omniupf/bin/upf_ex start
sudo /opt/omniupf/bin/upf_ex stop
sudo /opt/omniupf/bin/upf_ex daemon   # بدء كخدمة في الخلفية
```

توافق المحاكيات

نظرة عامة

تتوافق OmniUPF مع جميع المحاكيات الرئيسية ومنصات الافتراضية. يعتمد وضع إرفاق XDP وتكوين الشبكة على قدرات الشبكة الخاصة بالمحاكي.

للحصول على تعليمات خطوة بخطوة حول تمكين XDP المحلي على Proxmox والمحاكيات الأخرى، راجع دليل أوضاع XDP.



Proxmox VE

التكوينات المدعومة:

1. وضع الجسر (XDP العام)

حالة الاستخدام: الشبكات القياسية للآلة الافتراضية

التكوين:

- جهاز الشبكة: VirtIO أو E1000
- وضع XDP: `generic`
- الأداء: ~1-2 Mpps

إعدادات VM Proxmox:

```
جهاز الشبكة: net0
النموذج: VirtIO (مُعزز)
الجسر: vmbr0
```

تكوين OmniUPF (في `runtime.exs`):

```
xdp_interfaces = "eth0"
xdp_attach_mode = "generic"
```

---

2. تمرير SR-IOV (XDP المحلي)

حالة الاستخدام: إنتاج عالي الأداء

التكوين:

- جهاز الشبكة: وظيفة افتراضية SR-IOV
- وضع XDP: `native`
- الأداء: ~5-10 Mpps

المتطلبات:

- NIC مادي يدعم SR-IOV (Intel X710، Mellanox ConnectX-5)
- تم تمكين SR-IOV في BIOS
- تم تمكين IOMMU (`intel_iommu=on` أو `amd_iommu=on` في GRUB)

تمكين SR-IOV على Proxmox:

```
# تحرير تكوين GRUB
nano /etc/default/grub

# إضافة إلى GRUB_CMDLINE_LINUX_DEFAULT:
intel_iommu=on iommu=pt

# تحديث GRUB وإعادة التشغيل
update-grub
reboot

# تمكين VFs على NIC (مثال: 4 وظائف افتراضية على eth0)
echo 4 > /sys/class/net/eth0/device/sriov_numvfs

# جعلها دائمة
echo "echo 4 > /sys/class/net/eth0/device/sriov_numvfs" >>
/etc/rc.local
chmod +x /etc/rc.local
```

إعدادات VM Proxmox:

```
جهاز -> إضافة -> الأجهزة PCI
اختر: وظيفة افتراضية SR-IOV
جميع الوظائف: لا
الأساسي: لا GPU
PCI-Express: نعم (اختياري)
```

تكوين OmniUPF (في `runtime.exs`):

```
xdp_interfaces = "ens1f0"    # اسم VF لـ SR-IOV
xdp_attach_mode = "native"
```

---

3. تمرير PCI (XDP المحلي)

حالة الاستخدام: NIC مخصص لآلة افتراضية واحدة

التكوين:

- تمرير NIC مادي بالكامل إلى VM

- وضع XDP: `native` أو `offload` (إذا كان SmartNIC)
- الأداء: ~5-40 Mpps (يعتمد على NIC)

إعدادات VM Proxmox:

```
جهاز -> إضافة -> الأجهزة PCI
اختر: NIC (0000:01:00.0 مثل) مادي
جميع الوظائف: نعم
الأساسي: لا GPU
PCI-Express: نعم
```

تكوين OmniUPF (في `runtime.exs`):

```
xdp_interfaces = "ens1f0"
xdp_attach_mode = "native"    # أو "offload" لـ SmartNIC
```

KVM/QEMU

وضع الجسر:

```
virt-install \
  --name omniupf \
  --network bridge=br0,model=virtio \
  --disk path=/var/lib/libvirt/images/omniupf.qcow2 \
  ...
```

تمرير SR-IOV:

```
<interface type='hostdev' managed='yes'>
  <source>
    <address type='pci' domain='0x0000' bus='0x01' slot='0x10'
function='0x1'/>
  </source>
</interface>
```

VMware ESXi

vSwitch القياسي (XDP العام):

- محول الشبكة: VMXNET3
- وضع XDP: `generic`

SR-IOV (XDP المحلي):

- تمكين SR-IOV في إعدادات مضيف ESXi
- إضافة محول شبكة SR-IOV إلى VM
- وضع XDP: `native`

Microsoft Hyper-V

مفتاح افتراضي (XDP العام):

- محول الشبكة: صناعي
- وضع XDP: `generic`

SR-IOV (XDP المحلي):

- تمكين SR-IOV في Hyper-V Manager
- تكوين SR-IOV على محول الشبكة الافتراضي
- وضع XDP: `native`

VirtualBox

وضع NAT/الجسر (XDP العام فقط):

- محول الشبكة: VirtIO-Net أو Intel PRO/1000
- وضع XDP: `generic`
- ملاحظة: VirtualBox لا يدعم **SR-IOV**

توافق NIC

فهم Mpps مقابل السعة

ليست متساوية مباشرة - العلاقة تعتمد **السعة (Gbps) والحزم في الثانية (Mpps)** تمامًا على حجم الحزمة. تختلف حركة مرور الشبكة المحمولة بشكل كبير في حجم الحزمة، من حزم VoIP الصغيرة إلى إطارات بث الفيديو الكبيرة.

تأثير حجم الحزمة على السعة

في الشبكات المحمولة، يقوم UPF بمعالجة الحزم المغلفة بـ GTP-U على واجهة N3 وحزم IP الأصلية على واجهة N6.

عبء تغليف GTP-U (واجهة N3):

- **رأس IPv4 الخارجي**: 20 بايت
- **رأس UDP الخارجي**: 8 بايت
- **رأس GTP-U**: 8 بايت
- **إجمالي عبء GTP-U**: 36 بايت

أدنى حزمة GTP-U (N3):

- **رأس IP الداخلي** (IPv4): 20 بايت
- **رأس UDP الداخلي**: 8 بايت
- **الحمولة الدنيا**: 1 بايت
- **إجمالي الحزمة الداخلية**: 29 بايت
- **بالإضافة إلى عبء GTP-U**: 36 بايت
- **إجمالي حجم الحزمة**: 65 بايت

السعة عند 1 Mpps مع أدنى حزم GTP-U:

```
65 bytes x 1,000,000 pps x 8 bits/byte = 520 Mbps
```

أقصى حزمة GTP-U (N3 مع MTU 1500):

- **MTU IP الداخلي**: 1500 بايت (حزمة IP الداخلية كاملة)
- **بالإضافة إلى عبء GTP-U**: 36 بايت

- **إجمالي حجم الحزمة**: 1536 بايت

1 السعة عند Mpps مع أقصى حزم GTP-U:

```
1536 bytes x 1,000,000 pps x 8 bits/byte = 12,288 Mbps ~ 12.3 Gbps
```

حزم IP الأصلية (واجهة N6):

على N6 (نحو الإنترنت)، تكون الحزم IP أصلية بدون GTP-U:

أدنى حزمة N6:

- **رأس IP**: 20 بايت
- **رأس UDP**: 8 بايت
- **الحمولة الدنيا**: 1 بايت
- **الإجمالي**: 29 بايت

1 السعة عند Mpps مع أدنى حزم N6:

```
29 bytes x 1,000,000 pps x 8 bits/byte = 232 Mbps
```

أقصى حزمة N6 (MTU 1500):

- **MTU IP**: 1500 بايت
- **الإجمالي**: 1500 بايت

1 السعة عند Mpps مع أقصى حزم N6:

```
1500 bytes x 1,000,000 pps x 8 bits/byte = 12,000 Mbps = 12 Gbps
```

أمثلة الأداء في العالم الحقيقي

NIC Intel X710 (10 سعة Mpps على واجهة N3 مع GTP-U):

نمط الحركة	حجم الحزمة الداخلية	إجمالي GTP-U	السعة عند 10 Mpps	حالة الاستخدام النموذجية
مكالمات VoIP (N3)	65-150 بايت	101-186 بايت	**0.8-1.5 Gbps**	صوت AMR-WB، G.711
ويب خفيف (N3)	400-600 بايت	436-636 بايت	**3.5-5.1 Gbps**	HTTP/HTTPS، رسائل
محمول حديث (N3)	**1200 بايت**	**1236 بايت**	**9.9 Gbps**	**نمط حركة مرور 2024 النموذجي**
بث الفيديو (N3)	1400-1450 بايت	1436-1486 بايت	**11.5-11.9 Gbps**	قطع فيديو HD/4K
أقصى MTU (N3)	1500 بايت	1536 بايت	**12.3 Gbps**	تنزيلات TCP كبيرة

على واجهة N6 (IP أصلية، بدون GTP-U):

نمط الحركة	حجم الحزمة	السعة عند 10 Mpps	حالة الاستخدام النموذجية
حزم VoIP	65-150 بايت	**0.5-1.2 Gbps**	تدفقات صوتية RTP
ويب خفيف	400-600 بايت	**3.2-4.8 Gbps**	طلبات HTTP
محمول حديث	**1200 بايت**	**9.6 Gbps**	**حركة مرور 2024 النموذجية**
بث الفيديو	1400-1450 بايت	**11.2-11.6 Gbps**	تنزيلات الفيديو
أقصى MTU	1500 بايت	**12.0 Gbps**	نقل ملفات كبيرة

مع حركة مرور محمولة حد��ثة (متوسط 1200 بايت)، توقع **~10 Mpps عند 10 Gbps** **سعة** على كل من واجهتي N3 و N6.

تخطيط السعة العملي:

مع حجم حزمة متوسط 1200 بايت (نموذجي لشبكات المحمول الحديثة مع بث الفيديو):

سعة NIC Mpps	سعة N3 (GTP-U)	سعة N6 (IP الأصلية)	نشر واقعي
1 Mpps	~1.0 Gbps	~1.0 Gbps	موقع خلية صغيرة، بوابة IoT
5 Mpps	~4.9 Gbps	~4.8 Gbps	موقع خلية متوسطة، مؤسسة
10 Mpps	~9.9 Gbps	~9.6 Gbps	موقع خلية كبيرة، مدينة صغيرة
20 Mpps	~19.7 Gbps	~19.2 Gbps	منطقة حضرية، مدينة متوسطة
40 Mpps	~39.4 Gbps	~38.4 Gbps	منطقة حضرية كبيرة، مركز إقليمي

برامج تشغيل الشبكة القابلة لـ XDP

تتطلب OmniUPF برامج تشغيل الشبكة التي تدعم XDP. يعمل **التحميل** و **المحلي** لوضعي الوضع العام مع **أي** NIC.

NICs Intel

النموذج	السائق	دعم XDP	الوضع	الأداء
Intel X710	i40e	نعم	محلي	~10 Mpps
Intel XL710	i40e	نعم	محلي	~10 Mpps
Intel E810	ice	نعم	محلي	~15 Mpps
Intel 82599ES	ixgbe	نعم	محلي	~8 Mpps
Intel I350	igb	محدود	عام	~1 Mpps
Intel E1000	e1000	لا	عام فقط	~1 Mpps

NICs Mellanox/NVIDIA

النموذج	السائق	دعم XDP	الوضع	الأداء
Mellanox ConnectX-5	mlx5	نعم	محلي	~12 Mpps
Mellanox ConnectX-6	mlx5	نعم	محلي	~20 Mpps
Mellanox BlueField	mlx5	نعم	محلي + تحميل	~40 Mpps
Mellanox ConnectX-4	mlx4	محدود	عام	~2 Mpps

NICs Broadcom

النموذج	السائق	دعم XDP	الوضع	الأداء
Broadcom BCM57xxx	bnxt_en	نعم	محلي	~8 Mpps
Broadcom NetXtreme II	bnx2x	لا	عام فقط	~1 Mpps

بائعون آخرون

النموذج	السائق	دعم XDP	الوضع	الأداء
Netronome Agilio CX	nfp	نعم	تحميل	~30 Mpps
Amazon ENA	ena	نعم	محلي	~5 Mpps
Solarflare SFC9xxx	sfc	نعم	محلي	~8 Mpps
VirtIO	virtio_net	محدود	عام	~2 Mpps

التحقق من دعم XDP لـ NIC

تحقق مما إذا كان السائق يدعم XDP:

```
# العثور على سائق NIC
ethtool -i eth0 | grep driver

# تحقق من دعم XDP في السائق
modinfo <driver_name> | grep -i xdp

# مثال لـ Intel i40e
modinfo i40e | grep -i xdp
```

تحقق من إرفاق برنامج XDP:

```
# تحقق مما إذا كان برنامج XDP ملحقًا
ip link show eth0 | grep -i xdp

# مثال على الإخراج (XDP ملحق):
# 2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 xdp qdisc mq
```

NICs الموصى بها حسب حالة الاستخدام

(حركة مرور محمولة حديثة) مع حجم حزمة متوسط 1200 بايت:

سيناريو النشر	السعة (N3)	سعة Mpps	الوضع	NIC الموصى بها	حالة الاستخدام
اختبار المختبر، PoC	1-2 Gbps	1-2 Mpps	عام	أي NIC (VirtIO، E1000)	**الاختبار/ التطوير**
خلية ريفية، مؤسسة	5-10 Gbps	5-10 Mpps	محلي	Intel X710، Mellanox CX-5	**موقع خلية صغيرة**
خلية حضرية، مدينة صغيرة	10-20 Gbps	10-20 Mpps	محلي	Intel E810، Mellanox CX-6	**خلية حضرية/ متوسطة**
منطقة حضرية، مدينة متوسطة	20-40 Gbps	20-40 Mpps	محلي	Mellanox CX-6، Intel E810 (ثنائي)	**منطقة حضرية كبيرة**
تجميع إقليمي	40+ Gbps	40+ Mpps	تحميل	Mellanox BlueField، Netronome Agilio	**مركز إقليمي**
للاختبار فقط	1-2 Gbps	1-2 Mpps	عام	VirtIO	**VM Proxmox (الجسر)**
الإنتاج VM	5-10 Gbps	5-10 Mpps	محلي	Intel X710/E810 VF، Mellanox CX-5 VF	**VM Proxmox (SR-IOV)**

موارد إضافية

وثائق XDP الرسمية:

- مشروع XDP
- وثائق XDP في النواة

قوائم توافق NIC:

- قائمة دعم XDP لـ Cilium
- سائقو XDP لـ IO Visor

أمثلة التكوين

المثال 1: بيئة تطوير (الوضع العام)

السيناريو: اختبار OmniUPF على جهاز ��مبيوتر محمول أو VM بدون SR-IOV

```
# تكوين التطوير (/etc/omniupf/runtime.exs)
xdp_interfaces = "eth0"
xdp_attach_mode = "generic"
api_port = 8080
pfcp_address = "127.0.0.1"
pfcp_port = 8805
node_id = "127.0.0.1"
n3_address = "127.0.0.1"
log_level = :debug
max_sessions = 1_000
```

المثال 2: الإنتاج على العتاد الفعلي (الوضع المحلي)

السيناريو: UPF الإنتاج على خادم العتاد الفعلي مع NIC Intel X710

```
# تكوين الإنتاج على العتاد الفعلي (/etc/omniupf/runtime.exs)
xdp_interfaces = "ens1f0,ens1f1"    # N3 على ens1f0، N6 على ens1f1
xdp_attach_mode = "native"
api_port = 8080
pfcp_address = "10.100.50.241"
pfcp_port = 8805
node_id = "10.100.50.241"
n3_address = "10.100.50.233"
n9_address = "10.100.50.234"
log_level = :info
max_sessions = 500_000
gtp_echo_interval = 30
gtp_peers = [
  %{address: parse_ip.("10.100.50.10"), echo: true, echo_interval:
30},
  %{address: parse_ip.("10.100.50.11"), echo: true, echo_interval:
30},
]
heartbeat_interval_ms = 10_000
feature_ueip = true
ueip_pool = "10.45.0.0/16"
```

المثال 3: VM Proxmox مع SR-IOV (الوضع المحلي)

السيناريو: UPF الإنتاج على VM Proxmox مع تمرير SR-IOV

```
# تكوين Proxmox SR-IOV (/etc/omniupf/runtime.exs)
xdp_interfaces = "ens1f0"           # VF لـ SR-IOV
xdp_attach_mode = "native"
api_port = 8080
pfcp_address = "192.168.100.10"
pfcp_port = 8805
node_id = "192.168.100.10"
n3_address = "192.168.100.10"
log_level = :info
max_sessions = 100_000
gtp_echo_interval = 15
gtp_peers = [
  %{address: parse_ip.("192.168.100.50"), echo: true},
]
```

المثال 4: وضع PGW-U (4G EPC)

السيناريو: OmniUPF تعمل كـ PGW-U في شبكة EPC 4G

```
# تكوين PGW-U (/etc/omniupf/runtime.exs)
xdp_interfaces = "eth0"
xdp_attach_mode = "native"
api_port = 8080
pfcp_address = "10.200.1.10"
pfcp_port = 8805
node_id = "10.200.1.10"
n3_address = "10.200.1.10"          # واجهة S5/S8 (GTP-U)
log_level = :info
max_sessions = 200_000
gtp_echo_interval = 20
gtp_peers = [
  %{address: parse_ip.("10.200.1.50"), echo: true, echo_interval:
20},
]
heartbeat_interval_ms = 5_000
```

المثال 5: وضع متعدد (UPF + PGW-U في نفس الوقت)

السيناريو: OmniUPF 5 تخدم كل من الشبكات G 4 وG 5 في وقت واحد

```
# تكوين الوضع المتعدد (/etc/omniupf/runtime.exs)
xdp_interfaces = "eth0,eth1"
xdp_attach_mode = "native"
api_port = 8080
pfcp_address = "10.50.1.100"
pfcp_port = 8805
node_id = "10.50.1.100"
n3_address = "10.50.1.100"
n9_address = "10.50.1.101"
log_level = :info
max_sessions = 300_000
gtp_echo_interval = 15
gtp_peers = [
  %{address: parse_ip.("10.50.2.10"), echo: true, echo_interval:
15},
  %{address: parse_ip.("10.50.2.20"), echo: true, echo_interval:
15},
]
heartbeat_interval_ms = 10_000
feature_ueip = true
ueip_pool = "10.60.0.0/16"
```

المثال 6: وضع تحميل SmartNIC

السيناريو: نشر عالي السعة للغاية مع SmartNIC Netronome Agilio CX

```
# تكوين SmartNIC للتحميل (/etc/omniupf/runtime.exs)
xdp_interfaces = "enp1s0np0"        # واجهة SmartNIC
xdp_attach_mode = "offload"
api_port = 8080
pfcp_address = "10.10.1.50"
pfcp_port = 8805
node_id = "10.10.1.50"
n3_address = "10.10.1.50"
log_level = :warning
max_sessions = 1_000_000
far_map_size = 2_000_000
qer_map_size = 1_000_000
gtp_echo_interval = 30
gtp_peers = [
  %{address: parse_ip.("10.10.2.10"), echo: true},
  %{address: parse_ip.("10.10.2.20"), echo: true},
  %{address: parse_ip.("10.10.2.30"), echo: true},
]
heartbeat_interval_ms = 15_000
```

تخطيط سعة الخريطة وحجمها

ت��يين أحجام الخريطة

قم بتعيين `max_sessions` الخاص بك وتكوين أحجام الخريطة لنشر:

```
max_sessions = 100_000
far_map_size = 131_070
qer_map_size = 65_535
urr_map_size = 65_535
```

استخدام الذاكرة: ~91 ميجابايت لـ 100K جلسات

تقدير السعة

احسب الحد الأقصى للجلسات:

```
Max Sessions = min(
  pdr_map_size / 2,
  far_map_size / 2,
  qer_map_size
)
```

مثال:

- خريطة PDR: 200,000
- خريطة FAR: 200,000
- خريطة QER: 100,000

Max Sessions = min(100,000, 100,000, 100,000) = **100,000**

متطلبات الذاكرة

استخدام الذاكرة لكل جلسة:

- PDR: 2 x 212 B = 424 B
- FAR: 2 x 20 B = 40 B
- QER: 1 x 36 B = 36 B
- URR: 2 x 20 B = 40 B
- 540~ :**الإجمالي** B لكل جلسة

جلسات: ~52 ميجابايت من ذاكرة النواة **100 لـ K**

التوصية: تأكد من أن حد الذاكرة المقفلة يسمح باستخدام 2 المقدرة x:

```
# تحقق من الحد الحالي
ulimit -l

# تعيين غير محدود (مطلوب لـ eBPF)
ulimit -l unlimited
```

الوثائق ذات الصلة

- **دليل العمارة** - تفاصيل تقنية eBPF/XDP وتحسين الأداء
- **دليل إدارة القواعد** - تكوين URR و QER و FAR و PDR
- **دليل المراقبة** - الإحصائيات، مراقبة السعة، والتنبيه
- **مرجع المقاييس** - مرجع كامل لمقاييس Prometheus
- **دليل واجهة الويب** - عمليات لوحة التحكم
- **دليل العمليات** - نظرة عامة على عمارة UPF والنشر
- **دليل الحديقة المسورة** - إعادة التوجيه خارج الائتمان، بوابة الأسر، وتكوين القائمة البيضاء

مرجع المقاييس

تصف هذه الوثيقة جميع مقاييس Prometheus المعروضة بواسطة OmniUPF على نقطة النهاية `/metrics`.

فئات المقاييس

1. **مقاييس رسائل PFCP** - عدادات رسائل بروتوكول التحكم ووقت الاستجابة لكل نظير
2. **مقاييس إجراءات XDP** - أحكام حزم البيانات (إسقاط، تمرير، إعادة توجيه، إلخ)
3. **مقاييس الحزم** - عدادات الحزم المستلمة حسب نوع البروتوكول
4. **مقاييس جلسات PFCP والارتباطات** - عدد الجلسات والارتباطات لكل نظير
5. **مقاييس URR** - عدادات حجم الحركة مجمعة لكل نظير PFCP
6. **مقاييس تخزين الحزم** - حالة تخزين الحزم، السعة، ومعدل النقل
7. **مقاييس تقرير بيانات التنزيل (الإشعار)** - إشعارات طلب تقرير جلسة PFCP وتتبع فهرس FAR
8. **مقاييس سعة خريطة eBPF** - استخدام خريطة eBPF والسعة

مرجع المقاييس

مقاييس رسائل PFCP

مقاييس لتتبع رسائل بروتوكول PFCP بين UPF وعقد التحكم.

اسم المقياس	النوع	التسميات	الوصف
upf_pfcp_rx	عداد	`message_name`, `peer_address`	العدد الإجمالي لرسائل PFCP المستلمة حسب نوع الرسالة والنظير
upf_pfcp_tx	عداد	`message_name`, `peer_address`	العدد الإجمالي لرسائل PFCP المرسلة حسب نوع الرسالة والنظير
upf_pfcp_rx_errors	عداد	`message_name`, `cause_code`, `peer_address`	العدد الإجمالي لرسائل PFCP المرفوضة بسبب خطأ لكل نوع رسالة ونظير
upf_pfcp_rx_latency	ملخص	`message_type`, `peer_address`	مدة معالجة رسالة PFCP بالميكروثانية (p50، p90، p99) لكل نوع رسالة ونظير

ملاحظة: تتبع جميع العدادات الرسائل لكل نظير PFCP لرؤية دقيقة في سلوك عقد التحكم.

مقاييس إجراءات XDP

تتبع هذه المقاييس القرار في خط البيانات لكل XDP. عدادات الحزم حسب إجراء/حكم برنامج حزمة.

اسم المقياس	النوع	التسميات	الوصف
upf_xdp_aborted	عداد	لا شيء	العدد الإجمالي للحزم المرفوضة (XDP_ABORTED)
upf_xdp_drop	عداد	لا شيء	العدد الإجمالي للحزم المرفوضة (XDP_DROP)
upf_xdp_pass	عداد	لا شيء	العدد الإجمالي للحزم المرسلة إلى النواة (XDP_PASS)
upf_xdp_tx	عداد	لا شيء	العدد الإجمالي للحزم المرسلة (XDP_TX)
upf_xdp_redirect	عداد	لا شيء	العدد الإجمالي للحزم المعاد توجيهها (XDP_REDIRECT)

مقاييس الحزم

عدادات للحزم المستلمة حسب نوع البروتوكول. تستخدم جميع المقاييس تسمية `packet_type`.

اسم المقيا�المقيا	النوع	التسميات	الوصف
upf_rx	عداد	`packet_type`	العدد الإجمالي للحزم المستلمة حسب النوع
upf_route	عداد	`packet_type`	العدد الإجمالي للحزم الموجهة حسب نتيجة البحث

قيم packet_type لـ upf_rx:

- `arp` - حزم ARP
- `icmp` - حزم ICMP
- `icmp6` - حزم ICMPv6
- `ip4` - حزم IPv4
- `ip6` - حزم IPv6

- `tcp` - حزم TCP
- `udp` - حزم UDP
- `other` - أنواع الحزم الأخرى
- `gtp-echo` - طلب/استجابة GTP echo
- `gtp-pdu` - GTP-U PDU (بيانات المستخدم المغلفة)
- `gtp-other` - أنواع رسائل GTP الأخرى
- `gtp-unexp` - حزم GTP غير متوقعة/معطوبة

قيم packet_type لـ upf_route:

- `ip4-cache` - نجاحات ذاكرة تخزين IPv4
- `ip4-ok` - نجاح بحث FIB لـ IPv4
- `ip4-error-drop` - فشل بحث FIB لـ IPv4، تم إسقاط الحزمة
- `ip4-error-pass` - فشل بحث FIB لـ IPv4، تم تمرير الحزمة إلى النواة
- `ip6-cache` - نجاحات ذاكرة تخزين IPv6
- `ip6-ok` - نجاح بحث FIB لـ IPv6
- `ip6-error-drop` - فشل بحث FIB لـ IPv6، تم إسقاط الحزمة
- `ip6-error-pass` - فشل بحث FIB لـ IPv6، تم تمرير الحزمة إلى النواة

مقاييس جلسات PFCP والارتباطات

مقاييس لتتبع جلسات PFCP والارتباطات بين UPF وعقد التحكم.

اسم المقياس	النوع	التسميات	الوصف
upf_pfcp_sessions	مقياس	لا شيء	العدد الإجمالي لجلسات PFCP الحالية (جميع النظائر)
upf_pfcp_associations	مقياس	لا شيء	العدد الإجمالي للارتباطات PFCP الحالية لجلسات (جميع النظائر)
upf_pfcp_association_status	مقياس	`node_id`, `address`	حالة ارتباط PFCP لكل نظير (1=مفعل، 0=معطل)
upf_pfcp_sessions_per_node	مقياس	`node_id`, `address`	عدد جلسات PFCP النشطة لكل عقدة في خط التحكم

مقاييس URR (قاعدة تقارير الاستخدام)

مقاييس حجم الحركة مجمعة لكل نظير PFCP. يمثل حجم كل نظير مجموع جميع عدادات URR عبر جميع الجلسات من تلك العقدة في خط التحكم.

اسم المقياس	النوع	التسميات	الوصف
upf_urr_uplink_volume_bytes	مقياس	`peer_address`	إجمالي حجم حركة البيانات في الاتجاه الصاعد بالبايت لجميع الجلسات من هذا النظير
upf_urr_downlink_volume_bytes	مقياس	`peer_address`	إجمالي حجم حركة البيانات في الاتجاه النازل بالبايت لجميع الجلسات من هذا النظير
upf_urr_total_volume_bytes	مقياس	`peer_address`	إجمالي حجم الحركة بالبايت (الرفع + النزول) لجميع الجلسات من هذا النظير

ملاحظة: يتم تجميع الأحجام لكل نظير PFCP لتجنب مشاكل الكاردينالية العالية. الإحصائيات الفردية لـ URR متاحة عبر واجهة برمجة التطبيقات REST على `/api/v1/urr_map`.

مقاييس تخزين الحزم

مقاييس UPF لتتبع حالة وأداء تخزين الحزم. يمكن لـ تخزين الحزم في الاتجاه النازل عندما يكون UE في حالة الخمول، محتفظًا بها حتى يتم استدعاء UE وينتقل إلى حالة الاتصال.

اسم المقياس	النوع	التسميات	ف
upf_buffer_packets_total	عداد	لا شيء	عدد الي حزم افة إلى زين مر من)
upf_buffer_packets_dropped	عداد	`reason`	عدد الي حزم وضة من زين
upf_buffer_packets_flushed	عداد	لا شيء	عدد الي حزم تم يغها من زين
upf_buffer_packets_current	مقياس	لا شيء	عدد الي في زين
upf_buffer_bytes_total	عداد	لا شيء	الي تات افة إلى زين

اسم المقياس	النوع	التسميات	ف
			ر من)
upf_buffer_bytes_current	مقياس	لا شيء	تات الية في زين
upf_buffer_fars_active	مقياس	لا شيء	عدد ي لـ FAR حزم خزنة
upf_buffer_listener_packets_received_total	عداد	لا شيء	عدد الي حزم تلمة من تمع زين خط نات eBF
upf_buffer_listener_packets_buffered_total	عداد	لا شيء	عدد الي حزم ي تم زينها جاح بطة تمع
upf_buffer_listener_errors_total	عداد	`type`	طاء في

اسم المقياس	النوع	التسميات	ف
			لجة حزم تمع زين
upf_buffer_listener_error_indications_sent_total	عداد	`remote_peer`	عدد الي سائل سارة خطأ GTF سلة حديد TEI غير وفة بطة ظير بعيد
upf_buffer_flush_success_total	عداد	لا شيء	عدد الي يات فريغ زين جحة
upf_buffer_flush_errors_total	عداد	`reason`	عدد الي يات فريغ زين شلة
upf_buffer_flush_packets_sent_total	عداد	لا شيء	عدد الي

اسم المقياس	النوع	التسميات	ف
			حزم سلة أثناء يات فريغ

قيم reason لـ upf_buffer_packets_dropped:

- `expired` - حزم تم إسقاطها بسبب انتهاء صلاحية TTL
- `global_limit` - تم إسقاطها بسبب الوصول إلى الحد الأقصى للتخزين الكلي
- `far_limit` - تم إسقاطها بسبب الوصول إلى الحد الأقصى للتخزين لكل FAR
- `cleared` - حزم تم مسحها يدويًا من التخزين

قيم type لـ upf_buffer_listener_errors_total:

- `read_error` - خطأ في القراءة من مقبس التخزين
- `too_small` - حزمة GTP صغيرة جدًا لرأس
- `invalid_gtp_type` - نوع رسالة GTP غير G-PDU
- `unknown_teid` - لم يتم العثور على TEID لـ PDR/FAR
- `not_buffering_far` - FAR لا يحتوي على إجراء BUFF
- `truncated_ext` - رؤوس امتداد GTP مقطوعة
- `no_payload` - حزمة GTP لا تحتوي على حمولة
- `buffer_full` - تجاوز سعة التخزين

قيم reason لـ upf_buffer_flush_errors_total:

- `far_lookup_failed` - فشل في البحث عن معلومات FAR من خريطة eBPF
- `no_forw_action` - FAR لا يحتوي على إجراء FORW محدد
- `connection_failed` - فشل في إنشاء اتصال UDP للتفريغ

مقاييس تقرير بيانات التنزيل (الإشعار)

مقاييس لإشعارات طلب تقرير جلسة PFCP. المرسلة إلى خط التحكم عندما يتم تخزين الحزم. تحفز هذه الإشعارات خط التحكم لاستدعاء UE.

اسم المقياس	النوع	التسميات	صف
upf_dldr_sent_total	عداد	لا شيء	العدد جمالي عارات تقرير بيانات لتنزيل (DLD رسلة S إلى
upf_dldr_send_errors	عداد	لا شيء	العدد جمالي لأخطاء إرسال عارات تقرير بيانات لتنزيل
upf_dldr_active_notifications	مقياس	لا شيء	العدد الي لـ FARs عارات DLDR لة (لم يتم سحها بعد)
upf_far_index_size	مقياس	لا شيء	العدد الي لـ FARs سجلة في FarIn

اسم المقياس	النوع	التسميات	صف
			عارات DLDF
upf_far_index_registrations_total	عداد	لا شيء	العدد جمالي جيلات FAR ي FarIn
upf_far_index_unregistrations_total	عداد	لا شيء	العدد جمالي لإلغاء سجيل FARs FarIn
upf_buffer_notify_to_flush_duration_seconds	هيستوجرام	`pfcp_peer`	ت بين إرسال إشعار DLDF وتفريغ الحزم مخزنة

upf_buffer_notify_to_flush_duration_seconds:

- دلو الهيستوجرام: 0.01، 0.05، 0.1، 0.5، 1.0، 2.0، 5.0، 10.0، 30.0، 60.0 ثانية
- تسمية `pfcp_peer`: عنوان SMF/PGW-C (مثل، `10.100.50.241`)
- يقيس الكمون بين إرسال SMF للإشعار إلى UPF واستجابة SMF بتعديل الجلسة لتفريغ الحزم
- مفيد لمراقبة استجابة خط التحكم أثناء الانتقالات من الخمول إلى الاتصال

مقاييس إشارة خطأ GTP-U

مقاييس GTP-U لتتبع رسائل إشارة خطأ المرسلة والمستلمة. يتم إرسال إشارات الخطأ عندما يتلقى نظير حزمًا لتحديد TEIDs غير معروفة، مما يشير إلى عدم تطابق حالة النفق (غالبًا بسبب إعادة تشغيل النظير).

اسم المقياس	النوع	سميات
upf_buffer_listener_error_indications_sent_total	عداد	`node_id` `peer_ad`
upf_buffer_listener_error_indications_received_total	عداد	`node_id` `peer_ad`
upf_buffer_listener_error_indication_sessions_deleted_total	عداد	`node_id` `peer_ad`

تعريفات التسميات:

- `node_id`: معرف عقدة PFCP من الارتباط (مثل، `"pgw-u-1"`، `"smf-1"`). يتم تعيينه إلى `"unknown"` إذا لم يكن هناك ارتباط PFCP موجود لذلك النظير.
- `peer_address`: عنوان IP للنظير البعيد (مثل، `"192.168.50.10"`)

عندما يتم إرسال إشارات الخطأ:

- يتلقى UPF حزمة GTP-U لتحديد TEID غير موجود (مثل، بعد إعادة تشغيل UPF، تم حذف الجلسة بالفعل)
- يقوم المرسل (eNodeB، gNodeB، UPF العلوي) بإعادة توجيه إلى نفق قديم/محذوف
- يرسل UPF إشارة خطأ لإبلاغ المرسل بالتوقف عن الإرسال

عندما يتم استلام إشارات الخطأ:

- يقوم UPF (PGW-U، SGW-U، UPF) بإعادة توجيه حزمة GTP-U إلى نظير أسفل لتحديد TEID غير معروف
- الوجهة (مثل، أعيد تشغيل النظير وفقد حالة النفق) لا يتعرف النظير البعيد على TEID
- يقوم UPF تلقائيًا بحذف الجلسات المتأثرة للتوقف عن إعادة التوجيه إلى الأنفاق الميتة

حالات الاستخدام:

- اكتشاف إعادة تشغيل النظير (معدل إشارة الخطأ المرتفع يشير إلى فقدان الحالة)
- تحديد عدم تطابق التكوين (مشاكل تخصيص TEID)
- مراقبة صحة تزامن النفق بين عناصر الشبكة
- تنبيه بشأن حذف الجلسات غير المتوقعة

مثال PromQL استعلامات:

```
# معدل إشارات الخطأ المستلمة لكل نظير (في الثانية)
rate(upf_buffer_listener_error_indications_received_total[5m])

# العدد الإجمالي للجلسات المحذوفة بسبب إشارات الخطأ من نظير محدد
upf_buffer_listener_error_indication_sessions_deleted_total{peer_addr

# النظائر التي ترسل TEIDs غير معروفة إلى هذا UPF
sum by (node_id, peer_address) (upf_buffer_listener_error_indications
```

مقاييس سعة خريطة eBPF

تساعد هذه المقاييس في مراقبة استخدام الموارد .eBPF مقاييس لتتبع استخ��ام خريطة واكتشاف مشاكل السعة المحتملة.

اسم المقياس	النوع	التسميات	الوصف
upf_ebpf_map_capacity	مقياس	`map_name`	السعة القصوى لخريطة eBPF
upf_ebpf_map_used	مقياس	`map_name`	العدد الحالي للإدخالات في خريطة eBPF

قيم map_name الشائعة:

- `pdr_map` - خريطة قاعدة اكتشاف الحزم
- `far_map` - خريطة قاعدة إجراء التوجيه
- `qer_map` - خريطة تنفيذ QoS
- `session_map` - خريطة بحث الجلسات
- `teid_map` - خريطة TEID إلى الجلسة
- `ue_ip_map` - خريطة عنوان IP لـ UE إلى الجلسة

استخدام مقاييس Prometheus

الوصول إلى المقاييس

تُعرض المقاييس على نقطة النهاية `/metrics` في العنوان المحدد بواسطة `metrics_address` في ملف التكوين (افتراضيًا `:9090`):

```
# عرض المقاييس الخام
curl http://localhost:9090/metrics

# مثال على المخرجات
upf_pfcp_sessions 42
upf_pfcp_associations 2
upf_urr_total_volume_bytes{peer_address="10.100.50.241"}
1048576000
```

تكوين Prometheus

أضف هدف OmniUPF إلى ملف `prometheus.yml` الخاص بك:

```yaml
scrape_configs:
  - job_name: 'omniupf'
    static_configs:
      - targets: ['localhost:9090']
```

لوحات معلومات Grafana

استورد المقاييس إلى Grafana للتصور:

- عدد الجلسات والاتجاهات
- حجم الحركة لكل نظير PFCP
- معدلات معالجة الحزم
- استخدام التخزين
- مراقبة سعة خريطة eBPF

الوثائق ذات الصلة

- **دليل المراقبة** - مراقبة الإحصائيات، تخطيط السعة، والتنبيهات
- **دليل التكوين** - تكوين `metrics_address` وخيارات UPF الأخرى
- **دليل واجهة الويب** - عرض المقاييس في صفحة الإحصائيات
- **دليل الهندسة المعمارية** - مسار بيانات eBPF وتحسين الأداء
- **دليل إدارة القواعد** - فهم مقاييس PDR، FAR، QER، URR

- **دليل ستكشاف الأخطاء**- استخدام المقاييس للتشخيص

دليل المراقبة

جدول المحتويات

1. نظرة عامة
2. مراقبة الإحصائيات
3. مراقبة السعة
4. مقاييس الأداء
5. التنبيهات والحدود
6. تخطيط السعة
7. استكشاف مشكلات الأداء

نظرة عامة

تعتبر المراقبة الفعالة لـ OmniUPF أمرًا حيويًا للحفاظ على جودة الخدمة، ومنع استنفاد السعة، واستكشاف مشكلات الأداء. يوفر OmniUPF مقاييس شاملة في الوقت الحقيقي من خلال واجهة الويب الخاصة به وواجهة برمجة التطبيقات REST.

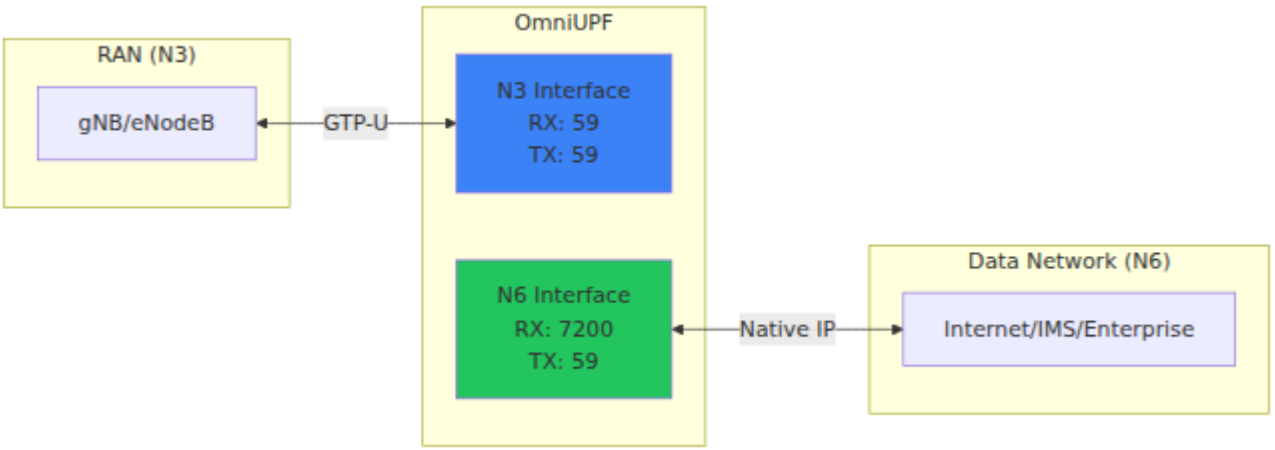
فئات المراقبة

الفئة	الغرض	تكرار التحديث	المقاييس الرئيسية
إحصائيات الحزم	تتبع معدلات معالجة الحزم والأخطاء	في الوقت الحقيقي	حزم RX/TX، السقوط، تحليل البروتوكول
إحصائيات الواجهة	مراقبة توزيع حركة المرور N3/N6	في الوقت الحقيقي	N3 RX/TX، N6 RX/TX
إحصائيات XDP	تتبع أداء مسار البيانات في النواة	في الوقت الحقيقي	XDP المعالجة، المارة، الساقطة، الملغاة
إحصائيات التوجيه	مراقبة قرارات توجيه الحزم	في الوقت الحقيقي	عمليات بحث FIB، ضربات/إخفاقات الذاكرة المؤقتة
سعة خريطة eBPF	منع استنفاد الموارد	كل 10 ثوانٍ	نسب استخدام الخريطة، المستخدمة مقابل السعة
إحصائيات المخزن المؤقت	تتبع تخزين الحزم أثناء التنقل	كل 5 ثوانٍ	حزم مخزنة، عمر المخزن المؤقت، عدد FAR

مراقبة الإحصائيات

إحصائيات واجهة N3/N6

توفر إحصائيات واجهة N3/N6 رؤية لتوزيع حركة المرور بين RAN (N3) والشبكة البيانات (N6).



المقاييس:

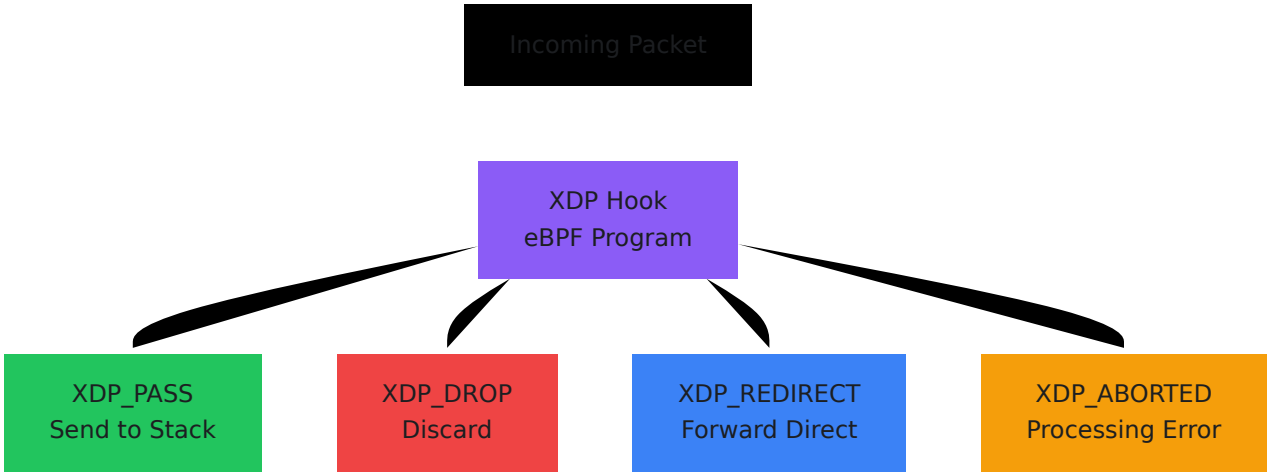
- **RX N3**: الحزم المستلمة من RAN (حركة مرور GTP-U الصاعدة)
- **TX N3**: الحزم المرسلة إلى RAN (حركة مرور GTP-U النازلة)
- **RX N6**: الحزم المستلمة من الشبكة البيانات (IP الأصلي النازل)
- **TX N6**: الحزم المرسلة إلى الشبكة البيانات (IP الأصلي الصاعد)
- **الإجمالي**: العدد الكلي للحزم عبر جميع الواجهات

السلوك المتوقع:

- **RX N3 ≈ TX N6**: تتدفق الحزم الصاعدة من RAN إلى الشبكة البيانات
- **RX N6 ≈ TX N3**: تتدفق الحزم النازلة من الشبكة البيانات إلى RAN
- قد تشير عدم التوازن الكبير إلى:
  - حركة مرور غير متكافئة (التنزيلات >> التحميلات)
  - سقوط الحزم أو أخطاء في التوجيه
  - تكوينات توجيه خ��طئة

إحصائيات XDP

تظهر إحصائيات XDP (مسار البيانات السريع) أداء معالجة الحزم على مستوى النواة.



المقاييس:

- **الملغاة** XDP (0 خطأ (يجب أن تكون دائمًا: واجه برنامج
- **السقوط** XDP: الحزم التي تم إسقاطها عمدًا بواسطة برنامج

- **المرور**: الحزم التي تم تمريرها إلى كومة الشبكة لمزيد من المعالجة
- **إعادة التوجيه**: الحزم التي تم إعادة توجيهها مباشرة إلى واجهة الإخراج
- **TX**: الحزم المرسلة عبر XDP

التفسير:

- **الملغاة > 0**: مشكلة حادة مع برنامج eBPF أو توافق النواة
- **السقوط > 0**: سقوط بناءً على السياسة أو حزم غير صالحة
- **المرور مرتفع**: تتم معالجة معظم الحزم في كومة الشبكة (طبيعي)
- **إعادة التوجيه مرتفع**: تم إعادة توجيه الحزم مباشرة (أداء مثالي)

إحصائيات الحزم

تحليل تفصيلي لبروتوكول الحزم وعدد المعالجات.

عدادات البروتوكول:

- **RX ARP**: حزم بروتوكول حل العناوين
- **RX GTP ECHO**: طلب/استجابة GTP-U Echo (keepalive)
- **RX GTP OTHER**: رسائل التحكم GTP الأخرى
- **RX GTP PDU**: بيانات المستخدم المغلفة بواسطة GTP-U (الحركة الرئيسية)
- **RX GTP UNEXP**: أنواع حزم GTP غير المتوقعة
- **RX ICMP**: بروتوكول رسائل التحكم في الإنترنت (ping، الأخطاء)
- **RX ICMP6**: حزم ICMPv6
- **RX IP4**: حزم IPv4
- **RX IP6**: حزم IPv6
- **RX OTHER**: بروتوكولات أخرى
- **RX TCP**: حزم بروتوكول التحكم في النقل
- **RX UDP**: حزم بروتوكول بيانات المستخدم

حالات الاستخدام:

- **مراقبة عدد PDU GTP-U**: مؤشر حركة ا��مستخدم الرئيسي
- **التحقق من ICMP للاتصال**: اختبار وصول الشبكة
- **تتبع نسبة TCP مقابل UDP**: أنماط حركة تطبيق

- **الكشف عن البروتوكولات غير المتوقعة**: مشكلات الأمان أو التكوين الخاطئ

إحصائيات التوجيه

إحصائيات بحث FIB (قاعدة معلومات التوجيه) لقرارات التوجيه.

بحث FIB IPv4:

- **الذاكرة المؤقتة**: عمليات بحث التوجيه المخزنة (مسار سريع)
- **OK**: عمليات بحث التوجيه الناجحة

بحث FIB IPv6:

- **الذاكرة المؤقتة**: عمليات بحث التوجيه IPv6 المخزنة
- **OK**: عمليات بحث التوجيه IPv6 الناجحة

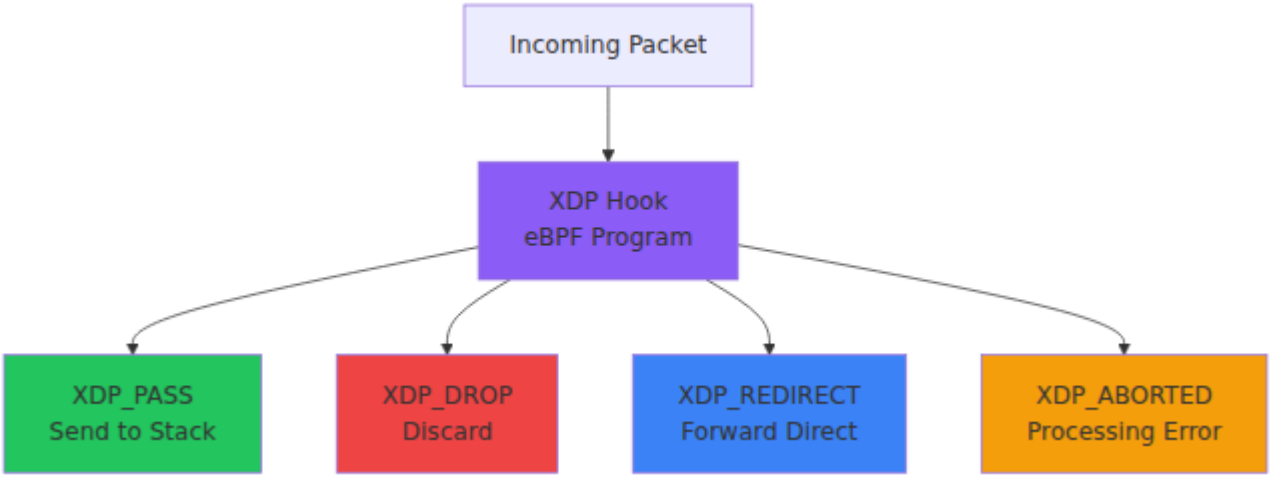
مؤشرات الأداء:

- **معدل ضرب الذاكرة المؤقتة مرتفع**: يشير إلى أداء جيد لذاكرة التوجيه
- **عدد OK مرتفع**: يؤكد أن جداول التوجيه تم تكوينها بشكل صحيح
- **عمليات بحث منخفضة أو صفر**: قد تشير إلى عدم تدفق الحركة أو تجاوز التوجيه

مراقبة السعة

سعة خريطة eBPF

مراقبة سعة خريطة eBPF تساعد في منع فشل إنشاء الجلسات بسبب استنفاد الموارد.



خرائط eBPF الحرجة

far_map (قواعد إجراء التوجيه):

- **السعة**: 131,070 إدخال
- **حجم المفتاح**: 4 ب (معرف FAR)
- **حجم القيمة**: 16 ب (معلمات التوجيه)
- **استخدام الذاكرة**: ~2.6 ميجابايت
- **الأهمية**: عالية - تستخدم لجميع قرارات توجيه الحزم

pdr_map_downlin (PDRs النازلة - IPv4):

- **السعة**: 131,070 إدخال
- **حجم المفتاح**: 4 ب (عنوان UE IPv4)
- **حجم القيمة**: 208 ب (معلومات PDR)
- **استخدام الذاكرة**: ~27 ميجابايت
- **الأهمية**: حرجة - تفشل إنشاء الجلسة إذا كانت ممتلئة

pdr_map_downlin_ip6 (PDRs النازلة - IPv6):

- **السعة**: 131,070 إدخال
- **حجم المفتاح**: 16 ب (عنوان UE IPv6)
- **حجم القيمة**: 208 ب (معلومات PDR)
- **استخدام الذاكرة**: ~29 ميجابايت
- **الأهمية**: حرجة - تفشل إنشاء جلسة IPv6 إذا كانت ممتلئة

pdr_map_teid_ip (PDRs الصاعدة):

- **السعة**: 131,070 إدخال
- **حجم المفتاح**: 4 ب (TEID)
- **حجم القيمة**: 208 ب (معلومات PDR)
- **استخدام الذاكرة**: ~27 ميجابايت
- **الأهمية**: حرجة - تفشل حركة المرور الصاعدة إذا كانت ممتلئة

qer_map (قواعد تنفيذ QoS):

- **السعة**: 65,535 إدخال
- **حجم المفتاح**: 4 ب (معرف QER)
- **حجم القيمة**: 32 ب (معلمات QoS)
- **استخدام الذاكرة**: ~2.3 ميجابايت
- **الأهمية**: متوسطة - تنفيذ QoS فقط

urr_map (قواعد تقارير الاستخدام):

- **السعة**: 131,070 إدخال
- **حجم المفتاح**: 4 ب (معرف URR)
- **حجم القيمة**: 16 ب (عدادات الحجم)
- **استخدام الذاكرة**: ~2.6 ميجابايت
- **الأهمية**: منخفضة - تؤثر على الشحن فقط

حدود السعة

الحد	الإجراء المطلوب
0-50% (أخضر)	تشغيل طبيعي - لا حاجة لإجراء
50-70% (أصفر)	حذر - راقب اتجاهات النمو، خطط لزيادة السعة
70-90% (كهرماني)	تحذير - جدولة زيادة السعة خلال أسبوع واحد
90-100% (أحمر)	حرجة - إجراء فوري مطلوب، ستفشل الجلسات الجديدة

إجراء زيادة السعة

قبل زيادة السعة:

1. مراجعة اتجاهات الاستخدام الحالية
2. تقدير معدل النمو المستقبلي

3. حساب السعة المطلوبة

خطوات زيادة سعة الخريطة:

1. إيقاف خدمة OmniUPF
2. تحديث ملف تكوين UPF الجديدة بأحجام الخرائط
3. إعادة تشغيل خدمة OmniUPF
4. التحقق من السعة الجديدة في عرض السعة
5. مراقبة إنشاء الجلسات بنجاح

ملاحظة: يتطلب تغيير سعة خريطة eBPF إعادة تشغيل UPF ويؤدي إلى مسح جميع الجلسات الحالية.

مقاييس الأداء

للحصول على معلومات تفصيلية حول جميع مقاييس Prometheus التي يكشف عنها OmniUPF، انظر **مرجع المقاييس**.

معدل معالجة الحزم

الحساب:

```
معدل الحزم (pps) = (فرق عدد الحزم) / (فرق الزمن بالثواني)
```

مثال:

- الأولية: 7,000 RX الحزم
- بعد 10 ثوانٍ: 17,000
- معدل الحزم pps = (17,000 - 7,000) / 10 = 1,000

أهداف الأداء:

- **صغير**: 10,000 - 100,000 pps **UPF**
- **متوسط**: 100,000 - 1,000,000 pps **UPF**
- **كبير**: 1,000,000 - 10,000,000 pps **UPF**

مؤشرات الاختناق:

- زيادة عدد الملغاة في XDP
- ارتفاع استخدام وحدة المعالجة المركزية
- زيادة سقوط الحزم
- زيادة الكمون

حساب الإنتاجية

الحساب:

```
الإنتاجية (Mbps) = (فرق عدد البايت × 8) / (فرق الزمن بالثواني × 1,000,000)
```

مثال:

- الأولية: 500 ميجابايت RX البايت
- بعد 60 ثانية: 800 ميجابايت
- الإنتاجية Mbps = (300 ميجابايت × 8) / (1,000,000 × 60) = 40

تخطيط السعة:

- مراقبة أوقات الذروة للإنتاجية (مثل ساعات المساء)
- مقارنة بسعة الرابط (سرعات واجهة N3/N6)
- التخطيط لزيادة السعة بمقدار 2x للإنتاجية القصوى لتوفير مساحة إضافية

معدل السقوط

لحساب��:

```
معدل السقوط (%) = (إجمالي الحزم الساقطة / الحزم RX) × 100
```

الحدود المقبولة:

- **< 0.1%**: ممتاز (فقدان الحزم الطبيعي بسبب الأخطاء)
- **0.1% - 1%**: جيد (مشكلات طفيفة أو تحديد معدل)
- **1% - 5%**: ضعيف (أو مشكلات السعة تحقيق QoS)
- **> 5%**: حرجة (مشكلة كبيرة في التوجيه أو السعة)

أسباب السقوط الشائعة:

- تحديد معدل QER (تم تجاوزه MBR)
- فشل بحث خريطة eBPF
- غير صالحة UE أو عناوين TEIDs
- أخطاء في التوجيه

التنبيهات والحدود

التنبيهات الموصى بها

التنبيهات الحرجة (تتطلب استجابة فورية):

- سعة خريطة eBPF > 90%
- عدد الملغاة في XDP > 0
- معدل السقوط% > 5
- فشل فحص صحة UPF

تنبيهات التحذير (استجابة خلال ساعة واحدة):

- سعة خريطة eBPF > 70%
- معدل السقوط% > 1
- معدل الحزم يقترب من سعة الرابط
- تجاوز TTL المخزن المؤقت (حزم أقدم من 30 ثانية)

تنبيهات معلوماتية (راقب الاتجاهات):

- سعة خريط eBPF > 50%
- زيادة عدد الحزم المخزنة
- جديدة PFCP إنشاء/إلغاء ارتباطات

- تجاوز حدود حجم URR

تكوين التنبيهات

يمكن تكوين التنبيهات عبر:

1. **مقاييس Prometheus**: تصدير المقاييس للمراقبة الخارجية (انظر مرجع المقاييس للقائمة الكاملة)
2. **مراقبة السجلات**: تحليل سجلات OmniUPF للأنماط الأخطاء
3. **استعلام REST API**: استعلام دوري لنقاط النهاية `/map_info`، `/packet_stats`
4. **مراقبة واجهة الويب**: المراقبة اليدوية عبر صفحات الإحصائيات والسعة

تخطيط السعة

تقدير سعة الجلسة

حساب الحد الأقصى للجلسات:

```
الحد الأقصى للجلسات = min(
  سعة خريطة PDR / 2,  # PDRs لكل جلسة الصاعدة + النازلة
  سعة خريطة FAR / 2,  # FARs لكل جلسة الصاعدة + النازلة
  سعة خريطة QER       # اختياري، QER واحد لكل جلسة
)
```

مثال:

- سعة خريطة PDR: 131,070
- سعة خريطة FAR: 131,070
- سعة خريطة QER: 65,535

الحد الأقصى للجلسات = min(131,070 / 2, 131,070 / 2, 65,535) = **65,535 جلسة**

سعة الذ��كرة

حساب إجمالي ذاكرة خريطة eBPF:

```
الذاكرة = Σ ((حجم المفتاح + حجم القيمة) × سعة الخريطة)
```

مثال على التكوين:

- خرائط PDR: 3 × 131,070 × 212 ميجابايت = 83.3 ب
- خريطة FAR: 131,070 × 20 ميجابايت = 2.6 ب
- خريطة QER: 65,535 × 36 ميجابايت = 2.3 ب
- خريطة URR: 131,070 × 20 ميجابايت = 2.6 ب
- **الإجمالي**: ~91 ميجابايت من ذاكرة النواة

اعتبارات ذاكرة النواة:

- التأكد من وجود حد كافٍ للذاكرة المقفلة (`ulimit -l`)
- الاستخدام المقدر لهامش الأمان xحجز 2
- مراقبة توفر ذاكرة النواة

سعة الحركة

حساب سعة الإنتاجية المطلوبة:

1. **تقدير متوسط إنتاجية الجلسة**:
   - بث الفيديو: ~5 Mbps
   - تصفح الويب: ~1 Mbps
   - VoIP: ~0.1 Mbps
2. **حساب الإنتاجية الإجمالية**:

   ```
   إجمالي الإنتاجية = الجلسات × متوسط إنتاجية الجلسة
   ```

3. **إضافة مساحة إضافية**:

   ```
   السعة المطلوبة = إجمالي الإنتاجية × 2  # 100% مساحة إضافية
   ```

مثال:

- 10,000 جلسة متزامنة
- 2 متوسط Mbps لكل جلسة
- الإجمالي: 20 جيجابت في الثانية
- السعة المطلوبة: 40 جيجابت في الثانية (N3 + N6 واجهات)

تخطيط النمو

تحليل الاتجاهات:

1. تسجيل عدد الجلسات القصوى اليومية
2. حساب معدل النمو الأسبوعي
3. تقدير السعة إلى الحد الأقصى

صيغة معدل النمو:

```
أسابيع إلى السعة = (السعة - الاستخدام الحالي) / (معدل النمو الأسبوعي)
```

مثال:

- الجلسات الحالية: 30,000
- السعة: 65,535 جلسة
- معدل النمو الأسبوعي: 2,000 جلسة
- أسابيع إلى السعة: (65,535 - 30,000) / 2,000 = **17.8 أسبوع**

الإجراء: التخطيط لترقية السعة خلال 12 أسبوعًا (مما يترك 5 أسابيع كاحتياطي).

استكشاف مشكلات الأداء

معدل سقوط الحزم المرتفع

الأعراض: معدل السقوط > 1%، شكاوى المستخدمين من ضعف الاتصال

التشخيص:

1. تحقق من الإحصائيات → إحصائيات الحزم
2. تحديد ما إذا كانت السقوط خاصة بالبروتوكول
3. مراجعة إحصائيات XDP للسقوط مقابل الملغاة

الأسباب الشائعة:

- **تحديد معدل QER**: تحقق من قيم MBR لـ QER مقابل حركة المرور الفعلية
- **TEIDs غير صالحة**: تحقق من أن TEID PDR الصاعدة تتطابق مع تعيين gNB
- **عناوين UE غير معروفة**: تحقق من وجود PDR النازلة لعنوان UE
- **تجاوز المخزن المؤقت**: تحقق من إحصائيات المخزن المؤقت

الحل:

- إذا كان هناك تحديد معدل QER لـ MBR زيادة
- تحقق من أن SMF قد أنشأ PDRs الصحيحة
- مسح المخازن المؤقتة إذا تم الكشف عن تجاوز

أخطاء معالجة XDP

الأعراض: XDP 0 < الملغاة

التشخيص:

1. إحصائيات → الانتقال إلى الإحصائيات XDP
2. تحقق من عداد الملغاة
3. مراجعة سجلات OmniUPF لأخطاء eBPF

الأسباب الشائعة:

- فشل التحقق من برنامج eBPF
- عدم توافق إصدار النواة
- أخطاء الوصول إلى خريطة eBPF
- تلف في الذاكرة

الحل:

- إعادة تشغيل خدمة OmniUPF

- تحقق من أن إصدار النواة يلبي الحد الأدنى من المتطلبات (Linux 5.4+)
- مراجعة سجلات برنامج eBPF
- الاتصال بالدعم إذا استمرت المشكلة

استنفاد السعة

الأعراض: فشل إنشاء الجلسات، سعة الخريطة عند 100%

التشخيص:

1. الانتقال إلى صفحة السعة
2. تحديد أي خريطة عند 100%
3. تحقق مما إذا كانت الجلسات عالقة (لا يتم حذفها)

التخفيف الفوري:

1. تحديد الجلسات القديمة (تحقق من صفحة الجلسات)
2. طلب SMF لحذف الجلسات القديمة
3. مسح المخازن المؤقتة لتحرير إدخالات FAR

الحل على المدى الطويل:

1. زيادة سعة خريطة eBPF
2. جدولة إعادة تشغيل UPF مع خرائط أكبر
3. تنفيذ سياسات تنظيف الجلسات

تدهور الأداء

الأعراض: ارتفاع الكمون، انخفاض الإنتاجية، تشبع وحدة المعالجة المركزية

التشخيص:

1. تحقق من معدل الحزم مقابل الخط الأساسي التاريخي
2. مراجعة إحصائيات XDP لتأخيرات المعالجة
3. مراقبة استخدام وحدة المعالجة المركزية على مضيف UPF
4. تحقق من استخدام واجهة N3/N6

الأسباب الشائعة:

- حركة المرور تتجاوز سعة UPF
- عدم كفاية نوى وحدة المعالجة المركزية لمعالجة الحزم
- اختناق واجهة الشبكة
- تصادمات تجزئة خريطة eBPF

الحل:

- توسيع UPF أفقيًا (إضافة المزيد من المثيلات)
- ترقية وحدة المعالجة المركزية أو تمكين RSS (توزيع جانب الاستقبال)
- ترقية واجهات الشبكة إلى سرعة أعلى
- ضبط وظيفة تجزئة خريطة eBPF

الوثائق ذات الصلة

- Prometheus - مرجع كامل لمقاييس **مرجع المقاييس**
- UPF الهندسة العامة وعمليات - **دليل عمليات UPF**
- PDR، FAR، QER، URR - تكوين **دليل إدارة القواعد**
- ميزات مراقبة لوحة التحكم - **دليل عمليات واجهة الويب**
- المشكلات الشائعة والتشخيصات - **دليل استكشاف الأخطاء**
- eBPF وتحسين الأداء - مسار بيانات **دليل الهندسة المعمارية**

N9 Loopback: تشغيل SGWU و PGWU على نفس المثيل

نظرة عامة

يدعم OmniUPF تشغيل كل من **SGWU (بوابة الخدمة لطبقة المستخدم)** و **PGWU (بوابة PDN المستخدم لطبقة) حلقة** مع **نفس المثيل** على **N9** وضع النشر. **بدون تأخير** هذا مثالي لـ:

- **نشر EPC 4G** - مثيل **مبسط** UPF واحد بدلاً من اثنين
- **تحسين التكاليف** - تقليل التعقيد في البنية التحتية والعمليات
- **الحوسبة الطرفية** - تقليل التأخير في سيناريوهات الانفصال المحلي
- **بيئات المختبر/الاختبار** - طبقة مستخدم EPC كاملة على خادم واحد

باكتشاف حركة المرور OmniUPF يقوم، N9 و N3 لكل من واجهتي IP عند تكوينه بنفس عنوان د**eBPF** ن إرسال حزم إلى تلقائيًا ويعالجها **تمامًا في** PGWU و SGWU المتدفقة بين أدوار واجهة الشبكة.

كيف يعمل

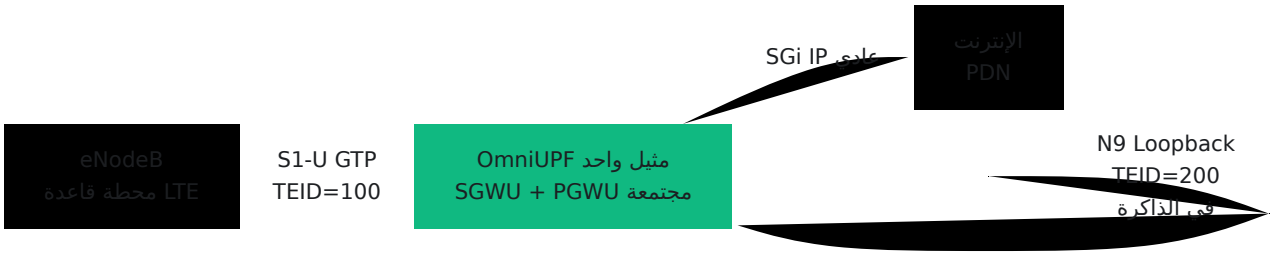
النشر التقليدي (مثيلين)

تدفق الحزم:

1. eNodeB → SGWU: تصل حزمة GTP (TEID=100) على S1-U
2. SGWU: تطابق PDR الصاعدة، وتغلف في نفق GTP جديد (TEID=200)
3. **تُرسل الحزمة عبر الشبكة الفيزيائية N9** إلى مثيل PGWU

4. PGWU: تستقبل GTP (TEID=200)، تفك التغليف، وتحيل إلى الإنترنت
5. **قفزة شبكة 1 + XDP الإجمالي: 2 تمريرات**

نشر حلقة N9 (مثيل واحد)



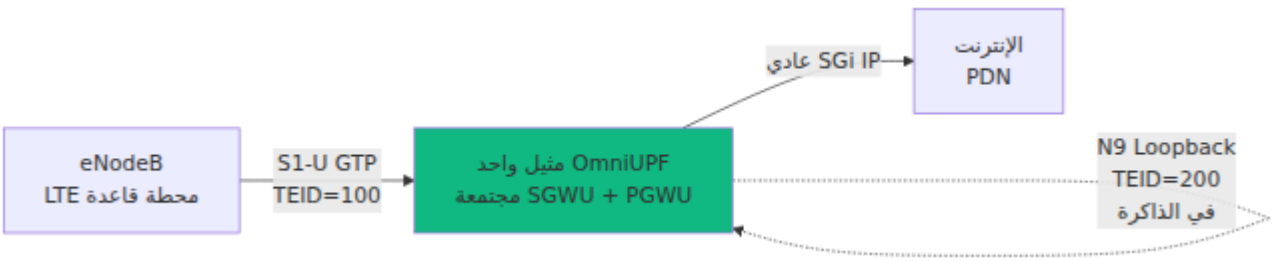
تدفق الحزم مع حلقة N9:

1. eNodeB → دور SGWU: تصل حزمة GTP (TEID=100) على S1-U
2. دور SGWU: تطابق PDR الصاعدة
3. عنوان IP الوجهة = عنوان IP (10.0.1.10) المحلي **كشف الحلقة:**
4. تحديث GTP TEID 200 إلى (جلسة PGWU) **معالجة في المكان:**
5. دور PGWU: تفك التغليف، وتحيل إلى الإنترنت
6. **صفر قفزات شبكة، XDP الإجمالي: 1 تمريرة**

فائدة الأداء: توجيه داخلي دون ميكروثانية مقابل ميلي ثانية لجولة الشبكة

تفاصيل معالجة الحزم

تدفق الصعود: eNodeB → SGWU → PGWU → الإنترنت

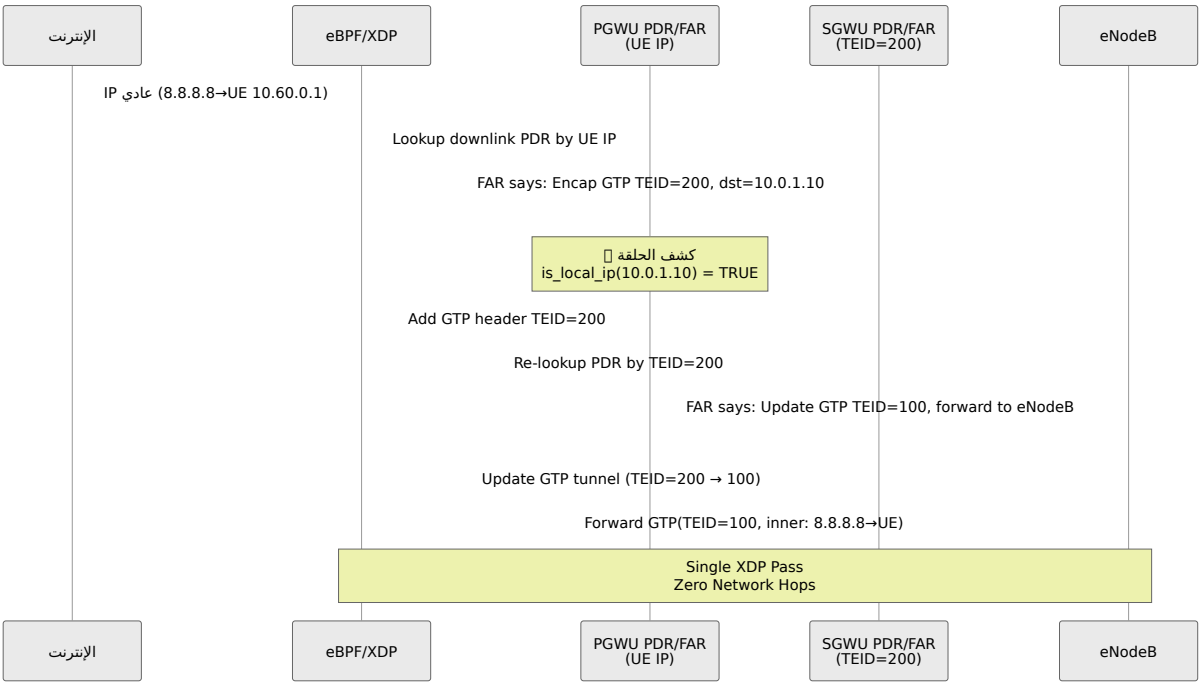


مسار كود eBPF: `ebpf/xdp/n3n6_entrypoint.c` lines 349-403

الخطوات الرئيسية:

1. حزمة GTP من eNodeB مع TEID=100 **استلام:**
2. البحث عن PDR الصاعدة لجلسة SGWU (TEID=100) **مطابقة PDR:**
3. تغليف في GTP مع TEID=200، 10.0.1.10 إحالة إلى **إجراء FAR:**
4. `is_local_ip(10.0.1.10)` تعود TRUE **تحقق الحلقة:**
5. تغيير `ctx->gtp->teid` (في ذاكرة النواة) من 100 إلى 200 **تحديث TEID:**
6. البحث عن PDR لـ TEID=200 (جلسة PGWU) **إعادة المعالجة:**
7. إزالة رأس GTP، إحالة إلى الإنترنت **إجراء FAR:**
8. إرسال حزمة IP عادية إلى واجهة N6 **توجيه:**

تدفق النزول: الإنترنت → PGWU → SGWU → eNodeB



مسار كود eBPF: `ebpf/xdp/n3n6_entrypoint.c` lines 137-194 (IPv4), 265-322 (IPv6)

الخطوات الرئيسية:

1. حزمة IP عادية من الإنترنت موجهة إلى UE (10.60.0.1) **استلام:**
2. البحث عن PDR النزولية بواسطة UE IP (جلسة PGWU) **مطابقة PDR:**
3. تغليف في GTP مع TEID=200، 10.0.1.10 إحالة إلى **إجراء FAR:**
4. `is_local_ip(10.0.1.10)` تعود TRUE **تحقق الحلقة:**

5. **إضافة GTP:** تغليف الحزمة مع TEID=200
6. البحث عن **إعادة المعالجة:** PDR لـ TEID=200 (جلسة SGWU)
7. **إجراء FAR:** تحديث نفق GTP إلى eNodeB TEID=100
8. إرسال حزمة **توجيه:** GTP إلى واجهة S1-U (eNodeB)

التكوي��

المتطلبات

طائرة التحكم:

- **SGWU-C**: يجب أن تتصل بواجهة PFCP لـ OmniUPF (على سبيل المثال، `192.168.1.10:8805`)
- **PGWU-C**: يجب أن تتصل بنفس واجهة PFCP لـ OmniUPF

الشبكة:

- **عنوان IP واحد** لكل من واجهتي N9 و N3
- **عناوين IP مختلفة** لـ PGWU-C و SGWU-C (إذا كانت تعمل على نفس المضيف، استخدم منافذ مختلفة)

تكوين OmniUPF

`/etc/omniupf/runtime.exs`:

```
# واجهات الشبكة
xdp_interfaces = "eth0"              # واجهة واحدة لـ N9 و S1-U
xdp_attach_mode = "native"           # استخدم native لأفضل أداء

# واجهة PFCP
pfcp_address = "192.168.1.10"        # عنوان PFCP لـ OmniUPF
pfcp_port = 8805                     # منفذ PFCP
node_id = "192.168.1.10"             # معرف عقدة PFCP لـ OmniUPF

# واجهات طبقة المستخدم
n3_address = "10.0.1.10"             # IP واجهة S1-U/N3
n9_address = n3_address               # IP واجهة N9 (نفسها كـ N3)

# تجمعات الموارد
feature_ueip = true
ueip_pool = "10.60.0.0/16"           # تجمع عناوين IP لـ UE
feature_ftup = true
teid_pool_start = 1
teid_pool_end = 65_535

# السعة
max_sessions = 100_000               # الحد الأقصى لجلسات UE
المتزامنة

# API
api_port = 8080
```

التكوين الرئيسي:

- **يجب أن تكون متطابقة** `n3_address` **و** `n9_address` لتمكين الحلقة
- عنوان PFCP واحد للاستماع لكل من طائرات التحكم
- عدد كافٍ من `max_sessions` لتحميل SGWU + PGWU المجمع

تكوين طائرة التحكم

تكوين SGWU-C

```
# الإشارة إلى واجهة PFCP لـ OmniUPF
upf_pfcp_address: "192.168.1.10:8805"

# واجهة S1-U (نفسها كـ n3_address لـ OmniUPF)
sgwu_s1u_address: "10.0.1.10"

# واجهة N9 للإحالة إلى PGWU (نفسها كـ OmniUPF)
sgwu_n9_address: "10.0.1.10"
```

تكوين PGWU-C

```
# الإشارة إلى نفس واجهة PFCP لـ OmniUPF
upf_pfcp_address: "192.168.1.10:8805"

# واجهة N9 (تستقبل من SGWU)
pgwu_n9_address: "10.0.1.10"

# واجهة SGi للاتصال بالإنترنت
pgwu_sgi_address: "192.168.100.1"
```

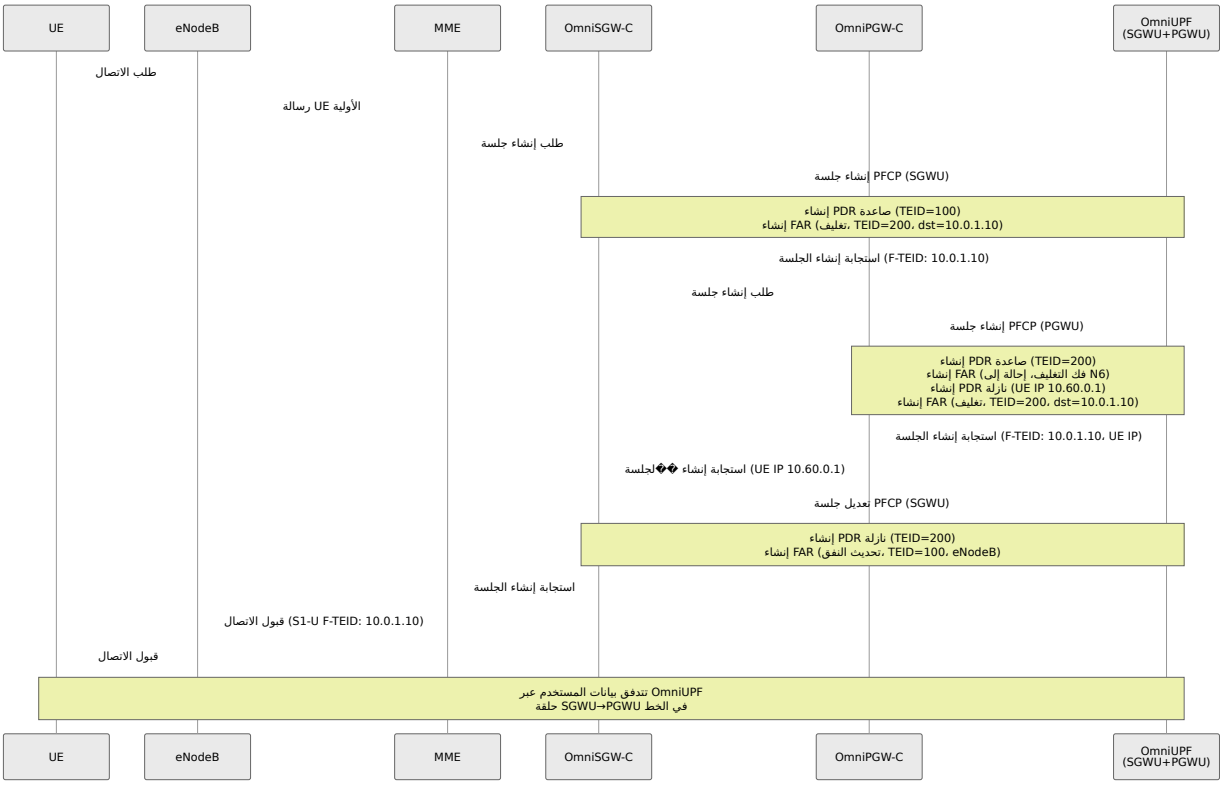
مهم:

- كلا طائرتي التحكم تتصلان بـ **نقطة نهاية** **PFCP** (`8805:`) **نفسها**
- يقوم OmniUPF بإنشاء **جمعيات** **PFCP** لـ **منفصلة** SGWU-C و PGWU-C
- الجلسات معزولة لكل طائرة تحكم (تتبع بواسطة معرف العقدة)

مثال على تدفق الجلسة

اتصال UE وتأسيس جلسة PDU

السيناريو: يتصل UE بالشبكة، ويؤسس جلسة البيانات



الجلسات PFCP التي تم إنشاؤها:

جلسة SGWU (من OmniSGW-C):

- **PDR** مطابقة **صاعدة:** TEID=100 (من eNodeB) → FAR: تغليف TEID=200، dst=10.0.1.10
- **PDR** مطابقة **نازلة:** TEID=200 (من PGWU) → FAR: تحديث النفق TEID=100، إحالة إلى eNodeB

جلسة PGWU (من OmniPGW-C):

- **PDR** مطابقة **صاعدة:** TEID=200 (من SGWU) → FAR: فك التغليف، إحالة إلى الإنترنت
- **PDR** مطابقة **نازلة:** UE IP=10.60.0.1 → FAR: تغليف TEID=200، dst=10.0.1.10

المراقبة والتحقق

تحقق من أن حلقة N9 نشطة

تحقق من سجلات XDP:

```
# عرض مخرجات تصحيح eBPF في الوقت الحقيقي
sudo cat /sys/kernel/debug/tracing/trace_pipe | grep loopback
```

المخرجات المتوقعة:

```
upf: [n3] session for teid:100 -> 200 remote:10.0.1.10
upf: [n9-loopback] self-forwarding detected, processing inline
TEID:200
upf: [n9-loopback] decapsulated, routing to N6

upf: [n6] use mapping 10.60.0.1 -> teid:200
upf: [n6-loopback] downlink self-forwarding detected, processing
inline TEID:200
upf: [n6-loopback] SGWU updating GTP tunnel to eNodeB TEID:100
upf: [n6-loopback] forwarding to eNodeB
```

مراقبة الجلسات عبر واجهة REST API

قائمة جمعيات PFCP:

```
curl http://localhost:8080/api/v1/upf_pipeline | jq
```

المخرجات المتوقعة:

```
{
  "associations": [
    {
      "node_id": "sgwc.example.com",
      "address": "192.168.1.20:8805",
      "sessions": 1000
    },
    {
      "node_id": "pgwc.example.com",
      "address": "192.168.1.21:8805",
      "sessions": 1000
    }
  ],
  "total_sessions": 2000
}
```

تحقق من وجود جمعيتين منفصلتين (واحدة لـ SGWU-C، واحدة لـ PGWU-C)

قائمة الجلسات النشطة:

```
curl http://localhost:8080/api/v1/sessions | jq '.sessions[] |
{local_seid, ue_ip, uplink_teid}'
```

المخرجات المتوقعة:

```
{
  "local_seid": 12345,
  "ue_ip": "10.60.0.1",
  "uplink_teid": 100
}
{
  "local_seid": 67890,
  "ue_ip": "10.60.0.1",
  "uplink_teid": 200
}
```

جلستان لكل UE:

- جلسة من SGWU-C (TEID=100، واجهة S1-U)
- جلسة من PGWU-C (TEID=200، واجهة N9)

مقاييس الأداء

تحقق من إحصائيات الحزم:

```
curl http://localhost:8080/api/v1/xdp_stats | jq
```

المقاييس الرئيسية:

- `xdp_processed`: إجمالي الحزم المعالجة في eBPF
- `xdp_pass`: الحزم التي تم تمريرها إلى كومة الشبكة (يجب أن تكون صفرًا لحركة المرور في الحلقة)
- `xdp_redirect`: الحزم التي تم إحالتها عبر إعادة توجيه XDP
- `xdp_tx`: الحزم المرسلة (تستخدم حركة المرور في الحلقة هذا)

لحركة مرور حلقة N9:

- **أقل ما يمكن** يجب أن تكون `xdp_pass` (فقط حركة المرور غير الحلقية)
- تعد حركة المرور في الحلقة `xdp_redirect` أو `xdp_tx`

استكشاف الأخطاء وإصلاحها

حركة مرور N9 تذهب إلى الشبكة بدلاً من الحلقة

الأعراض: الحزم المرسلة إلى واجهة الشبكة، تأخير مرتفع

السبب الجذري: `n3_address` ≠ `n9_address`

الحل (في `runtime.exs`):

```
# خاطئ:
n3_address = "10.0.1.10"
n9_address = "10.0.1.20"   # IP مختلف، لا حلقة!

# صحيح:
n3_address = "10.0.1.10"
n9_address = n3_address     # نفس IP، يمكّن الحلقة
```

التحقق:

```
curl http://localhost:8080/api/v1/dataplane_config | jq
```

يجب أن يظهر:

```
{
  "n3_ipv4_address": "10.0.1.10",
  "n9_ipv4_address": "10.0.1.10"
}
```

غير موجود بعد الحلقة PDR

الأعراض: السجلات تظهر `[n9-loopback] no PDR for destination TEID`

السبب الجذري: لم يتم إنشاء جلسة PGWU أو عدم تطابق TEID

التشخيص:

1. **تحقق من جلسات PFCP:**

   ```
   curl http://localhost:8080/api/v1/sessions | jq
   '.sessions[] | select(.uplink_teid == 200)'
   ```

2. **تحقق من تكوين FAR:**

```
curl http://localhost:8080/api/v1/far_map | jq '.[] |
select(.teid == 200)'
```

الحل: تأكد من أن PGWU-C تنشئ جلسة مع TEID متطابق يستخدمه SGWU-C للإحالة N9

استخدام CPU مرتفع

الأعراض: استخدام CPU أعلى من المتوقع

السبب الجذري: برنامج eBPF يعالج الحزم عدة مرات أو عمليات بحث خريطة مفرطة

التشخيص:

```
# تحقق من أنماط الوصول إلى خريطة eBPF
sudo bpftool map dump name pdr_map_teid_ip4 | wc -l
sudo bpftool map dump name far_map | wc -l
```

الحل:

- إذا كانت الخريطة ممتلئة (تسبب فشل البحث) `max_sessions` زيادة
- لا يسبب فقدان الحزم وإعادة الإرسال QER تحقق من أن تحديد معدل
- تحقق من وجود تخزين مفرط للحزم

فقدان الحزم أثناء الانتقال

الأعراض: فقدان الحزم أثناء انتقال eNodeB

السبب الجذري: عدم تكوين التخزين المؤقت أو حدود التخزين المؤقت غير كافية

التكوين:

```
# في runtime.exs
buffer_port = 22152
```

التحقق:

```
curl http://localhost:8080/api/v1/upf_buffer_info | jq
```

فوائد حلقة N9

الأداء

التحسين	حلقة) مثيل واحد N9)	مثيلين	المقياس
أسرع 1000 مرة	< 1 ميكروثانية	1-5 مللي ثانية	**التأخير**
أعلى 2-3 مرات	محدودة بالمعالج/ الذاكرة	محدودة بالشبكة	**الإنتاجية**
تخفيض 40- 50%	1× تمريرة XDP	2× تمريرات XDP + كومة الشبكة	**استخدام CPU**
تم القضاء عليه	صفر (في الذاكرة)	خطر أثناء ازدحام الشبكة	**فقدان الحزم**

العمليات

- **نشر مبسط:** مثيل OmniUPF واحد بدلاً من اثنين
- **تقليل البنية التحتية:** نصف الخوادم، منافذ الشبكة، عناوين IP
- **تقليل التعقيد:** تكوين واحد، نقطة مراقبة واحدة
- **توفير التكاليف:** تقليل الأجهزة، الطاقة، التبريد، والصيانة
- **تسهيل استكشاف الأخطاء:** تتبع حزمة واحدة، مخرجات تصحيح eBPF واحدة

حالات الاستخدام

مثالي لـ:

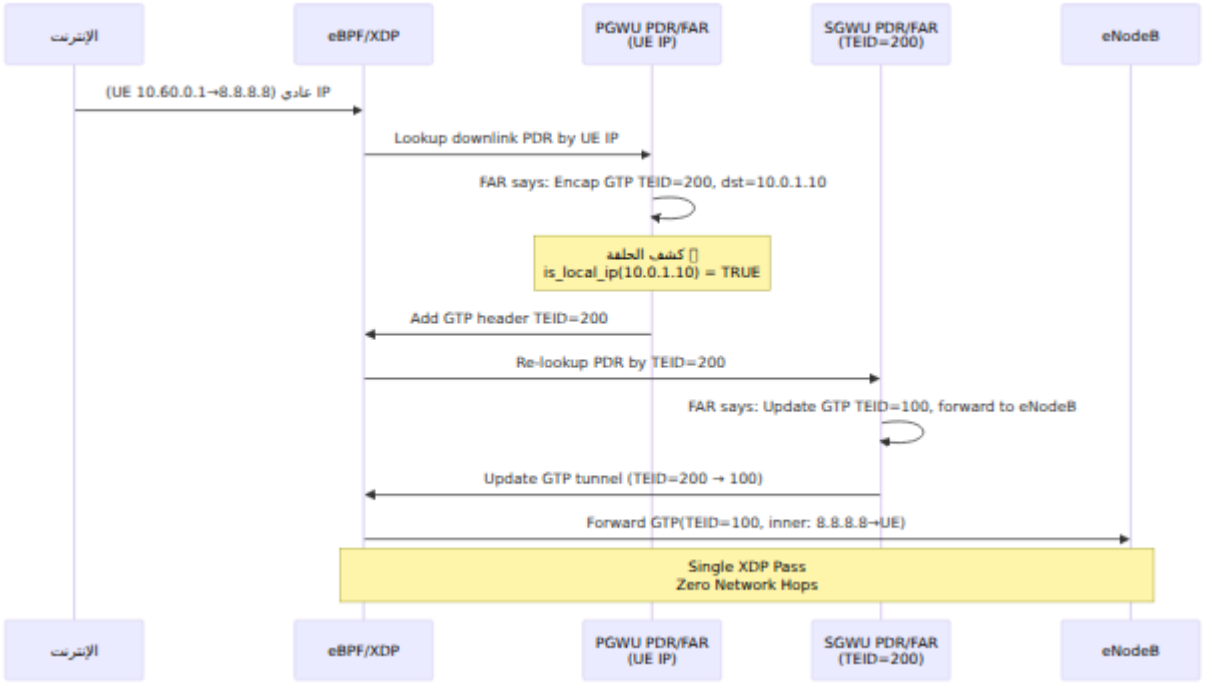
- ✅ **الحوسبة الطرفية:** تقليل التأخير للانفصال المحلي
- ✅ **نشر صغير/متوسط:** < 100K مشتركين
- ✅ **مختبر/اختبار:** طبقة مستخدم EPC كاملة على VM واحدة
- ✅ **مقيد بالتكاليف:** ميزانية أجهزة محدودة

غير موصى به لـ:

- ❌ **المرونة الجغرافية:** SGWU و PGWU في مراكز بيانات مختلفة
- ❌ **المقياس الضخم:** > 1M مشتركين (فكر في التوسع الأفقي)
- ❌ **متطلبات تنظيمية:** فصل إلزامي بين SGW و PGW

المقارنة مع أوضاع النشر الأخرى

مقابل مثيلين مفصولين (N9 حلقة) مثيل واحد



الملخص

تمكن حلقة N9 **على مستوى الناقل على مثيل واحد من EPC 4G طبقة مستخدم** دون قفزات eBPF بالكامل في SGWU→PGWU من معالج حركة المرور **OmniUPF** الشبكة. وهذا يوفر:

- ⚡ **تأخير دون ميكروثانية** للإحالة بين البوابات
- 🚀 **تخفيض CPU 40-50% في استخدام** مقارنة بالمثيلين المفصولين
- 🔧 **عمليات مبسطة** - مثيل واحد، تكوين، مراقبة
- 💰 **تكلفة أقل** - نصف البنية التحتية
- 📦 **امتثال كامل لـ 3GPP** - بروتوكولات PFCP و GTP-U القياسية

التكوين تلقائي `n3_address == n9_address` - لا حاجة إلى علامات أو إعدادات خاصة. يكتشف مسار بيانات eBPF لـ OmniUPF ظروف الحلقة ويعالج الحزم في الخط.

لمزيد من المعلومات:

- **التكوين:** CONFIGURATION.md
- **الهندسة المعمارية:** ARCHITECTURE.md
- **مرجع المقاييس:** METRICS.md
- **المراقبة:** MONITORING.md
- **العمليات:** OPERATIONS.md
- **استكشاف الأخطاء:** TROUBLESHOOTING.md

مرجع رموز أسباب PFCP

نظرة عامة

يستخدم بروتوكول التحكم في تحويل الحزم (PFCP) رموز الأسباب في رسائل الاستجابة للإشارة إلى نتيجة الطلبات. يصف هذا المستند رموز الأسباب المطبقة في OmniUPF أثناء ومتى تحدث معالجة رسائل PFCP.

ويتم إرجاعها في رسائل استجابة **GPP TS 129.244** تتوافق جميع رموز الأسباب مع مواصفات **3** للإشارة إلى النجاح أو الفشل أو ظروف الخطأ المحددة PFCP.

مراقبة رموز الأسباب

يتتبع OmniUPF نتائج رسائل PFCP باستخدام مقاييس Prometheus. تتضمن كل استجابة PFCP رمز سبب يتم تسجيله في:

```
upf_pfcp_rx_errors{message_name="...", cause_code="...",
peer_address="..."}
```

يسمح ذلك بمراقبة:

- في مستوى التحكم node لكل نوع رسالة و **معدلات النجاح**
- **أنم��ط الأخطاء** التي تشير إلى تكوينات خاطئة أو مشكلات بروتوكول
- **صحة الارتباط** بناءً على معدلات الرفض

انظر مرجع المقاييس للحصول على الوثائق الكاملة لمقاييس PFCP.

فئات رموز الأسباب

رموز النجاح

الرمز	الاسم	متى يحدث
1	**RequestAccepted**	تم معالجة الطلب بنجاح. جميع عناصر المعلومات الإلزامية موجودة وصحيحة. تم إنشاء/تعديل/حذف القواعد بنجاح.

رموز أخطاء العميل

الرمز	الاسم	متى يحدث
64	**RequestRejected**	رفض عام لأخطاء غير محددة. يستخدم عندما لا ينطبق رمز سبب محدد.
65	**SessionContextNotFound**	تم طلب تعديل أو حذف جلسة ل SEID غير معروف. الجلسة المحددة غير موجودة على هذا UPF.
66	**MandatoryIEMissing**	عنصر المعلومات المطلوب مفقود. أمثلة: NodeID مفقود في إعداد الارتباط، F-SEID مفقود في إنشاء الجلسة، RecoveryTimeStamp مفقود.
67	**ConditionalIEMissing**	عنصر المعلومات المطلوب بشكل شرطي مفقود بناءً على عناصر المعلومات الأخرى الموجودة. يستخدم عندما تعتمد عناصر المعلومات على وجود بعضها البعض.
69	**MandatoryIEIncorrect**	عنصر المعلومات المطلوب موجود ولكنه يحتوي على بيانات غير صالحة. أمثلة: تنسيق NodeID غير قابل للتحليل، قيمة RecoveryTimeStamp غير صالحة، F-SEID مشوه.
72	**NoEstablishedPFCPAssociation**	تم محاولة إجراء عملية جلسة بدون ارتباط نشط. يجب إنشاء ارتباط PFCP قبل إنشاء الجلسات.
73	**RuleCreationModificationFailure**	خطأ في تطبيق قواعد FAR أو PDR أو QER أو URR على مسار بيانات

الرمز	الاسم	متى يحدث
		الأسباب المحتملة: امتلاء eBPF. معلمات قاعدة eBPF، سعة خريطة غير صالحة، فشل تخصيص الموارد.

رموز أخطاء الخادم/المورد

الرمز	الاسم	متى يحدث
74	**PFCPEntityInCongestion**	يعاني UPF من حمل مرتفع أو استنفاد الموارد. غير قادر مؤقتًا على معالجة الطلبات.
75	**NoResourcesAvailable**	موارد غير كافية لتلبية الطلب. أمثلة: امتلاء eBPF، سعة خريطة فشل تخصيص الذاكرة، نفاد مجموعة TEID.
77	**SystemFailure**	خطأ داخلي حرج يمنع معالجة الطلب. أمثلة: eBPF، خطأ في واجهة النواة، فشل برنامج تلف قاعدة البيانات.

رموز الميزات غير المدعومة

الرمز	الاسم	متى يحدث
68	**InvalidLength**	لا يتطابق مع الطول IE حقل طول الفعلي للبيانات. غير مستخدم حاليًا في OmniUPF.
70	**InvalidForwardingPolicy**	سياسة التوجيه غير مدعومة من قبل UPF. غير مستخدم حاليًا في OmniUPF.
71	**InvalidFTEIDAllocationOption**	غير مدعوم. غير F-TEID خيار تخصيص مستخدم حاليًا في OmniUPF.
76	**ServiceNotSupported**	الخدمة أو الميزة المطلوبة غير مطبقة. غير مستخدم حاليًا في OmniUPF.
78	**RedirectionRequested**	يطلب إعادة التوجيه إلى مثيل UPF آخر. غير مستخدم حاليًا في UPF OmniUPF.

السيناريوهات والأسباب الشائعة

فشل إعداد ��لارتباط

السيناريو: NodeID مفقود

```
SMF → UPF: Association Setup Request (no NodeID)
UPF → SMF: Association Setup Response (Cause: MandatoryIEMissing)
```

الحل: تأكد من أن SMF يتضمن عنصر NodeID في جميع طلبات إعداد الارتباط.

السيناريو: تنسيق NodeID غير صالح

```
SMF → UPF: Association Setup Request (NodeID="invalid")
UPF → SMF: Association Setup Response (Cause:
MandatoryIEIncorrect)
```

الحل: يجب أن يكون NodeID عنوان FQDN صالح أو عنوان IPv4/IPv6.

السيناريو: الطابع الزمني للاسترداد مفقود

```
SMF → UPF: Association Setup Request (no RecoveryTimeStamp)
UPF → SMF: Association Setup Response (Cause: MandatoryIEMissing)
```

الحل: تضمين RecoveryTimeStamp في طلب إعداد الارتباط.

فشل إنشاء الجلسة

السيناريو: لا يوجد ارتباط تم إنشاؤه

```
SMF → UPF: Session Establishment Request
UPF → SMF: Session Establishment Response (Cause:
NoEstablishedPFCPAssociation)
```

الحل: إنشاء ارتباط PFCP قبل إنشاء الجلسات.

السيناريو: فشل إنشاء القاعدة

```
SMF → UPF: Session Establishment Request
UPF processes FARs, QERs, URRs successfully
UPF fails to create PDR (eBPF map full)
UPF → SMF: Session Establishment Response (Cause:
RuleCreationModificationFailure)
```

الحل:

- تحقق من سعة خريطة eBPF (انظر مراقبة السعة)
- زيادة أحجام الخرائط في تكوين UPF
- تقليل عدد الجلسات النشطة

السيناريو: F-SEID مفقود

```
SMF → UPF: Session Establishment Request (no CP F-SEID)
UPF → SMF: Session Establishment Response (Cause:
MandatoryIEMissing)
```

الحل: تضمين CP F-SEID في طلب إنشاء الجلسة.

فشل تعديل الجلسة

السيناريو: SEID غير معروف

```
SMF → UPF: Session Modification Request (SEID=12345)
UPF has no session with SEID 12345
UPF → SMF: Session Modification Response (Cause:
SessionContextNotFound)
```

الحل:

- تحقق من أن SEID يتطابق مع القيمة من استجابة إنشاء الجلسة
- تحقق مما إذا كانت الجلسة قد تم حذفها بالفعل
- تأكد من استخدام مثيل UPF الصحيح (سيناريوهات حلقة N9)

فشل حذف الجلسة

السيناريو: SEID غير معروف

```
SMF → UPF: Session Deletion Request (SEID=67890)
UPF has no session with SEID 67890
UPF → SMF: Session Deletion Response (Cause:
SessionContextNotFound)
```

الحل: قد يكون SEID قد تم حذفه بالفعل أو لم يكن موجودًا أبدًا.

استكشاف الأخطاء باستخدام رموز الأسباب

استخدام مقاييس Prometheus

استعلام Prometheus لتحديد أنماط الأخطاء:

```
# معدل الأخطاء حسب رمز السبب
rate(upf_pfcp_rx_errors{cause_code!="RequestAccepted"}[5m])

# الأسباب الرئيسية للرفض
topk(5, sum by (cause_code) (upf_pfcp_rx_errors))

# الأخطاء حسب نظير SMF
sum by (peer_address, cause_code)
(upf_pfcp_rx_errors{cause_code!="RequestAccepted"})

# فشل إنشاء الجلسات
upf_pfcp_rx_errors{message_name="SessionEstablishmentRequest",
cause_code!="RequestAccepted"}
```

استخدام واجهة الويب

انتقل إلى صفحة **الجلسات** لعرض:

- عدد الجلسات النشطة لكل node في مستوى التحكم
- معدلات نجاح/فشل إنشاء الجلسة
- الأخطاء الأخيرة في الجلسة

انتقل إلى صفحة **السعة** لتشخيص:

- استخدام خريطة eBPF (سبب الجذر لفشل إنشاء ال��اعدة)
- مؤشرات استنفاد الموارد

انظر دليل واجهة الويب للحصول على تعليمات المراقبة التفصيلية.

خطوات تصحيح الأخطاء الشائعة

معدل مرتفع من MandatoryIEMissing:

1. تحقق من تكوين SMF لعناصر المعلومات المطلوبة
2. تحقق من توافق إصدار مكتبة PFCP
3. راجع سجلات SMF لأخطاء بناء عناصر المعلومات

فشل متكرر في RuleCreationModificationFailure:

1. تحقق من سعة خريطة eBPF: `GET /api/v1/map_info`
2. راقب استخدام الخريطة: `upf_ebpf_map_used / upf_ebpf_map_capacity`
3. زيادة أحجام الخرائط في التكوين إذا كانت > 70% مستخدمة
4. انظر تخطيط السعة

أخطاء NoEstablishedPFCPAssociation:

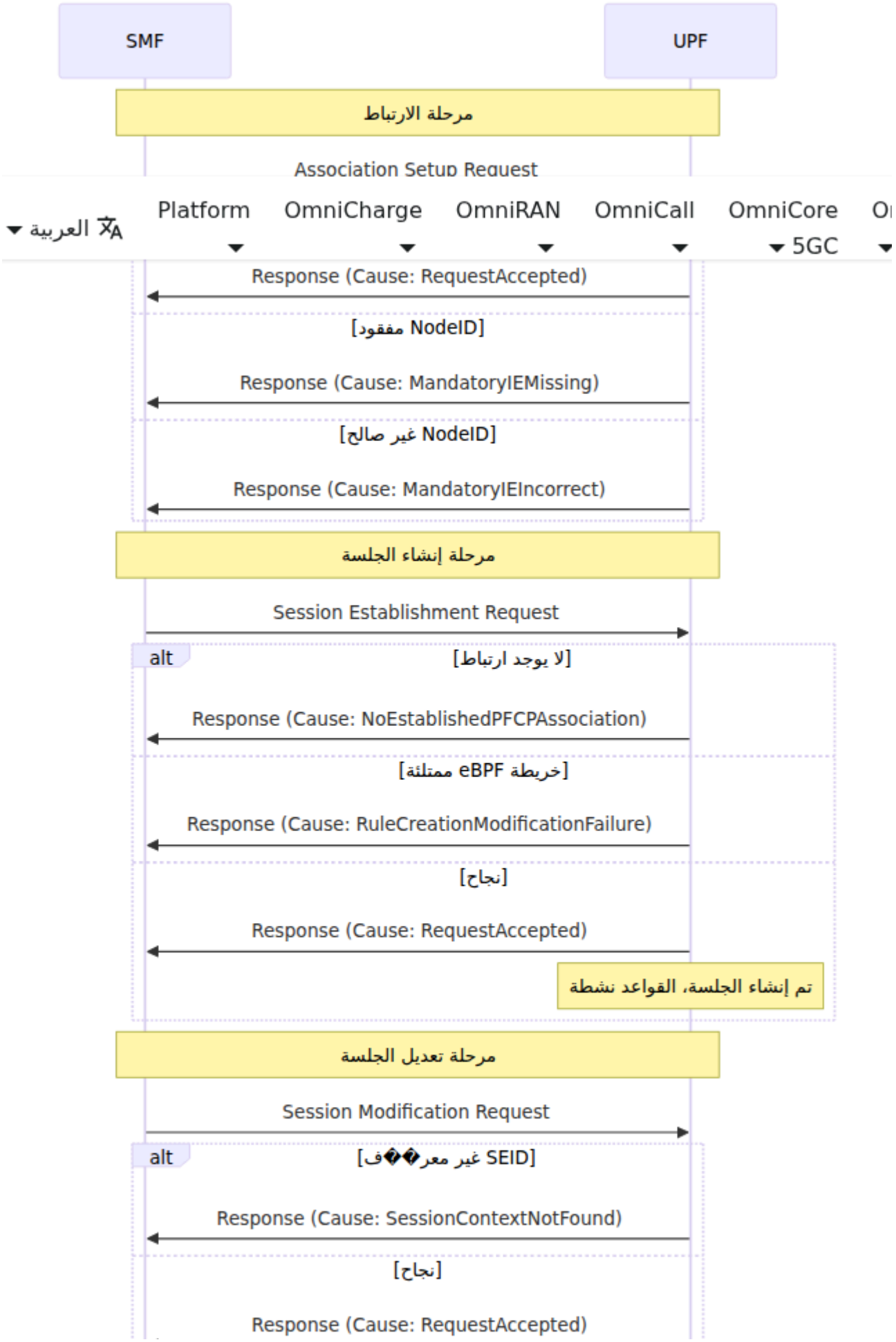
1. تحقق من وجود الارتباط: `GET /api/v1/pfcp_associations`
2. تحقق من تكوين مهلة نبضات القلب
3. راجع سجلات إعداد الارتباط
4. تأكد من أن SMF و UPF يمكنهما الوصول إلى بعضهما البعض

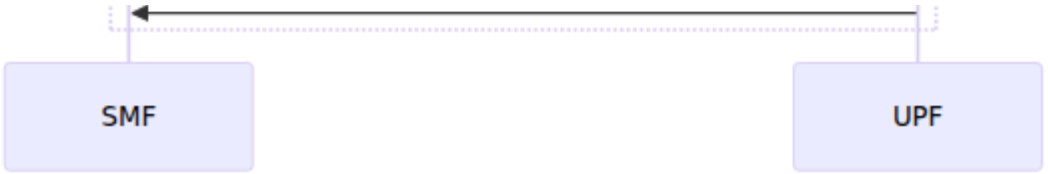
عند التعديل SessionContextNotFound:

1. تحقق من SEID من استجابة إنشاء الجلسة
2. تحقق مما إذا كانت الجلسة قد حذفت
3. تأكد من استخدام نقطة نهاية UPF الصحيحة بالنسبة لحلقة N9:
4. استعلام الجلسات النشطة: `GET /api/v1/pfcp_sessions`

تأثير رمز السبب على العمليات

دورة حياة الجلسة


SMF
UPF
مرحلة الارتباط
Association Setup Request
العربية
Platform
OmniCharge
OmniRAN
OmniCall
OmniCore
5GC
Response (Cause: RequestAccepted)
[NodeID مفقود]
Response (Cause: MandatoryIEMissing)
[NodeID غير صالح]
Response (Cause: MandatoryIEIncorrect)
مرحلة إنشاء الجلسة
Session Establishment Request
alt
[لا يوجد ارتباط]
Response (Cause: NoEstablishedPFCPAssociation)
[خريطة eBPF ممتلئة]
Response (Cause: RuleCreationModificationFailure)
[نجاح]
Response (Cause: RequestAccepted)
تم إنشاء الجلسة، القواعد نشطة
مرحلة تعديل الجلسة
Session Modification Request
alt
[SEID غير معر��ف]
Response (Cause: SessionContextNotFound)
[نجاح]
Response (Cause: RequestAccepted)



المقاييس والتنبيه

التنبيهات الموصى بها:

```
# حرج: معدل رفض مرتفع
- alert: PfcpHighRejectionRate
  expr: |
    rate(upf_pfcp_rx_errors{cause_code!="RequestAccepted"}[5m]) > 0.1
  annotations:
    summary: "معدل رفض PFCP مرتفع: {{ $value }}/s"

# تحذير: مشكلات السعة
- alert: PfcpRuleCreationFailures
  expr: |

rate(upf_pfcp_rx_errors{cause_code="RuleCreationModificationFailure"}
[5m]) > 0
  annotations:
    summary: "تم الكشف عن فشل إنشاء قواعد PFCP"

# تحذير: مشكلات الارتباط
- alert: PfcpNoAssociation
  expr: |

rate(upf_pfcp_rx_errors{cause_code="NoEstablishedPFCPAssociation"}
[5m]) > 0
  annotations:
    summary: "بدون ارتباط PFCP تمت محاولة جلسات"
```

الامتثال لمعايير 3GPP

تنفذ OmniUPF رموز الأسباب وفقًا لـ:

- **3GPP TS 129.244 v16.4.0** - مواصفة PFCP
- **القسم 8.2.1** IE السبب - تعريف

- جدول قيم الأسباب - **القسم 8.19**

الوثائق ذات الصلة

- **تكامل بروتوكول PFCP** - بنية PFCP وتعامل الرسائل
- **مرجع المقاييس** - وثائق مقياس upf_pfcp_rx_errors
- **دليل المراقبة** - مراقبة السعة والتنبيه
- **دليل استكشاف الأخطاء** - مشكلات ارتباط PFCP والجلسة
- **دليل واجهة الويب** - مراقبة الجلسات والارتباطات

دليل إدارة القواعد

جدول المحتويات

1. نظرة عامة
2. قواعد اكتشاف الحزم (PDR)
3. قواعد إجراءات التوجيه (FAR)
4. قواعد تنفيذ جودة الخدمة (QER)
5. قواعد تقارير الاستخدام (URR)
6. علاقات القواعد
7. العمليات الشائعة
8. استكشاف الأخطاء وإصلاحها

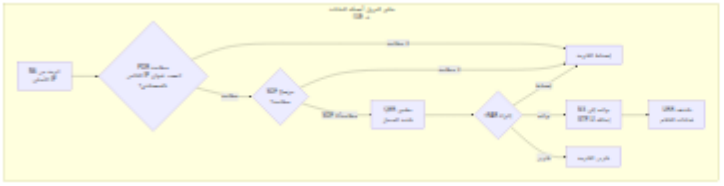
نظرة عامة

تستخدم OmniUPF مجموعة من القواعد المترابطة لتصنيف وتوجيه وتشكيل وتتبع حركة مرور مستوى المستخدم. يتم تثبيت هذه القواعد بواسطة SMF عبر PFCP وتخزينها في خرائط eBPF لمعالجة الحزم عالية الأداء. فهم هذه القواعد وعلاقاتها أمر حيوي لتشغيل واستكشاف الأخطاء في UPF.

أنواع القواعد

نوع القاعدة	الغرض	الحقل الرئيسي	المثبت بواسطة
PDR (قاعدة اكتشاف الحزم)	تصنيف الحزم إلى تدفقات	IP أو عنوان TEID الخاص بالمستخدم	SMF عبر إنشاء/ تعديل جلسة PFCP
FAR (قاعدة إجراء التوجيه)	تحديد إجراء التوجيه	FAR ID	SMF عبر إنشاء/ تعديل جلسة PFCP
QER (قاعدة تنفيذ جودة الخدمة)	تطبيق حدود النطاق الترددي والتوسيم	QER ID	SMF عبر إنشاء/ تعديل جلسة PFCP
URR (قاعدة تقارير الاستخدام)	تتبع أحجام البيانات للفوترة	URR ID	SMF عبر إنشاء/ تعديل جلسة PFCP

تدفق معالجة القواعد



قواعد اكتشاف الحزم (PDR)

الغرض

تصنف PDRs الحزم ال��واردة إلى تدفقات حركة المرور. إنها نقطة الدخول لجميع معالجة الحزم في UPF.

هيكل PDR

PDR النزول
المفتاح: عنوان IP الخاص
بالمستخدم
IPv6 أو IPv4
FAR ID
QER ID
URR IDs
وضع SDF
مرشحات SDF

PDR الرفع
المفتاح: TEID
عدد صحيح 32 بت
FAR ID
QER ID
URR IDs
إزالة الرأس الخارجي

PDRs الرفع

تطابق PDRs الرفع الحزم التي تصل إلى واجهة N3 من RAN.

الحقل الرئيسي: TEID (معرف نقطة النفق)

- عدد صحيح غير موقع 32 بت
- يتم تعيينه بواسطة SMF ويتم الإشارة إليه إلى gNB
- فريد لكل تدفق حركة مرور المستخدم

حقول القيمة:

- **FAR ID**: مرجع لقاعدة إجراء التوجيه
- **QER ID**: مرجع لقاعدة تنفيذ جودة الخدمة (اختياري)
- **URR IDs**: قائمة بقواعد تقارير الاستخدام (اختياري)
- **إزالة الرأس الخارجي** GTP-U: علامة لإزالة الت��ليف

عملية البحث:

1. استخراج TEID من رأس GTP-U
2. بحث تجزئة في خريطة `uplink_pdr_map` eBPF
3. إذا تم العثور على مطابقة، استرجاع FAR ID وQER ID وURR IDs
4. إذا لم توجد مطابقة، إسقاط الحزمة

مثال:

```
TEID: 5678
FAR ID: 2
QER ID: 1
إزالة الرأس الخارجي: خطأ
وضع SDF: لا SDF
```

PDRs النزول

تطابق PDRs النزول الحزم التي تصل إلى واجهة N6 من شبكة البيانات.

الحقل الرئيسي: عنوان IP الخاص بالمستخدم

- عنوان IPv4 (32 بت) أو عنوان IPv6 (128 بت)
- يتم تعيينه بواسطة SMF أثناء إنشاء جلسة PDU
- فريد لكل مستخدم

حقول القيمة:

- **FAR ID**: مرجع لقاعدة إجراء التوجيه
- **QER ID**: مرجع لقاعدة تنفيذ جودة الخدمة (اختياري)
- **URR IDs**: قائمة بقواعد تقارير الاستخدام (اختياري)
- **وضع SDF**: وضع مرشح تدفق البيانات الخدمة
  - `لا SDF`: لا تصفية، جميع الحركة تتطابق
  - `فقط SDF`: يتم توجيه الحركة المطابقة فقط

    - `SDF + افتراضي`: تستخدم الحركة المطابقة SDF قواعد محددة، وتستخدم FAR الافتراضي الحركة الأخرى
- **مرشحات SDF**: مرشحات محددة للتطبيق (IP المنافذ، البروتوكولات، نطاقات)

عملية البحث:

1. استخراج عنوان IP الوجهة من رأس الحزمة
2. بحث في `downlink_pdr_map` (IPv4) أو `downlink_pdr_map_ip6` (IPv6)
3. إذا تم العثور على مطابقة، تحقق من مرشحات SDF (إذا تم تكوينها)
4. استرجاع FAR ID وQER ID وURR IDs
5. إذا لم توجد مطابقة، إسقاط الحزمة

مثال:

```
عنوان IP 10.45.0.1 :الخاص بالمستخدم
FAR ID: 1
QER ID: 1
إزالة الرأس الخارجي: خطأ
وضع SDF: لا SDF
```

مرشحات SDF (تدفق البيانات الخدمة)

توفر مرشحات SDF تصنيف حركة المرور المحددة للتطبيق داخل PDR.

حالات الاستخدام:

- تمييز حركة مرور YouTube عن تصفح الويب
- تطبيق جودة خدمة مختلفة على VoIP الأفضل مقابل البيانات ذات الجهد
- توجيه تطبيقات محددة عبر مسارات شبكة مختلفة

معايير التصفية:

- **البروتوكول**: TCP، UDP، ICMP
- **نطاق المنافذ**: المنافذ الوجهة (مثل 443 لـ HTTPS، 5060 لـ SIP)
- **نطاق عنوان IP**: الشبكات الوجهة المحددة
- **وصف التدفق**: قوالب التدفق المحددة من 3GPP

مثال على تكوين SDF:

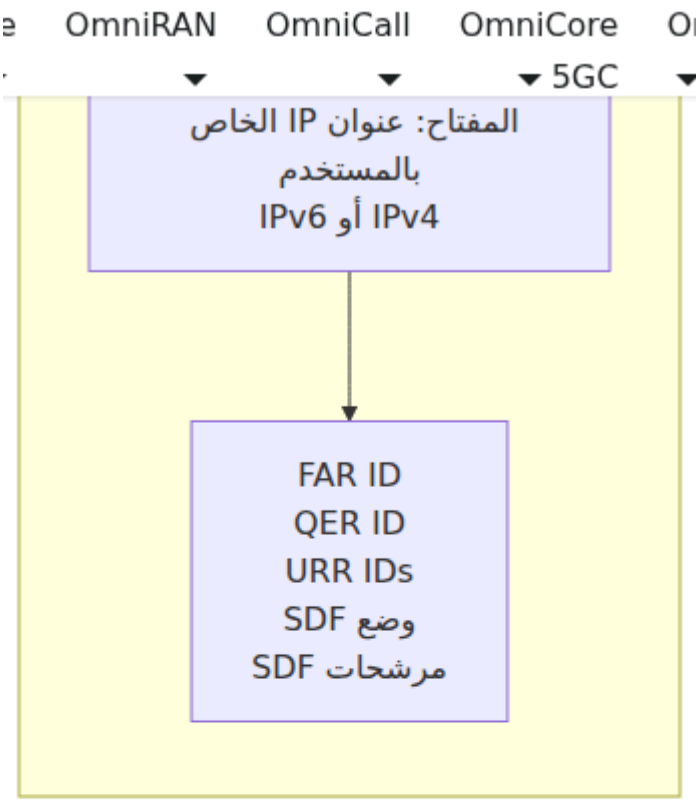
```
PDR ID: 10
عنوان IP 10.45.0.1 :الخاص بالمستخدم
وضع SDF: SDF فقط
مرشحات SDF:
  - البروتوكول: UDP، 5061-5060 :المنافذ → FAR ID 5 (FAR VoIP)
  - البروتوكول: TCP، 443 :المنفذ → FAR ID 1 (FAR الافتراضي)
```

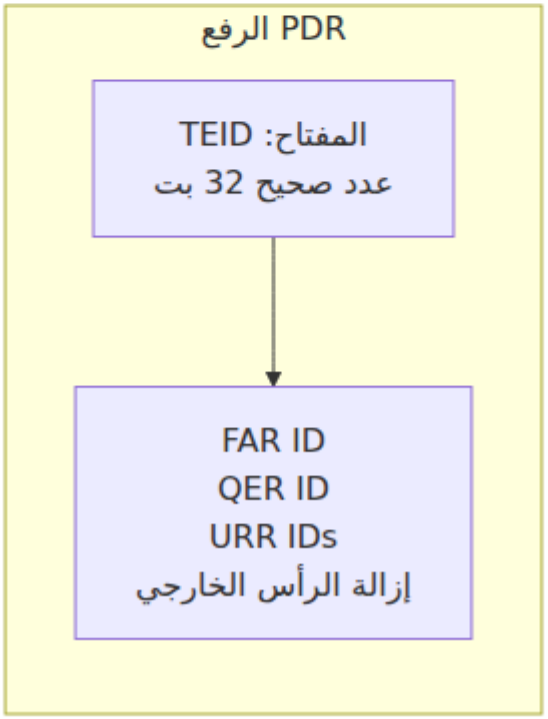
قواعد إجراءات التوجيه (FAR)

الغرض

تحدد إجراءات التوجيه، ومعلمات PDR. ما يجب القيام به مع الحزم التي تتطابق مع FARs تغليف GTP-U، ونقاط النهاية الوجهة.

هيكل FAR


e OmniRAN OmniCall OmniCore O
5GC
المفتاح: عنوان IP الخاص بالمستخدم
IPv6 أو IPv4
FAR ID
QER ID
URR IDs
وضع SDF
مرشحات SDF


PDR الرفع
المفتاح: TEID
عدد صحيح 32 بت
FAR ID
QER ID
URR IDs
إزالة الرأس الخارجي

أعلام الإجراء

تكون إجراءات FAR أعلام بتية يمكن دمجها:

العلم	البت	القيمة	الوصف
توجيه	1	2	توجيه الحزمة إلى الوجهة
تخزين	2	4	تخزين الحزمة في المخزن
إسقاط	0	1	التخلص من الحزمة
إخطار	3	8	إرسال إشعار إلى مستوى التحكم
تكرار	4	16	تكرار الحزمة إلى وجهات متعددة

تركيبات الإجراء الشائعة:

- `الإجراء: 2 (توجيه)` - توجيه عادي (الأكثر شيوعًا)
- `الإجراء: 6 (توجيه + تخزين)` - توجيه وتخزين أثناء الانتقال
- `الإجراء: 4 (تخزين)` - تخزين فقط (أثناء تبديل المسار)
- `الإجراء: 1 (إسقاط)` - إسقاط الحزمة (نادر، عادةً للتنفيذ السياسي)

التحكم في التخزين

تتحكم علامة BUFFER (بت 2) في تخزين الحزم أثناء أحداث التنق��. يعتبر التخزين ميزة حيوية تمنع فقدان الحزم أثناء انتقالات حالة UPF في UE.

متى يتم استخدام التخزين

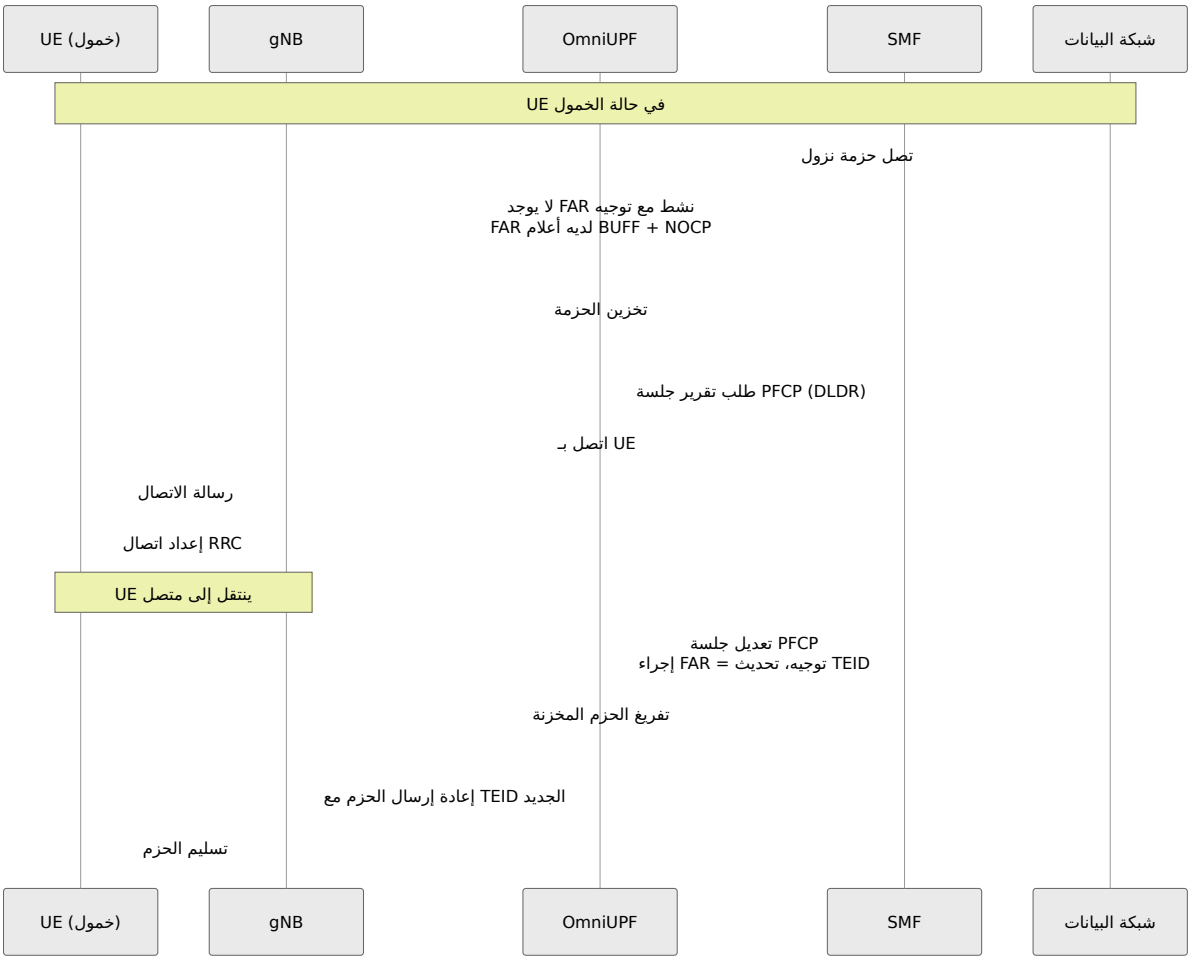
الانتقال من الخمول إلى الاتصال: عندما تصل حزم النزول إلى UE (غير في حالة الخمول متصل بـ gNB)، يقوم UPF:

1. بتخزين الحزم
2. بإرسال إشعار بيانات النزول (DLDR) إلى SMF
3. يقوم SMF بالاتصال بـ UE لإيقاظه والاتصال
4. بمجرد الاتصال، يقوم SMF بتحديث FAR بإجراء التوجيه

5. يقوم UPF بتفريغ الحزم المخزنة إلى UE

مؤقتًا بتخزين الحزم UPF يقوم ،gNB إلى gNB **الانتقال (متصل إلى متصل)**: أثناء الانتقال من لمنع الفقد:

1. يتم إسقاط الاتصال القديم بـ gNB
2. يقوم SMF بتعيين إجراء FAR إلى التخزين
3. تتراكم الحزم أثناء تبديل المسار
4. يتصل UE بـ gNB الجديد
5. يقوم SMF بتحديث FAR مع TEID الجديد وإجراء التوجيه
6. يقوم UPF بتفريغ الحزم إلى gNB الجديد



سعة التخزين والحدود

الحدود العالمية للتخزين:

- **أقصى عدد ��لحزم**: 100,000 (قابل للتكوين)
- **أقصى عدد للبكسل**: بناءً على الذاكرة المتاحة

- **TTL (مدة الحياة)**: 60 ثانية (قابل للتكوين)
- **الحزم التي تتجاوز TTL**: يتم إسقاطها تلقائيًا

الحدود لكل FAR:

- **أقصى عدد للحزم لكل FAR**: 10,000 (قابل للتكوين)
- **الغرض**: منع واحد من استنفاد سعة التخزين FAR

سلوك تجاوز التخزين:

- يتم إسقاط الحزم الجديدة، FAR عند الوصول إلى الحد العالمي أو الحد لكل
- تتعقب المقاييس الإسقاطات مع `reason="global_limit"` أو `reason="far_limit"`
- لا يتم طرد الحزم الأقدم تلقائيًا (إسقاط صريح فقط عند انتهاء TTL)

إشعار بيانات النزول (DLDR)

عندما يقوم UPF بتخزين حزمة لـ UE في حالة الخمول، يرسل طلب تقرير جلسة PFCP إلى SMF:

محتويات DLDR:

- **نوع التقرير**: تقرير بيانات النزول (DLDR)
- **FAR ID**: FAR الذي تسبب في التخزين
- **معلومات خدمة بيانات النزول**: QFI الاختياري، مؤشر سياسة الاتصال

إجراءات SMF على DLDR:

1. اتصل بـ UE عبر AMF → gNB
2. انتظر حتى يقوم UE بإنشاء اتصال RRC
3. إرسال طلب تعديل جلسة PFCP لتحديث FAR
4. يتغير إجراء FAR من `BUFF+NOCP` إلى `FORW`
5. يقوم UPF بتفريغ الحزم المخزنة

المقاييس لـ DLDR:

- `upf_dldr_sent_total`: إجمالي DLDRs المرسلة
- `upf_dldr_send_errors`: DLDRs الفاشلة

- `upf_buffer_notify_to_flush_duration_seconds`: زمن الانتظار من DLDR إلى التفريغ

انظر مرجع المقاييس للحصول على القائمة الكاملة.

عمليات التخزين

تمكين التخزين (تعيين علامة BUFF):

- إجراء FAR `|= 0x04` (تعيين بت 2)
- مثال: `الإجراء: 2 (FORW)` → `الإجراء: 6 (FORW+BUFF)`
- يستخدم أثناء التحضير للانتقال

وضع التخزين فقط (BUFF بدون FORW):

- إجراء FAR `= 0x04` (تخزين فقط)
- يتم تخزين الحزم ولكن لا يتم توجيهها
- يستخدم لحالة UE (في انتظار الاتصال) الخمول

تعطيل التخزين (مسح علامة BUFF):

- إجراء FAR `&= ~0x04` (مسح بت 2)
- مثال: `الإجراء: 6 (FORW+BUFF)` → `الإجراء: 2 (FORW)`
- تظل الحزم المخزنة حتى يتم تفريغها أو مسحها

تفريغ التخزين:

- إعادة إرسال جميع الحزم المخزنة باستخدام قواعد FAR **الحالية**
- يتم توجيه الحزم مع TEID/وجهة محدثة
- يتم إفراغ التخزين بعد التفريغ الناجح
- يجب أن يكون لدى FAR إجراء توجيه مضبوط

مسح التخزين:

- التخلص من جميع الحزم المخزنة دون توجيه
- استخدم عندما يفشل الانتقال أو يتم حذف الجلسة
- تتعقب المقاييس مع `reason="cleared"`

مراقبة الحزم المخزنة

صفحة التخزين (واجهة الويب): انتقل إلى **التخزين** لعرض:

- إجمالي الحزم المخزنة
- إجمالي البايتات المخزنة
- عدد FARs مع حزم مخزنة
- عدد الحزم لكل FAR
- الطابع الزمني لأقدم حزمة
- تمكين/تعطيل التخزين لكل FAR
- عمليات التفريغ أو المسح

المؤشرات الرئيسية:

- **الحزم > 10 ثوانٍ قديمة**: تأخير محتمل في الاتصال
- **الحزم > 30 ثانية قديمة**: فشل محتمل في الاتصال، امسح التخزي
- **عدد الحزم العالي**: تحقق من الجلسات العالقة أو فشل الاتصال

مقاييس Prometheus:

- `upf_buffer_packets_current`: الحزم المخزنة الحالية
- `upf_buffer_bytes_current`: البايتات المخزنة الحالية
- `upf_buffer_fars_active`: FARs مع حزم مخزنة
- `upf_buffer_packets_dropped{reason}`: عدد الحزم المفقودة

انظر مرجع المقاييس للحصول على المقاييس الكاملة للتخزين.

سيناريوهات التخزين الشائعة

السيناريو 1: بيانات النزول لـ UE في حالة الخمول

```
الحالة الأولية:
- UE في وضع الخمول (لا اتصال gNB)
- إجراء FAR: 0x04 (تخزين فقط)

وصول البيانات:
1. ترسل DN حزمة نزول
2. تتطابق UPF مع PDR، وتطبق FAR
3. FAR لديه علم BUFF → تم تخزين الحزمة
4. ترسل UPF DLDR إلى SMF
5. يقوم SMF بالاتصال بـ UE
6. يتصل UE بـ gNB
7. يقوم SMF بتعديل FAR: 0 = الإجراءx02 (توجيه)
8. تقوم UPF بتفريغ الحزم المخزنة مع TEID الجديد
```

السيناريو 2: التحضير للانتقال

```
الحالة الأولية:
- UE متصل بـ gNB-1 (TEID 1234)
- إجراء FAR: 0x02 (توجيه)

عملية الانتقال:
1. يقوم SMF بتعديل FAR: 0 = الإجراءx06 (توجيه + تخزين)
2. يتم توجيه الحزم إلى gNB-1 وتخزينها
3. يتبدل UE إلى gNB-2
4. يقوم SMF بتعديل FAR: TEID = 5678، 0 = الإجراءx02 (توجيه)
5. تقوم UPF بتفريغ الحزم المخزنة إلى gNB-2 مع TEID الجديد
6. لا يوجد فقدان للحزم أثناء الانتقال
```

السيناريو 3: تبديل المسار

```
الحالة الأولية:
- UE متصل، تدفق بيانات نشط

تبديل المسار:
1. يقوم SMF بتعديل FAR: 0 = الإجراءx04 (تخزين فقط)
2. يتم تخزين جميع الحزم الواردة (لا يتم توجيهها)
3. يعيد الشبكة تكوين المسار
4. يقوم SMF بتعديل FAR: 0 = الإجراءx02 (توجيه)، وجهة جديدة
5. تقوم UPF بتفريغ جميع الحزم المخزنة إلى المسار الجديد
```

إنشاء الرأس الخارجي

يحدد ما إذا كان يجب إضافة تغليف GTP-U.

FAR الرفع (N3 → N6):

- إنشاء الرأس الخارجي: خطأ
- الإجراء: إزالة GTP-U، توجيه حزمة IP الأصلية

FAR النزول (N6 → N3):

- إنشاء الرأس الخارجي: صحيح
- عنوان IP البعيد: عنوان IP لـ gNB (مثل 200.198.5.10)
- TEID: معرف النفق لحركة مرور UE
- الإجراء: إضافة رأس GTP-U، توجيه إلى gNB

بحث FAR في واجهة الويب

توفر صفحة إدارة القواعد بحث FAR حسب ID:

الخطوات:

1. انتقل إلى القواعد → علامة FARs
2. أدخل FAR ID في حقل البحث
3. انقر على "بحث" لعرض تفاصيل FAR

المعلومات المعروضة:

- FAR ID
- الإجراء (رقمي + أعلام مفككة)
- حالة التخزين (تشغيل/إيقاف)
- إنشاء الرأس الخارجي
- عنوان IP البعيد (مع التمثيل العددي)
- TEID
- توسيع مستوى النقل

قواعد تنفيذ جودة الخدمة (QER)

الغرض

تطبق QERs معايير جودة الخدمة على تدفقات حركة المرور، بما في ذلك حدود النطاق الترددي وتوسيم الحزم.

هیکل QER

معلمات QER

QFI
معرف تدفق جودة الخدمة

UL حالة البوابة
مفتوح/مغلق

DL حالة البوابة
مفتوح/مغلق

MBR الرفع
أقصى معدل بت

QER ID
معرف فريد

MBR النزول
أقصى معدل بت

GBR الرفع
معدل بت مضمون

GBR النزول
معدل بت مضمون

معايير جودة الخدمة

QFI (معرف تدفق جودة الخدمة):

- 5 بت لتدفقات جودة الخدمة في 6 معرفG
- القيم من 1-9 موحدة (مثل QFI 9 = حامل افتراضي)
- 5 تستخدم لتوسيم الحزم في GC

حالة البوابة:

- **مفتوح (0)**: يُسمح بحركة المرور
- **مغلق (غير صفري)**: يتم حظر حركة المرور

أقصى معدل بت (MBR):

- الحد الأقصى للنطاق الترددي المسموح به لتدفق الحركة
- محدد بالـ kbps
- **MBR = 0**: لا يوجد حد للسرعة (غير محدود)
- يتم إسقاط حركة المرور التي تتجاوز MBR

معدل البت المض��ون (GBR):

- الحد الأدنى من النطاق الترددي المضمون لتدفق الحركة
- محدد بالـ kbps
- **GBR = 0**: جهد أفضل (لا ضمان)
- **GBR > 0**: تدفق ذو أولوية مع نطاق ترددي مضمون

أنواع تدفقات جودة الخدمة

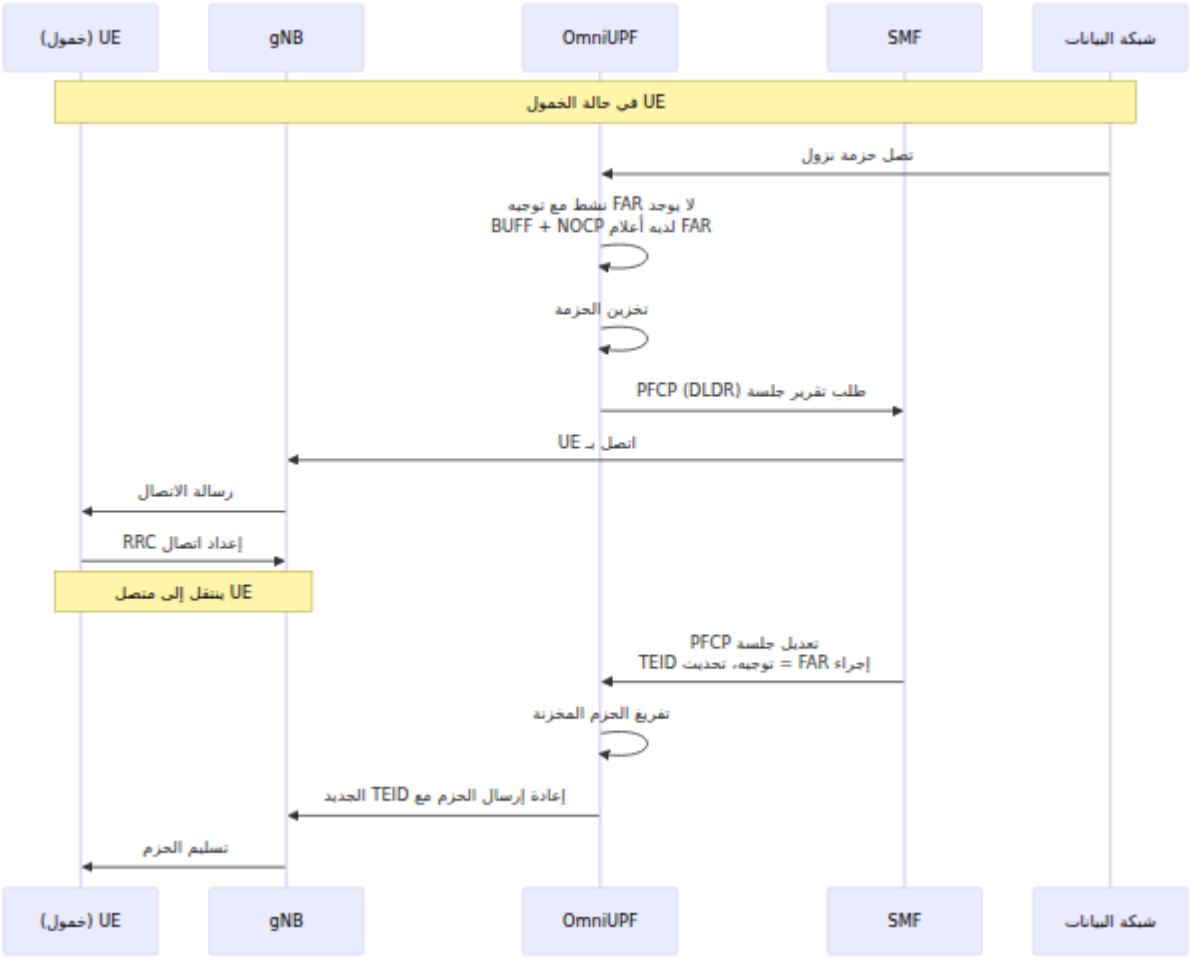
تدفقات الجهد الأفضل (GBR = 0):

```
QER ID: 1
QFI: 9
MBR 100000 :الرفع kbps (100 Mbps)
MBR 100000 :النزول kbps (100 Mbps)
GBR 0 :الرفع kbps
GBR 0 :النزول kbps
```

تدفقات مضمونة (GBR > 0):

```
QER ID: 2
QFI: 1
MBR 10000 :الرفع kbps (10 Mbps)
MBR 10000 :النزول kbps (10 Mbps)
GBR 5000 :الرفع kbps (5 Mbps)
GBR 5000 :النزول kbps (5 Mbps)
```

خوارزمية تنفيذ جودة الخدمة



آلية تنفيذ MBR

تفرض OmniUPF حدود MBR (أقصى معدل بت) باستخدام **محدد معدل نافذة منزلقة**، مما يضمن تنفيذه في مسار eBPF. تعمل هذه الخوارزمية بدقة نانو ثانية مباشرة في طبقة XDP، أداء بمعدل خطي دون تبديلات سياق النواة.

كيف تعمل

الخوارزمية: تحديد معدل النافذة المنزلقة

لكل حزمة، تقوم UPF بإجراء الفحوصات التالية:

1. **تحقق من حالة البوابة**: إذا كانت حالة البوابة `مغلقة` (غير صفري)، يتم إسقاط الحزمة على الفور
2. **تحقق من MBR**: إذا كان MBR = 0، يتم تجاوز تحديد المعدل (نطاق ترددي غير محدود)

3. **حساب زمن الإرسال**:

```
tx_time = (packet_size_bytes × 8) × (1,000,000,000 ns/sec)
/ MBR_kbps
```

4. **تحقق من النافذة**: إذا كان الوقت الحالي ضمن نافذة 5 مللي ثانية، يتم إسقاط الحزمة
5. **تقدم النافذة**: إذا تم السماح بالحزمة، يتم تقدم النافذة بواسطة `tx_time`

مثال على الحساب:

افترض:

- MBR = 100,000 kbps (100 Mbps)
- حجم الحزمة = 1500 بايت
- حجم النافذة = 5,000,000 ns (5 مللي ثانية)

```
الخطوة 1: حساب زمن الإرسال عند 100 Mbps
tx_time = (1500 بايت × 8 بت/بايت) × (1,000,000,000 ns/sec) /
100,000,000 bps
        = 12,000,000,000 / 100,000,000
        = 120 ns

الخطوة 2: تحقق مما إذا كانت الحزمة تناسب النافذة
current_time = 1000000000 ns
window_start = 999990000 ns
if (window_start + tx_time > current_time):
    إسقاط الحزمة (ستتجاوز حد المعدل)

الخطوة 3: إذا تم السماح، تقدم النافذة
window_start = window_start + 120 ns
تمرير الحزمة
```

سلوك النافذة المنزلقة

حجم النافذة 5 مللي ثانية:

- تستخدم الخوارزمية نافذة منزلقة بحجم 5 مللي ثانية
- يتم إعادة تعيين النافذة تلقائيًا إذا كانت غير نشطة لأكثر من 5 مللي ثانية
- تمنع تجويع الانفجارات أثناء فرض المعدل المتوسط

التعامل مع الانفجارات:

- يتم السماح بانفجارات صغيرة ضمن نافذة 5 مللي ثانية
- يتم تحديد معدل حركة المرور المستدامة فوق MBR
- أكثر دقة من خوارزميات دلو الرموز البسيطة

تحديد المعدل لكل اتجاه:

- يستخدم MBR الرفع الطابع الزمني `qer->ul_start`
- يستخدم MBR النزول الطابع الزمني `qer->dl_start`
- يتم تحديد كل اتجاه بشكل مستقل

نقاط تنفيذ تحديد المعدل

الرفع (N3 → N6):

1. تصل الحزمة على واجهة N3 (من gNB)
2. بحث PDR حسب TEID
3. بحث QER حسب QER ID
4. تحقق من `ul_gate_status` → إسقاط إذا كانت مغلقة
5. تطبيق `limit_rate_sliding_window()` مع `ul_maximum_bitrate`
6. وتحديث عدادات URR إذا تم السماح، يتم توجيهها إلى N6

النزول (N6 → N3):

1. تصل الحزمة على واجهة N6 (من شبكة البيانات)
2. بحث PDR حسب عنوان IP الخاص بالمستخدم
3. بحث QER حسب QER ID
4. تحقق من `dl_gate_status` → إسقاط إذا كانت مغلقة
5. تطبيق `limit_rate_sliding_window()` مع `dl_maximum_bitrate`
6. وتوجيهها إلى N3 إذا تم السماح، إضافة رأس GTP-U

حلقة N9 (SGWU ↔ PGWU):

- الرفع والنزول في سيناريوهات حلقة N9 قد تنطبق كل من QERs
- يتم التحقق من كل SGWU وPGWU بشكل مستقل عند حدود QER

مقابل الإنتاجية الملحوظة MBR

لماذا قد تختلف الإنتاجية الملحوظة عن MBR:

- **العبء البروتوكولي**: تضيف رؤوس GTP-U وUDP وIP حوالي 50-60 بايت لكل حزمة
- **تباين حجم الحزمة**: الحزم الأصغر = مزيد من العبء، كفاءة أقل
- **دقة تحديد المعدل**: يتم التنفيذ لكل حزمة، وليس لكل بايت
- **سلوك إعادة تعيين النافذة**: تسمح الفترات الخاملة التي تبلغ 5 مللي ثانية بانفجارات قصيرة فوق MBR

مثال:

```
MBR 100 :المكون Mbps
95-98~ :الإنتاجية الملحوظة Mbps (بسبب العبء البروتوكولي GTP-U/UDP/IP)
```

كيفية التحقق من تحديد المعدل:

1. تحقق من عدادات حجم URR بمرور الوقت: `upf_urr_*_volume_bytes`
2. احسب الإنتاجية: `(volume_delta_bytes × 8) / time_delta_seconds / 1000` = kbps
3. قارن مع MBR المكون في QER

GBR (معدل البت المضمون)

مهم: لا تفرض OmniUPF حاليًا الحد الأدنى من GBR. يتم تخزين GBR في QER ولكن لا يتم استخدامه لتحديد أولويات حركة المرور أو التحكم في القبول.

سلوك GBR:

- يتم قبول قيم GBR من SMF عبر PFCP
- يتم تخزين GBR في QER ويمكن رؤيته عبر API
- **لا يوجد حجز للنطاق الترددي** GBR أو تحديد أولويات حركة المرور بناءً على
- يعد GBR (جهد أفضل مقابل مضمون) بيانات وصفية لتتبع نوع التدفق

تحسين مستقبلي:

- يتطلب تنفيذ GBR جدولة حركة المرور أو صفوف موزونة
- في الإصدارات المستقبلية eBPF في QoS قد يتم تنفيذ ذلك باستخدام قدرات

قواعد تقارير الاستخدام (URR)

الغرض

تتبع URRs أحجام البيانات للفوترة والتحليلات وتنفيذ السياسات. تحتفظ بعدادات الحزم والبايتات لسجلات الفوترة SMF التي يتم الإبلاغ عنها إلى.

هيكل URR

تتبع الحجم

حجم الرفع:

- بايتات تم نقلها من UE إلى شبكة البيانات
- تقاس بعد إزالة تغليف GTP-U
- تشمل رأس IP والحمولة

حجم النزول:

- بايتات تم نقلها من شبكة البيانات إلى UE
- تقاس قبل تغليف GTP-U
- تشمل رأس IP والحمولة

إجمالي الحجم:

- مجموع أحجام الرفع والنزول
- يستخدم للإبلاغ عن الاستخدام الكلي

مشغلات تقارير الاستخدام

يمكن أن تؤدي URRs على تقارير بناءً إلى:

عتبة الحجم:

- الإبلاغ عند تجاوز الحجم الحد المكون
- مثال: الإبلاغ عن كل 1 جيجابايت من الاستخدام

عتبة الوقت:

- الإبلاغ على فترات دورية
- مثال: الإبلاغ كل 5 دقائق

استنادًا إلى الحدث:

- الإبلاغ عند إنهاء الجلسة
- الإبلاغ عند تغيير جودة الخدمة
- الإبلاغ عند الانتقال

تنسيق عرض الحجم

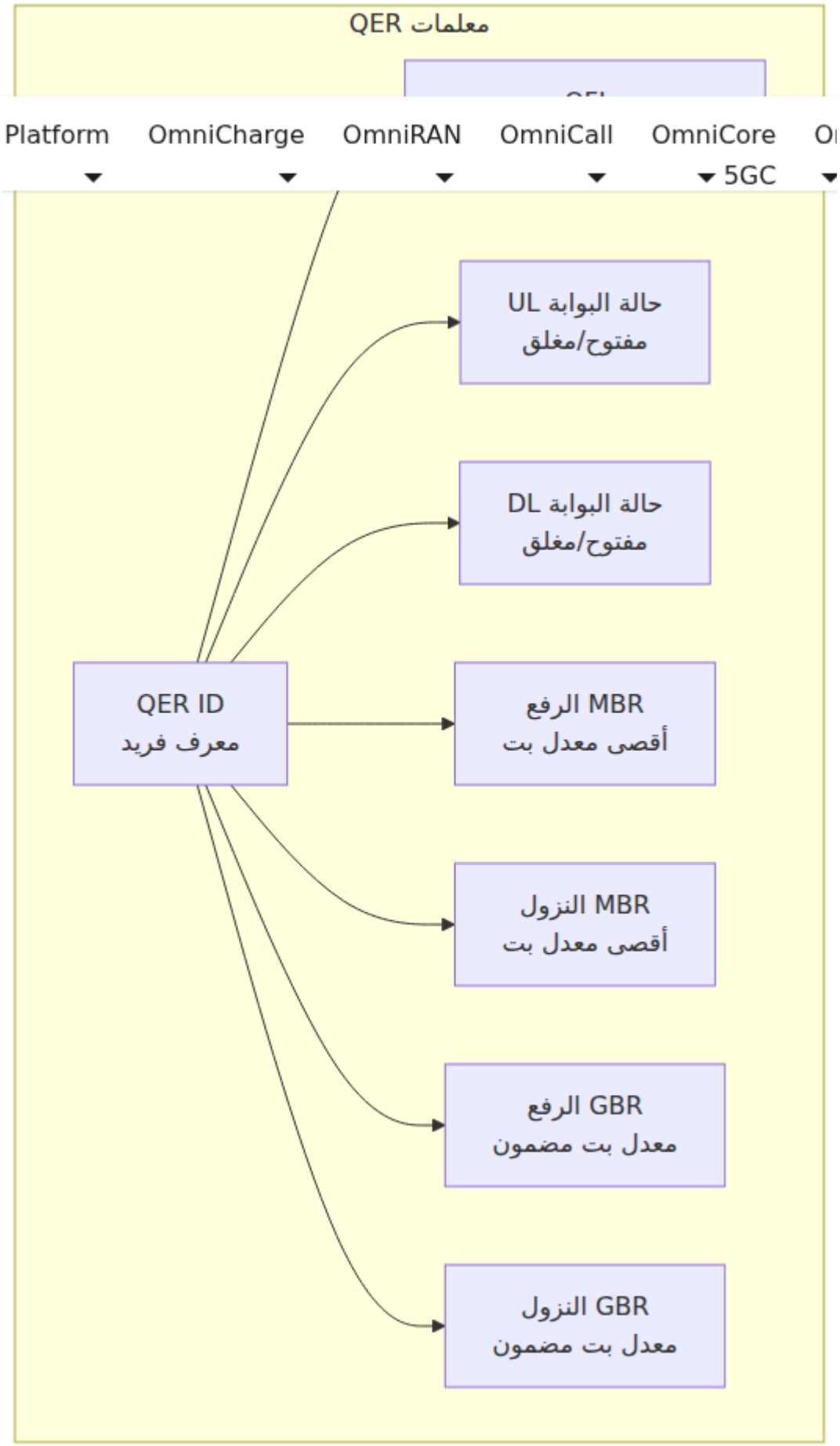
تقوم واجهة الويب تلقائيًا بتنسيق الحجم في وحدات قا��لة للقراءة البشرية:

العرض	بايتات
ب (بايت)	0 - 1023
ك ب (كيلوبايت)	1024 - 1048575
م ب (ميغابايت)	1048576 - 1073741823
ج ب (جيجابايت)	1073741824 - 1099511627775
ت ب (تيرابايت)	1099511627776+

مثال:

```
URR ID: 0
حجم الرفع: 12.3 ك ب
حجم النزول: 9.0 ك ب
إجمالي الحجم: 21.3 ك ب
```

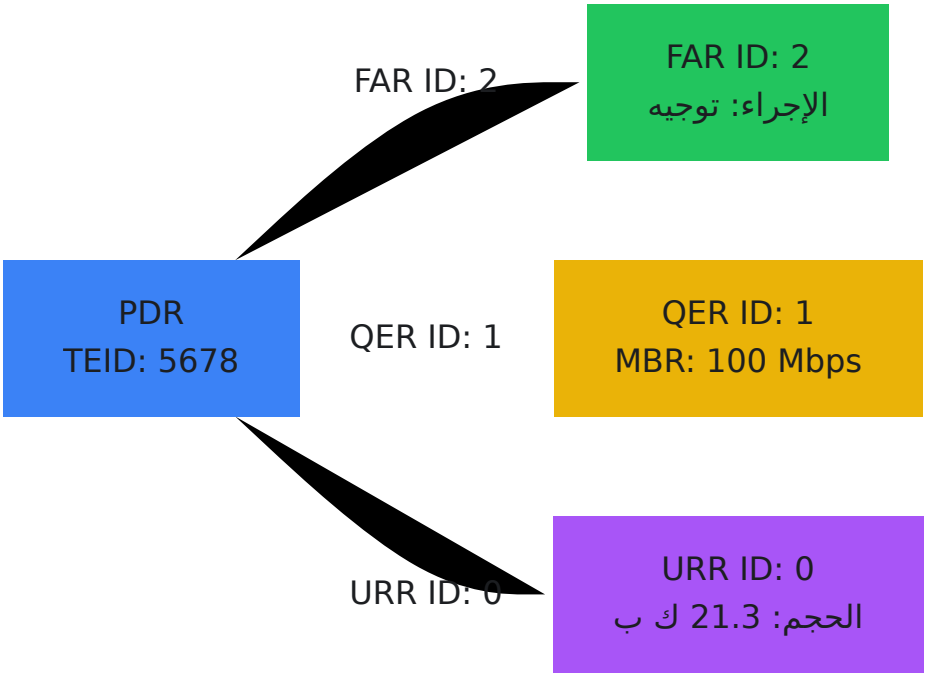
URR تدفق تقارير


معلمات QER
Platform
OmniCharge
OmniRAN
OmniCall
OmniCore 5GC
حالة البوابة UL
مفتوح/مغلق
حالة البوابة DL
مفتوح/مغلق
QER ID
معرف فريد
MBR الرفع
أقصى معدل بت
MBR النزول
أقصى معدل بت
GBR الرفع
معدل بت مضمون
GBR النزول
معدل بت مضمون

علاقات القواعد

سلسلة PDR → FAR → QER → URR

تشير كل PDR إلى FAR، والتي قد تشير إلى QER واحد أو أكثر من URRs.



مثال على تكوين الجلسة

PDR الرفع:

```
TEID: 5678
FAR ID: 2
QER ID: 1
URR IDs: [0]
إزالة الرأس الخارجي: خطأ
```

PDR النزول:

```
عنوان IP 10.45.0.1 :الخاص بالمستخدم
FAR ID: 1
QER ID: 1
URR IDs: [0]
وضع SDF: لا SDF
```

FAR ID 1 (النزول):

```
(توجيه) 2 :الإجراء
إنشاء الرأس الخارجي: صحيح
عنوان IP 200.198.5.10 :البعيد
TEID: 5678
```

FAR ID 2 (الرفع):

```
(توجيه) 2 :الإجراء
إنشاء الرأس الخارجي: خطأ
```

QER ID 1:

```
QFI: 9
MBR 100000 :الرفع kbps
MBR 100000 :النزول kbps
GBR 0 :الرفع kbps
GBR 0 :النزول kbps
```

URR ID 0:

```
حجم الرفع: 12.3 ك ب
حجم النزول: 9.0 ك ب
إجمالي الحجم: 21.3 ك ب
```

العمليات الشائعة

عرض القواعد لجلسة

عبر صفحة الجلسات:

1. انتقل إلى الجلسات
2. ابحث عن UE بواسطة IP أو TEID
3. انقر على "توسيع" لعرض جميع القواعد (PDR، FAR، QER، URR)

عبر صفحة القواعد:

1. انتقل إلى القواعد
2. استخدم البحث حسب TEID (الرفع) أو عنوان IP في علامة (النزول) الخاص بالمستخدم PDR
3. لاحظ FAR ID وQER ID وURR IDs
4. انتقل إلى علامات FAR/QER/URR لعرض القواعد المرجعية

تمكين/تعطيل التخزين

السيناريو: أثناء الانتقال، قم بتخزين الحزم لمنع الفقد

الخطوات:

1. انتقل إلى القواعد → FARs
2. أدخل FAR ID في حقل البحث
3. انقر على "بحث"
4. إذا كان التخزين مغلقًا، انقر على "تمكين التخزين"
5. تحقق من تعيين بت 2 لإجراء FAR (4 بمقدار الإجراء قيمة تزداد)

بديل عبر صفحة التخزين:

1. انتقل إلى التخزين
2. عرض FARs الممكّن مع التخزين
3. انقر على "تعطيل التخزين" عند الانتهاء من الانتقال

مراقبة الامتثال لجودة الخدمة

تحقق مما إذا كانت حركة المرور تخضع لتحديد المعدل:

1. انتقل إلى القواعد → QERs
2. ابحث عن QER ID المرتبطة بجلسة UE
3. لاحظ قيم MBR الرفع و MBR النزول
4. قارن مع معدل نمو حجم URR

احسب متوسط الإنتاجية:

```
الإنتاجية (kbps) = (فرق الحجم بالبايت × 8) / (فرق الوقت بالثواني × 1000)
```

إذا اقتربت الإنتاجية من MBR، فإن حركة المرور تخضع لتحديد المعدل.

تتبع استخدام البيانات

راقب أحجام URR:

1. انتقل إلى القواعد → URRs
2. عرض أحجام الرفع والنزول والإجمالي
3. فرز حسب الحجم الإجمالي للعثور على أعلى المستخدمين
4. تحديث بشكل دوري لمراقبة نمو الحجم

حالات الاستخدام:

- تحقق من تكامل الفوترة
- اكتشاف استخدام البيانات غير الطبيعي
- تخطيط السعة بناءً على أنماط الحركة

استكشاف الأخطاء وإصلاحها

عدم تدفق الحركة

تحقق من PDR:

1. تحقق من وجود PDR لـ TEID (الرفع) أو عنوان IP (النزول) الخاص بالمستخ��م
2. تأكد من أن FAR ID صالح
3. تحقق من أن مرشحات SDF لا تحظر الحركة

تحقق من FAR:

1. تحقق من أن الإجراء FAR (ليس إسقاط أو تخزين فقط) هو توجيه
2. تأكد من أن إنشاء الرأس الخارجي يتطابق مع الاتجاه
3. تحقق من صحة عنوان IP و TEID للنزول البعيد

تحقق من QER:

1. تحقق من أن حالة البوابة مفتوحة (0)
2. تحقق من أن MBR ليس صارمًا جدًا

إسقاط الحزم

تحقق من تحديد معدل QER:

1. انتقل إلى القواعد → QERs
2. تحقق من أن MBR كافٍ لحمل الحركة
3. تحقق من أن نمو حجم URR يتطابق مع الإنتاجية المتوقعة

تحقق من إجراء FAR:

1. انتقل إلى القواعد → FARs
2. تحقق من أن الإجراء هو توجيه، وليس إسقاط
3. تحقق من أن التخزين ليس عالقًا في وضع التخزين فقط

مشكلات التخزين

الحزم عالقة في التخزين:

1. انتقل إلى صفحة التخزين
2. تحقق من الطابع الزمني لأقدم حزمة
3. إذا كانت > 30 ثانية، قد يكون الانتقال قد فشل
4. م بتفريغ أو مسح التخزين يدويًا��

5. تعطيل التخزين على FAR

تجاوز التخزين:

1. تحقق من العدد الإجمالي للحزم مقابل الحد الأقصى الإجمالي (الافتراضي 100,000)
2. تحقق من الحزم لكل FAR مقابل الحد الأقصى لكل FAR (الافتراضي 10,000)
3. امسح التخزين إذا كان ممتلئًا
4. تحقق من سبب عدم تعطيل التخزين

URR لا تتبع

عدادات الحجم عند الصفر:

1. تحقق من أن URR ID تشير إلى PDR
2. تحقق من أن الحزم تتطابق مع PDR
3. تحقق من أن FAR (ليس إسقاطها) تقوم بتوجيه الحزم
4. تأكد من أن URR ID موجود في خريطة URR

الحجم لا يتم الإبلاغ عنه إلى SMF:

1. تحقق من تكوين طلب تقرير جلسة PFCP
2. تحقق من مشغلات تقارير URR (عتبات الحجم/الوقت)
3. راجع السجلات لرسائل تقرير جلسة PFCP

الوثائق ذات الصلة

- **دليل عمليات UPF** - نظرة عامة على بنية ومكونات OmniUPF
- **دليل عمليات واجهة الويب** - استخدام لوحة التحكم لعرض الق��اعد
- **دليل المراقبة** - الإحصائيات ومراقبة السعة
- **دليل استكشاف الأخطاء وإصلاحها** - المشكلات الشائعة والتشخيصات

دليل استكشاف أخطاء OmniUPF

جدول المحتويات

1. نظرة عامة
2. أدوات التشخيص
3. مشاكل التثبيت
4. مشاكل التكوين
5. مشاكل ارتباط PFCP
6. مشاكل معالجة الحزم
7. مشاكل eBPF و XDP
8. مشاكل الأداء
9. مشاكل محددة بالهايبرفايزر
10. مشاكل NIC والسائق
11. فشل إنشاء الجلسة
12. مشاكل التخزين المؤقت

نظرة عامة

يوفر هذا الدليل إجراءات استكشاف أخطاء منهجية لمشاكل OmniUPF الشائعة. تتضمن كل قسم الأعراض، وخطوات التشخيص، والأسباب الجذرية، وإجراءات الحل.

قائمة التحقق السريعة للتشخيص

قبل البدء في استكشاف الأخطاء بعمق، تحقق من:

```
# 1. تحقق من تشغيل OmniUPF
systemctl status omniupf

# 2. تحقق من ارتباط PFCP
curl http://localhost:8080/api/v1/upf_pipeline

# 3. تحقق من تحميل خرائط eBPF
ls /sys/fs/bpf/

# 4. تحقق من ارتباط برنامج XDP
ip link show | grep -i xdp

# 5. تحقق من سجلات النواة للأخطاء
dmesg | tail -50
journalctl -u omniupf -n 50
```

أدوات التشخيص

OmniUPF REST واجهة برمجة تطبيقات

تحقق من حالة UPF:

```
curl http://localhost:8080/api/v1/upf_status
```

تحقق من ارتباطات PFCP:

```
curl http://localhost:8080/api/v1/upf_pipeline
```

تحقق من عدد الجلسات:

```
curl http://localhost:8080/api/v1/sessions | jq 'length'
```

تحقق من سعة خريطة eBPF:

```
curl http://localhost:8080/api/v1/map_info
```

تحقق من إحصائيات الحزم:

```
curl http://localhost:8080/api/v1/packet_stats
```

تحقق من إحصائيات XDP:

```
curl http://localhost:8080/api/v1/xdp_stats
```

فحص خريطة eBPF

قائمة جميع خرائط eBPF:

```
ls -lh /sys/fs/bpf/
bpftool map list
```

عرض تفاصيل الخريطة:

```
bpftool map show
bpftool map dump name pdr_map_downlin
```

عد الإدخالات في الخريطة:

```
bpftool map dump name far_map | grep -c "key:"
```

فحص برنامج XDP

تحقق مما إذا كان برنامج XDP مرتبطًا:

```
ip link show eth0 | grep xdp
```

قائمة جميع برامج XDP:

```
bpftool net list
```

عرض تفاصيل برنامج XDP:

```
bpftool prog show
```

تفريغ إحصائيات XDP:

```
bpftool prog dump xlated name xdp_upf_func
```

استكشاف الشبكة

التقاط حركة مرور PFCP على N4 (خطة التحكم):

```
# PFCP يعمل بشكل طبيعي، tcpdump XDP لا تتم معالجته بواسطة
tcpdump -i eth0 -n udp port 8805 -w /tmp/pfcp_traffic.pcap
```

التقاط حركة مرور GTP-U على N3 (يتطلب التقاط خارج النطاق):

```
# تحذير: tcpdump القياسي على مضيف UPF لا يمكنه التقاط الحزم
المعالجة بواسطة XDP!
# تقوم XDP بمعالجة GTP-U النواة حزم ترى أن قبل.

# استخدم التقاط خارج النطاق بدلاً من ذلك:
# 1. TAP الشبكة بين gNB و UPF
# 2. نسخ منفذ التبديل / SPAN لنسخ حركة مرور N3
# 3. نسخ منفذ التبديل الافتراضي إلى VM المحلل

# المراقبة / التحليل على مضيف (ليس على UPF):
# tcpdump -i <mirror_interface> -n udp port 2152 -w
/tmp/n3_capture.pcap

# أو استخدم واجهة برمجة التطبيقات للإحصائيات لعدد الحزم:
curl http://localhost:8080/api/v1/packet_stats
curl http://localhost:8080/api/v1/n3n6_stats
```

مراقبة عدادات الحزم:

```
watch -n 1 'ip -s link show eth0'
```

تحقق من جدول التوجيه:

```
ip route show
ip route get 10.45.0.100  # تحقق من المسار لعنوان IP الخاص بـ UE
```

تحقق من جدول ARP:

```
ip neigh show
```

مشاكل التثبيت

المشكلة: "نظام ملفات eBPF غير مثبت"

الأعراض:

```
ERRO[0000] failed to load eBPF objects: mount bpf filesystem at
/sys/fs/bpf
```

السبب: نظام ملفات eBPF غير مثبت

الحل:

```
# قم بتركيب نظام ملفات eBPF
sudo mount bpffs /sys/fs/bpf -t bpf

# اجعلها دائمة (أضف إلى /etc/fstab)
echo "bpffs /sys/fs/bpf bpf defaults 0 0" | sudo tee -a /etc/fstab

# تحقق من التركيب
mount | grep bpf
```

المشكلة: إصدار النواة قديم جدًا

الأعراض:

```
ERRO[0000] kernel version 5.4.0 is too old, minimum required is
5.15.0
```

السبب: إصدار نواة Linux أقل من الحد الأدنى المطلوب

الحل:

```
# تحقق من إصدار النواة
uname -r

# ترقية النواة (Ubuntu/Debian)
sudo apt update
sudo apt install linux-generic-hwe-22.04
sudo reboot

# تحقق من النواة الجديدة
uname -r  # يجب أن تكون >= 5.15.0
```

المشكلة: اعتماد libbpf مفقود

الأعراض:

```
error while loading shared libraries: libbpf.so.0: cannot open
shared object file
```

السبب: مكتبة libbpf غير مثبتة

الحل:

```
# تثبيت libbpf (Ubuntu/Debian)
sudo apt update
sudo apt install libbpf-dev

# تحقق من التثبيت
ldconfig -p | grep libbpf
```

مشاكل التكوين

المشكلة: ملف التكوين غير صالح

الأعراض:

```
ERRO[0000] unable to read config file: unmarshal errors
```

السبب: خطأ في بناء جملة YAML في ملف التكوين

الحل:

```
# تحقق من بناء جملة YAML
cat config.yml | python3 -c "import yaml, sys;
yaml.safe_load(sys.stdin)"

# القضايا الشائعة:
# - تداخل غير صحيح (استخدم المسافات، وليس علامات التبويب)
# - الفواصل المفقودة بعد المفاتيح
# - سلاسل غير مشفرة تحتوي على أحرف خاصة
# - عناصر القائمة بدون شرطات

# مثال على YAML الصحيح:
cat > config.yml <<EOF
interface_name: [eth0]
xdp_attach_mode: generic
api_address: :8080
pfcp_address: :8805
EOF
```

المشكلة: اسم الواجهة غير موجود

الأعراض:

```
ERRO[0000] interface eth0 not found
```

السبب: الواجهة المكونة غير موجودة

الحل:

```
# قائمة بجميع واجهات الشبكة
ip link show

# تحقق من حالة الواجهة
ip addr show eth0

# إذا كانت الواجهة لها اسم مختلف، قم بتحديث config.yml:
interface_name: [ens1f0]  # استخدم الاسم الفعلي للواجهة

# بالنسبة للآلات الافتراضية، تحقق من نظام تسمية الواجهة
ls /sys/class/net/
```

المشكلة: المنفذ مستخدم بالفعل

الأعراض:

```
ERRO[0000] failed to start API server: address already in use
```

السبب: المنفذ 8080 أو 8805 أو 9090 مرتبط بالفعل بعملية أخرى

الحل:

```
# ابحث عن العملية التي تستخدم المنفذ
sudo lsof -i :8080
sudo netstat -tulpn | grep :8080

# إنهاء العملية المتعارضة
sudo kill <PID>

# أو تغيير منفذ OmniUPF في التكوين
api_address: :8081
pfcp_address: :8806
metrics_address: :9091
```

المشكلة: معرف عقدة PFCP غير صالح

الأعراض:

```
ERRO[0000] invalid pfcp_node_id: must be valid IPv4 address
```

السبب: معرف عقدة PFCP ليس عنوان IPv4 صالحًا

الحل:

```yaml
# استخدم عنوان IP صحيح (ليس اسم المضيف):
pfcp_node_id: 10.100.50.241

# غير صحيح:
# pfcp_node_id: localhost
# pfcp_node_id: upf.example.com
```

مشاكل ارتباط PFCP

المشكلة: لا توجد ارتباطات PFCP تم إنشاؤها

الأعراض:

- واجهة المستخدم على الويب تظهر "لا توجد ارتباطات"
- سجلات SMF تظهر "فشل إعداد ارتباط PFCP"

التشخيص:

```
# 1. تحقق مما إذا كان خادم PFCP يستمع
sudo netstat -ulpn | grep 8805

# 2. تحقق من قواعد جدار الحماية
sudo iptables -L -n | grep 8805
sudo ufw status

# 3. التقاط حركة مرور PFCP
tcpdump -i any -n udp port 8805 -vv

# 4. تحقق من ارتباطات PFCP عبر واجهة برمجة التطبيقات
curl http://localhost:8080/api/v1/upf_pipeline
```

الأسباب الشائعة والحلول:

جدار الحماية يحظر PFCP

الحل:

```
# السماح بحركة مرور PFCP (UDP 8805)
sudo ufw allow 8805/udp
sudo iptables -A INPUT -p udp --dport 8805 -j ACCEPT
```

خاطئ PFCP معرف عقدة

الحل:

```
# تعيين معرف عقدة PFCP إلى عنوان IP الصحيح لواجهة N4
pfcp_node_id: 10.100.50.241  # يجب أن يتطابق مع IP على شبكة N4
```

الشبكة غير قابلة للوصول إلى SMF

الحل:

```
# اختبار الاتصال بـ SMF
ping <SMF_IP>

# تحقق من التوجيه إلى SMF
ip route get <SMF_IP>

# إضافة مسار إذا كان مفقودًا
sudo ip route add <SMF_NETWORK>/24 via <GATEWAY>
```

خاطئ UPF مكون بعنوان SMF

الحل:

- تحقق من تكوين SMF لعنوان UPF
- تأكد من أن SMF يحتوي على IP `pfcp_node_id` الخاص بـ UPF
- تحقق من أن SMF يمكنه التوجيه إلى شبكة N4 الخاصة بـ UPF

المشكلة: فشل نبض PFCP

الأعراض:

```
WARN[0030] PFCP heartbeat timeout for association 10.100.50.10
```

التشخيص:

```
# تحقق من إحصائيات PFCP
curl http://localhost:8080/api/v1/upf_pipeline | jq
'.associations[] | {remote_id, uplink_teid_count}'

# مراقبة سجلات نبض القلب
journalctl -u omniupf -f | grep heartbeat
```

الأسباب والحلول:

فقدان حزم الشبكة

الحل:

```
# تحقق من فقدان الحزم إلى SMF
ping -c 100 <SMF_IP> | grep loss

# إذا كان الفقد مرتفعًا، تحقق من الشبكة:
# - تحقق من حالة الرابط
# - تحقق من صحة التبديل / الموجه
# - تحقق من الازدحام
```

فترة نبض القلب عدوانية جدًا

الحل:

```
# زيادة فترة نبض القلب
heartbeat_interval: 30  # زيادة من 5 إلى 30 ثانية
heartbeat_retries: 5    # زيادة المحاولات
heartbeat_timeout: 10   # زيادة المهلة
```

مشاكل معالجة الحزم

المشكلة: لا توجد حزم تتدفق (عند RX/TX 0 عدادات)

الأعراض:

- حزم RX/TX صفحة الإحصائيات تظهر 0
- لا يمكنه إنشا�� جلسة بيانات UE

التشخيص:

```
# 1. تحقق مما إذا كان برنامج XDP مرتبطًا
ip link show eth0 | grep xdp

# 2. تحقق من أن الواجهة نشطة
ip link show eth0

# 3. تحقق من إحصائيات الحزم (مدرك لـ XDP)
# ملاحظة: tcpdump لا يمكنه رؤية حزم GTP-U المعالجة بواسطة XDP
curl http://localhost:8080/api/v1/packet_stats
```

الحلول:

برنامج XDP غير مرتبط

الحل:

```
# أعد تشغيل OmniUPF لإعادة ربط XDP
sudo systemctl restart omniupf

# تحقق من الارتباط
ip link show eth0 | grep xdp
bpftool net list
```

الواجهة غير نشطة أو لا يوجد رابط

الحل:

```
# قم بتشغيل الواجهة
sudo ip link set eth0 up

# تحقق من حالة الرابط
ethtool eth0 | grep "Link detected"

# إذا كان الرابط غير نشط، تحقق من الاتصال الفيزيائي أو تكوين الشبكة للآلة الافتراضية
```

الواجهة المكونة خاطئة

الحل:

```
# تحديث config.yml مع الواجهة الصحيحة
interface_name: [ens1f0]  # استخدم الاسم الفعلي للواجهة من 'ip link show'
```

المش��لة: تم استلام حزم ولكن لم يتم إعادة توجيهها (معدل إسقاط مرتفع)

الأعراض:

- لا TX تزداد ولكن عدادات RX عدادات
- معدل الإسقاط% > 1

التشخيص:

```
# تحقق من إحصائيات الإسقاط
curl http://localhost:8080/api/v1/xdp_stats | jq '.drop'

# تحقق من إحصائيات التوجيه
curl http://localhost:8080/api/v1/packet_stats | jq '.route_stats'

# مراقبة إسقاط الحزم
watch -n 1 'curl -s http://localhost:8080/api/v1/packet_stats | jq ".total_rx, .total_tx, .total_drop"'
```

الأسباب الشائعة:

لا يوجد تطابق PDR (TEID أو UE IP غير معروف)

الحل:

```
# تحقق مما إذا كانت الجلسات موجودة
curl http://localhost:8080/api/v1/sessions

# إذا لم تكن هناك جلسات، تحقق من:
# - تم إنشاء ارتباط PFCP
# - قد أنشأ جلسات SMF
# - كان إنشاء الجلسة ناجحًا

# تحقق من إدخالات خريطة PDR
bpftool map dump name pdr_map_teid_ip | grep -c key
bpftool map dump name pdr_map_downlin | grep -c key
```

فشل التوجيه

الحل:

```
# تحقق من فشل البحث في FIB
curl http://localhost:8080/api/v1/packet_stats | jq '.route_stats'

# اختبار التوجيه لعنوان IP الخاص بـ UE
ip route get 10.45.0.100

# إضافة مسار مفقود
sudo ip route add 10.45.0.0/16 dev eth1  # توجيه مجموعة UE إلى N6
```

تحديد معدل QER

الأعراض:

- معدل النقل أقل من المتوقع
- حركة المرور مقيدة بمعدل معين
- تظهر سلوك هضمي URR عدادات حجم
- تزداد خلال انفجارات الحركة XDP عدادات إسقاط

التشخيص:

1. **تحقق من MBR المكون للجلسة**:

```
# ابحث عن معرف QER للجلسة
curl http://localhost:8080/api/v1/pfcp_sessions | jq
'.data[] | select(.ue_ip == "10.45.0.1")'

# ابحث عن تكوين QER
curl http://localhost:8080/api/v1/qer_map | jq '.data[] |
select(.qer_id == 1)'
```

2. **تحقق من حالة البوابة**:

```
# يجب أن تكون حالة البوابة 0 (مفتوحة) لكل من الرفع والهبوط
curl http://localhost:8080/api/v1/qer_map | jq '.data[] |
{qer_id, ul_gate: .ul_gate_status, dl_gate:
.dl_gate_status}'
```

3. **احسب معدل النقل الفعلي من URR**:

```
# استعلام عن عدادات حجم URR في نقطتين زمنيتين
curl http://localhost:8080/api/v1/urr_map | jq '.data[] |
select(.urr_id == 0)'

# احسب معدل النقل (يدوي):
# throughput_kbps = (volume_delta_bytes × 8) /
time_delta_seconds / 1000
```

4. **قارن MBR مقابل معدل النقل الفعلي**:
    - MBR (بسبب الحمل الزائد) ≈ 95-98% من معدل النقل المتوقع (للبروتوكول)
    - إذا كان معدل النقل أقل بكثير من MBR، تحقق من اختناقات أخرى
    - إذا كان معدل النقل يتطابق تمامًا مع MBR، فإن تحديد المعدل يعمل كما هو متوقع

الحل:

- **إذا كان MBR منخفضًا جدًا**: اطلب من SMF تحديث QER مع MBR أعلى عبر تعديل جلسة PFCP
- **إذا كانت البوابة مغلقة**: تحقق من سبب إغلاق SMF (سياسة، حصة، أو خطأ) للبوابة

- **إذا كان تحديد المعدل غير متوقع**: تحقق من تكوين سياسة QoS وملف تعريف SMF

فهم تنفيذ MBR:

يستخدم OmniUPF خوارزمية نافذة منزلقة لفرض حدود MBR في مسار eBPF. بدقة نانو ثانية.
للحصول على شرح مفصل عن MBR انظر دليل إدارة القواعد - آلية تنفيذ:

- كيف تحدد حجم الحزمة ومعدلها قرارات الإسقاط
- لماذا يختلف معدل النقل الملاحظ عن MBR المكون
- تحديد المعدل في كل اتجاه (رفع / هبوط)
- سلوك نافذة منزلقة لمدة 5 مللي ثانية

السيناريوهات الشائعة:

- **مكالمات VoIP تسقط**: تحقق مما إذا كان MBR كافيًا لمعدل بت الترميز (G.711 = ~80 كيلوبت في الثانية)
- **تخزين مؤقت لبث الفيديو**: تأكد من أن MBR > معدل بت الفيديو + الحمل الزائد (1080p = ~5-10 ميغابت في الثانية)
- **حركة مرور انفجارية**: يسمح بانفجارات صغيرة ضمن نافذة 5 مللي ثانية، يتم تحديد معدل الحركة المستدامة

المشكلة: حركة مرور أحادية الاتجاه (الرفع يعمل، الهبوط لا يعمل)

الأعراض:

- حزم N3 TX ولكن لا توجد حزم N3 RX (مشكلة هبوط)
- حزم N6 TX ولكن لا توجد حزم N6 RX (مشكلة رفع)

التشخيص:

```
# تحقق من إحصائيات واجهة N3/N6 (طريقة مدركة لـ XDP)
curl http://localhost:8080/api/v1/n3n6_stats
curl http://localhost:8080/api/v1/packet_stats

# ملاحظة: tcpdump القياسي لا يمكنه التقاط حركة مرور GTP-U المعالجة بواسطة XDP
# استخدم واجهة برمجة التطبيقات للإحصائيات أو xdpdump لتحليل الحركة
# انظر قسم "التقاط الحزم باستخدام XDP" للحصول على التفاصيل
```

فشل الرفع (RX N3، لا TX N6):

السبب: لا يوجد إجراء FAR أو مشكلة توجيه إلى N6

الحل:

```
# تحقق من أن إجراء FAR لديه FORWARD
curl http://localhost:8080/api/v1/sessions | jq '.[].fars[] |
select(.applied_action == 2)'

# تحقق من وجود مسار N6
ip route get 8.8.8.8  # اختبار المسار إلى الإنترنت

# إضافة مسار افتراضي إذا كان مفقودًا
sudo ip route add default via <N6_GATEWAY> dev eth1
```

فشل الهبوط (RX N6، لا TX N3):

السبب: لا يوجد هابط GTP أو عدم وجود تغليف PDR

الحل:

```
# تحقق من وجود PDR هابط لعنوان IP الخاص بـ UE
curl http://localhost:8080/api/v1/sessions | jq '.[].pdrs[] |
select(.pdi.ue_ip_address)'

# تحقق من أن FAR لديه OUTER_HEADER_CREATION
curl http://localhost:8080/api/v1/sessions | jq '.[].fars[] |
.outer_header_creation'

# تحقق من إمكانية الوصول إلى gNB
ping <GNB_N3_IP>
```

مشاكل XDP و eBPF

واختيار الوضع، واستكشاف الأخطاء، انظر دليل للحصول على تفاصيل تكوين XDP، أوضاع XDP.

المشكلة: فشل تحميل برنامج XDP

الأعراض:

```
ERRO[0000] failed to load XDP program: invalid argument
```

التشخيص:

```
# تحقق من دعم النواة لـ XDP
grep XDP /boot/config-$(uname -r)

# يجب أن تظهر:
# CONFIG_XDP_SOCKETS=y
# CONFIG_BPF=y
# CONFIG_BPF_SYSCALL=y

# تحقق من dmesg مفصل للحصول على خطأ
dmesg | grep -i bpf
```

الأسباب والحلول:

النواة تفتقر إلى دعم XDP

الحل:

```
# أو الترقية إلى نواة أحدث XDP إعادة بناء النواة مع دعم
# Ubuntu 22.04+ لديها XDP مفعلة بشكل افتراضي
sudo apt install linux-generic-hwe-22.04
sudo reboot
```

فشل التحقق من برنامج XDP

الحل:

```
# تحقق من سجلات OmniUPF لأخطاء المحقق
journalctl -u omniupf | grep verifier

# القضايا الشائعة:
# - تعقيد eBPF (زيادة حدود النواة) يتجاوز الحدود
# - وصول غير صالح إلى الذاكرة (خطأ في كود eBPF)

# زيادة مستوى سجل المحقق eBPF للتصحيح
sudo sysctl kernel.bpf_stats_enabled=1
```

المشكلة: زيادة عدد الإجهاضات في XDP

الأعراض:

- إحصائيات XDP تظهر aborted > 0
- زيادة إسقاط الحزم

التشخيص:

```
# تحقق من عدد الإجهاضات في XDP
curl http://localhost:8080/api/v1/xdp_stats | jq '.aborted'

# مراقبة إحصائيات XDP
watch -n 1 'curl -s http://localhost:8080/api/v1/xdp_stats'
```

خطأ في وقت التشغيل eBPF **السبب**: واجه برنامج

الحل:

```
# تحقق من سجلات النواة لأخطاء eBPF
dmesg | grep -i bpf

# أعد تشغيل OmniUPF لإعادة تحميل برنامج eBPF
sudo systemctl restart omniupf

# إذا استمرت المشكلة، قم بتمكين تسجيل eBPF (يتطلب إعادة بناء):
# بناء OmniUPF مع BPF_ENABLE_LOG=1
```

المشكلة: خريطة eBPF ممتلئة (استهلاك السعة)

الأعراض:

- فشل إنشاء الجلسة
- سعة الخريطة عند 100%

التشخيص:

```
# تحقق من سعة الخريطة
curl http://localhost:8080/api/v1/map_info | jq '.[] | {map_name,
capacity, used, usage_percent}'

# تحديد الخرائط المملوءة
curl http://localhost:8080/api/v1/map_info | jq '.[] |
select(.usage_percent > 90)'
```

التخفيف الفوري:

```
# 1. تحديد الجلسات القديمة
curl http://localhost:8080/api/v1/sessions | jq '.[] | {seid,
uplink_teid, created_at}'

# 2. حذف الجلسات القديمة اطلب من SMF
# (الإدارية أو واجهة برمجة التطبيقات SMF عبر واجهة)

# 3. مراقبة انخفاض استخدام الخريطة
watch -n 5 'curl -s http://localhost:8080/api/v1/map_info | jq ".
[] | select(.map_name==\"pdr_map_downlin\") | .usage_percent"'
```

الحل طويل الأجل:

```yaml
# زيادة سعة الخريطة في config.yml
max_sessions: 200000  # زيادة من 100000

# أو تعيين أحجام الخرائط الفردية
pdr_map_size: 400000
far_map_size: 400000
qer_map_size: 200000
```

مهم: تغيير أحجام الخرائط يتطلب إعادة تشغيل OmniUPF و**يمسح جميع الجلسات الحالية**.

مشاكل الأداء

المشكلة: معدل نقل منخفض (أقل من المتوقع)

الأعراض:

- معدل النقل < 1 جيجابت في الثانية على الرغم من NIC القادر
- استخدام CPU مرتفع

التشخيص:

```
# تحقق من معدل الحزم
curl http://localhost:8080/api/v1/packet_stats | jq '.total_rx,
.total_tx'

# تحقق من إحصائيات NIC
ethtool -S eth0 | grep -i drop

# تحقق من وضع XDP
ip link show eth0 | grep xdp
```

الحلول:

استخدام وضع XDP العام

الحل:

```
# التبديل إلى الوضع الأصلي لأداء أفضل
xdp_attach_mode: native  # يتطلب NIC / سائق يدعم XDP
```

اختناق أحادي النواة

الحل:

```
# تمكين RSS (توزيع جانب الاستلام) على NIC
ethtool -L eth0 combined 4  # استخدم 4 قوائم RX/TX

# تحقق من تمكين RSS
ethtool -l eth0

# تثبيت المقاطعات على وحدات المعالجة المركزية المحددة
# انظر /proc/interrupts واستخدم irqbalance أو التوافق اليدوي
```

ازدهار التخزين المؤقت

الحل:

```yaml
# تقليل حدود التخزين المؤقت لتقليل الكمون
buffer_max_packets: 5000
buffer_packet_ttl: 15
```

المشكلة: الكمون العالي

الأعراض:

- كمون ping > 50 مللي ثانية
- تدهور تجربة المستخدم

التشخيص:

```
# اختبار الكمون إلى UE
ping -c 100 <UE_IP> | grep avg

# تحقق من الحزم المخزنة
curl http://localhost:8080/api/v1/upf_buffer_info | jq
'.total_packets_buffered'

# تحقق من أداء ذاكرة التخزين المؤقت للتوجيه
curl http://localhost:8080/api/v1/packet_stats | jq '.route_stats'
```

الحلول:

الحزم المخزنة بشكل مفرط

الحل:

```
# تحقق من سبب تخزين الحزم
curl http://localhost:8080/api/v1/upf_buffer_info | jq '.buffers[]
| {far_id, packet_count, direction}'

# مسح التخزين المؤقت إذا كان عالقًا
# (إعادة تشغيل OmniUPF أو تحفيز تعديل جلسة PFCP لتطبيق FAR)
```

كمون بحث FIB

الحل:

```
# (خيار وقت البناء) تأكد من تمكين ذاكرة التخزين المؤقت للتوجيه
# البناء مع BPF_ENABLE_ROUTE_CACHE=1

# تحسين جدول التوجيه
# استخدم مسارات أقل، وأكثر تحديدًا بدلاً من العديد من المسارات الصغيرة
```

المشكلة: إسقاط الحزم تحت الحمل

الأعراض:

- معدل الإسقاط يزيد مع الحركة
- أخطاء RX على NIC

التشخيص:

```
# تحقق من أخطاء NIC
ethtool -S eth0 | grep -E "drop|error|miss"

# تحقق من حجم حلقة التخزين
ethtool -g eth0

# مراقبة الإسقاط في الوقت الحقيقي
watch -n 1 'ethtool -S eth0 | grep -E "drop|miss"'
```

الحل:

```
# زيادة حجم حلقة RX
ethtool -G eth0 rx 4096

# زيادة حجم حلقة TX
ethtool -G eth0 tx 4096

# تحقق من الإعدادات الجديدة
ethtool -g eth0
```

مشاكل محددة بالهايبرفايزر

للحصول على تعليمات تكوين خطوة بخطوة للهايبرفايزر، انظر دليل أوضاع XDP.

Proxmox: XDP لا يعمل في VM

الأعراض:

- XDP في الوضع الأصلي لا يمكن ربط برنامج
- يعمل فقط الوضع العام

السبب: VM تستخدم الشبكة الموصلة بدون SR-IOV

الحل:

الخيار 1: استخدم الوضع العام (الأبسط)

```
xdp_attach_mode: generic
```

الخيار 2: تكوين تمرير SR-IOV

```
# على مضيف Proxmox:
# 1. تمكين IOMMU
nano /etc/default/grub
# أضف: intel_iommu=on iommu=pt
update-grub
reboot

# 2. إنشاء VFs
echo 4 > /sys/class/net/eth0/device/sriov_numvfs

# 3. تخصيص VF إلى VM في واجهة Proxmox
# جهاز → إضافة → الأجهزة PCI → تحديد VF

# في VM:
interface_name: [ens1f0]  # SR-IOV VF
xdp_attach_mode: native
```

VMware: وضع promiscuous مطلوب

الأعراض:

- الحزم لا يتم استلامها بواسطة OmniUPF

السبب: vSwitch يمنع عناوين MAC غير المطابقة

الحل:

```
# تمكين وضع promiscuous على vSwitch (في vSphere Client):
# 1. اختر vSwitch → تحرير الإعدادات
# 2. وضع → الأمان promiscuous: قبول
# 3. الأمان → تغييرات عنوان MAC: قبول
# 4. الأمان → نقل مزيف: قبول
```

VirtualBox: أداء منخفض جدًا

الأعراض:

- معدل النقل < 100 ميغابت في الثانية

السبب: VirtualBox لا تدعم SR-IOV أو XDP الأصلي

الحل:

```
# استخدم الوضع العام (الخيار الوحيد)
xdp_attach_mode: generic

# تحسين إعدادات VirtualBox:
# - استخدم محول VirtIO-Net (إذا كان متاحًا)
# - تمكين promiscuous في وضع "السماح للجميع"
# - تخصيص المزيد من النوى CPU للآلة الافتراضية
# - استخدم الشبكة الموصلة بدلاً من NAT

# للحصول على أداء أفضل ضع في اعتبارك الانتقال إلى KVM/Proxmox
```

مشاكل NIC والسائق

المشكلة: سائق NIC لا يدعم XDP

الأعراض:

```
ERRO[0000] failed to attach XDP program: operation not supported
```

التشخيص:

```
# تحقق من سائق NIC
ethtool -i eth0 | grep driver

# تحقق مما إذا كان السائق يدعم XDP
modinfo <driver_name> | grep -i xdp

# قائمة الواجهات القابلة لـ XDP
ip link show | grep -B 1 "xdpgeneric\|xdpdrv\|xdpoffload"
```

الحل:

الخيار 1: استخدم الوضع العام

```
xdp_attach_mode: generic
```

الخيار 2: تحديث سائق NIC

```
# تحقق من تحديثات السائق (Ubuntu)
sudo apt update
sudo apt install linux-modules-extra-$(uname -r)

# أو تثبيت سائق محدد من البائع
# مثال لـ Intel:
# تحميل من https://downloadcenter.intel.com/
```

الخيار 3: استبدال NIC

```
# استخدم NIC يدعم XDP:
# - Intel X710، E810
# - Mellanox ConnectX-5، ConnectX-6
# - Broadcom BCM57xxx (bnxt_en driver)
```

المشكلة: سائق يتعطل أو يتسبب في ذبذبة النواة

الأعراض:

- ذبذبة النواة بعد ربط XDP
- يتوقف عن الاستجابة NIC

التشخيص:

```
# تحقق من سجلات النواة
dmesg | tail -100

# تحقق من أخطاء السائق
journalctl -k | grep -E "BUG:|panic:"
```

الحل:

```
# 1. تحديث النواة والسائقين
sudo apt update
sudo apt upgrade
sudo reboot

# 2. تعطيل XDP offload (استخدم الأصلي فقط)
xdp_attach_mode: native

# 3. استخدم الوضع العام كحل بديل
xdp_attach_mode: generic

# 4. الإبلاغ عن خطأ إلى بائع NIC أو فريق نواة Linux
```

فشل إنشاء الجلسة

المشكلة: فشل إنشاء الجلسة

الأعراض:

- SMF يبلغ عن فشل إنشاء الجلسة
- UE لا يمكنه إنشاء جلسة PDU

انظر مرجع رموز سبب PFCP للسيناريوهات الشائعة للفشل والحلول.

التشخيص:

```
# تحقق من سجلات OmniUPF لأخطاء الجلسة
journalctl -u omniupf | grep -i "session establishment"

# تحقق من عدد جلسات PFCP
curl http://localhost:8080/api/v1/sessions | jq 'length'

# التقاط حركة مرور PFCP أثناء إنشاء الجلسة
tcpdump -i any -n udp port 8805 -w /tmp/pfcp_session.pcap
```

الأسباب الشائعة:

سعة الخريطة ممتلئة

الحل:

```
# تحقق من استخدام الخريطة
curl http://localhost:8080/api/v1/map_info | jq '.[] |
select(.usage_percent > 90)'

# زيادة السعة (ممتلئة أعلاه eBPF انظر قسم خريطة)
```

معلمات PDR/FAR غير صالحة

الحل:

```
# تحقق من سجلات OmniUPF لأخطاء التحقق
journalctl -u omniupf | grep -E "invalid|error" | tail -20

# القضايا الشائعة:
# - عنوان IP الخاص بـ UE (أو مكرر) 0.0.0.0 غير صالح
# - TEID (أو مكرر) 0 غير صالح
# - FAR مفقود لـ PDR
# - إجراء FAR غير صالح

# تحقق من تكوين SMF ومعلمات الجلسة
```

ميزة غير مدعومة (UEIP/FTUP)

الحل:

```yaml
# تمكين الميزات المطلوبة إذا لزم الأمر
feature_ueip: true  # تخصيص IP لـ UE بواسطة UPF
ueip_pool: 10.60.0.0/16

feature_ftup: true  # تخصيص F-TEID بواسطة UPF
teid_pool: 100000
```

مشاكل التخزين المؤقت

المشكلة: حزم عالقة ف�� التخزين المؤقت

الأعراض:

- زيادة عدد الحزم المخزنة
- الحزم لا يتم تسليمها بعد النقل

التشخيص:

```
# تحقق من إحصائيات التخزين المؤقت
curl http://localhost:8080/api/v1/upf_buffer_info

# تحقق من التخزين المؤقت الفردي لـ FAR
curl http://localhost:8080/api/v1/upf_buffer_info | jq '.buffers[]
| {far_id, packet_count, oldest_packet_ms}'

# مراقبة حجم التخزين المؤقت
watch -n 5 'curl -s http://localhost:8080/api/v1/upf_buffer_info |
jq ".total_packets_buffered"'
```

الأسباب والحلول:

لم يتم تحديثه أبدًا إلى FORWARD FAR

السبب: لم ترسل SMF تعديل جلسة PFCP لتطبيق FAR

الحل:

```
# تحقق من حالة FAR
curl http://localhost:8080/api/v1/sessions | jq '.[].fars[] |
{far_id, applied_action}'

# الإجراء BUFF = 1 (التخزين المؤقت)
# الإجراء FORW = 2 (إعادة التوجيه)

# إذا كان عالقًا في حالة BUFF، اطلب من SMF:
# - إرسال طلب تعديل جلسة PFCP
# - تحديث FAR بإجراء FORW
```

انتهاء صلاحية TTL للتخزين المؤقت

السبب: الحزم انتهت صلاحيتها قبل تحديث FAR

الحل:

```
# زيادة TTL للتخزين المؤقت
buffer_packet_ttl: 60  # زيادة من 30 إلى 60 ثانية
```

تجاوز التخزين المؤقت

السبب: عدد كبير جدًا من الحزم المخزنة لكل FAR

الحل:

```
# زيادة حدود التخزين المؤقت
buffer_max_packets: 20000  # لكل FAR
buffer_max_total: 200000   # الحد العالمي
```

استكشاف الأخطاء المتقدمة

تمكين تسجيل الأخطاء

```
logging_level: debug  # trace | debug | info | warn | error
```

```
# إعادة تشغيل OmniUPF مع تسجيل الأخطاء
sudo systemctl restart omniupf

# مراقبة السجلات في الوقت الحقيقي
journalctl -u omniupf -f --output cat
```

تتبع برنامج eBPF

```
# تتبع تنفيذ برنامج eBPF (يتطلب bpftrace)
sudo bpftrace -e 'tracepoint:xdp:* { @[probe] = count(); }'

# تتبع عمليات الخريطة
sudo bpftrace -e 'tracepoint:bpf:bpf_map_lookup_elem {
printf("%s\n", str(args->map_name)); }'
```

التقاط الحزم باستخدام XDP

فهم قيود التقاط الحزم باستخدام XDP:

تقوم XDP بمعالجة الحزم **قبل** كومة الشبكة في النواة، لذا فإن `tcpdump` القياسي **لا يمكنه رؤية حركة المرور المعالجة بواسطة XDP**. يتم معالجة حزم GTP-U (منفذ UDP 2152) بواسطة N3 بواسطة XDP ولن تظهر في tcpdump على مضيف UPF.

طرق موصى بها لتحليل الحركة:

```
# الطريقة 1: استخدم واجهة برمجة التطبيقات للإحصائيات للمراقبة (موصى بها)
curl http://localhost:8080/api/v1/xdp_stats
curl http://localhost:8080/api/v1/packet_stats | jq
curl http://localhost:8080/api/v1/n3n6_stats

# الطريقة 2: التقاط حركة مرور PFCP (غير متأثرة بـ XDP)
tcpdump -i any -n udp port 8805 -w /tmp/pfcp.pcap

# الطريقة 3: التقاط الحزم خارج النطاق (موصى بها لـ GTP-U)
# استخدم TAP الشبكة أو نسخ منفذ التبديل لالتقاط الحركة
# أمثلة:
# - TAP فيزيائي بين gNB و UPF
# - نسخ منفذ التبديل SPAN لنسخ حركة مرور N3 إلى المحلل
# - نسخ منفذ التبديل الافتراضي في الهايبرفايزر
#
# على مضيف الالتقاط (ليس UPF):
# tcpdump -i <mirror_interface> -n udp port 2152 -w
/tmp/n3_mirror.pcap
```

أمثلة إعداد الالتقاط خارج النطاق:

الشبكة الفيزيائية:

```
# استخدم TAP الشبكة أو قم بتكوين نسخ منفذ التبديل
# تكوين مثال: SPAN على مفتاح Cisco
(config)# monitor session 1 source interface Gi1/0/1
(config)# monitor session 1 destination interface Gi1/0/24

# على مضيف المراقبة المتصل بـ Gi1/0/24:
tcpdump -i eth0 -n udp port 2152 -w /tmp/n3_capture.pcap
```

البيئة الافتراضية (VMware، KVM، إلخ):

```
# تكوين نسخ منفذ التبديل الافتراضي لإرسال حركة مرور UPF إلى VM
المحلل
# مختلف VM على tcpdump مع Linux مثال: جسر
# إلى واجهة UPF الخاصة بـ N3 على الهايبرفايزر، قم بنسخ واجهة
المحلل

# المحلل على VM:
tcpdump -i eth1 -n udp port 2152 -w /tmp/n3_virtual.pcap
```

لماذا يلزم الالتقاط خارج النطاق:

- تقوم XDP بتجاوز كومة الشبكة في النواة تمامًا
- أو الأجهزة NIC تتم معالجة الحزم في سائق
- يرى tcpdump المستند إلى المضيف الحزم بعد معالجة XDP (متأخر جدًا)
- يرى الالتقاط خارج النطاق حركة المرور الخام قبل م��الجة UPF

ما يمكنك التقاطه على مضيف UPF:

- XDP خطة التحكم، غير معالجة بواسطة - PFCP (UDP 8805) حركة مرور
- استجابات واجهة برمجة التطبيقات والإحصائيات
- XDP خطة البيانات، معالجة بواسطة - GTP-U (UDP 2152) حركة مرور

الحصول على المساعدة

إذا لم تحل خطوات استكشاف الأخطاء مشكلتك:

1. **جمع معلومات التشخيص**:

```
# معلومات النظام
uname -a
cat /etc/os-release

# معلومات OmniUPF
curl http://localhost:8080/api/v1/upf_status
curl http://localhost:8080/api/v1/map_info
curl http://localhost:8080/api/v1/packet_stats

# السجلات
journalctl -u omniupf --since "1 hour ago" >
/tmp/omniupf.log
dmesg > /tmp/dmesg.log

# معلومات الشبكة
ip addr > /tmp/network.txt
ip route >> /tmp/network.txt
ethtool eth0 >> /tmp/network.txt
```

2. مع **الإبلاغ عن المشكلة**:
    - إصدار OmniUPF
    - إصدار نواة Linux
    - مخطط الطوبولوجيا الشبكية
    - ملف التكوين (احذف المعلومات الحساسة)
    - مقتطفات السجل ذات الصلة
    - خطوات لإعادة الإنتاج

الوثائق ذات الصلة

- **دليل التكوين** - معلمات التكوين والأمثلة
- **دليل العمارة** - تفاصيل eBPF/XDP وضبط الأداء
- **دليل المراقبة** - الإحصائيات والسعة والتنبيه
- **مرجع المقاييس** - مقاييس Prometheus لاستكشاف الأخطاء

- **رموز سبب PFCP** - رموز خطأ PFCP واستكشاف الأخطاء
- **دليل إدارة القواعد** - مفاهيم PDR و FAR و QER و URR
- **دليل العمليات** - عمارة UPF ونظرة عامة

OmniUPF / حديقة مسورة إعادة توجيه خارج الرصيد

جدول المحتويات

1. نظرة عامة
2. العمارة
3. تدفق إشارات PFCP
4. كشف البوابة الأسرية
5. التكوين
6. إدارة القائمة البيضاء
7. روابط إعادة التوجيه لكل جلسة
8. API
9. مقاييس بروميثيوس
10. استكشاف الأخطاء وإصلاحها

نظرة عامة

تتيح ميزة الحديقة المسورة **تطبيق خارج الرصيد بشكل أصلي** مباشرة في UPF، مما يلغي الحاجة إلى أنظمة تطبيق خارجية (قوائم عناوين MikroTik، قواعد المانجل، DNAT).

يحتوي على FAR مع PFCP تعديل جلسة SMF عندما ينفد رصيد المشترك، يرسل `redirect_information`. يقوم OmniUPF باعتراض كل حركة المرور لتلك الجلسة في مساحة المستخدم ويطبق تجربة بوابة أسرية:

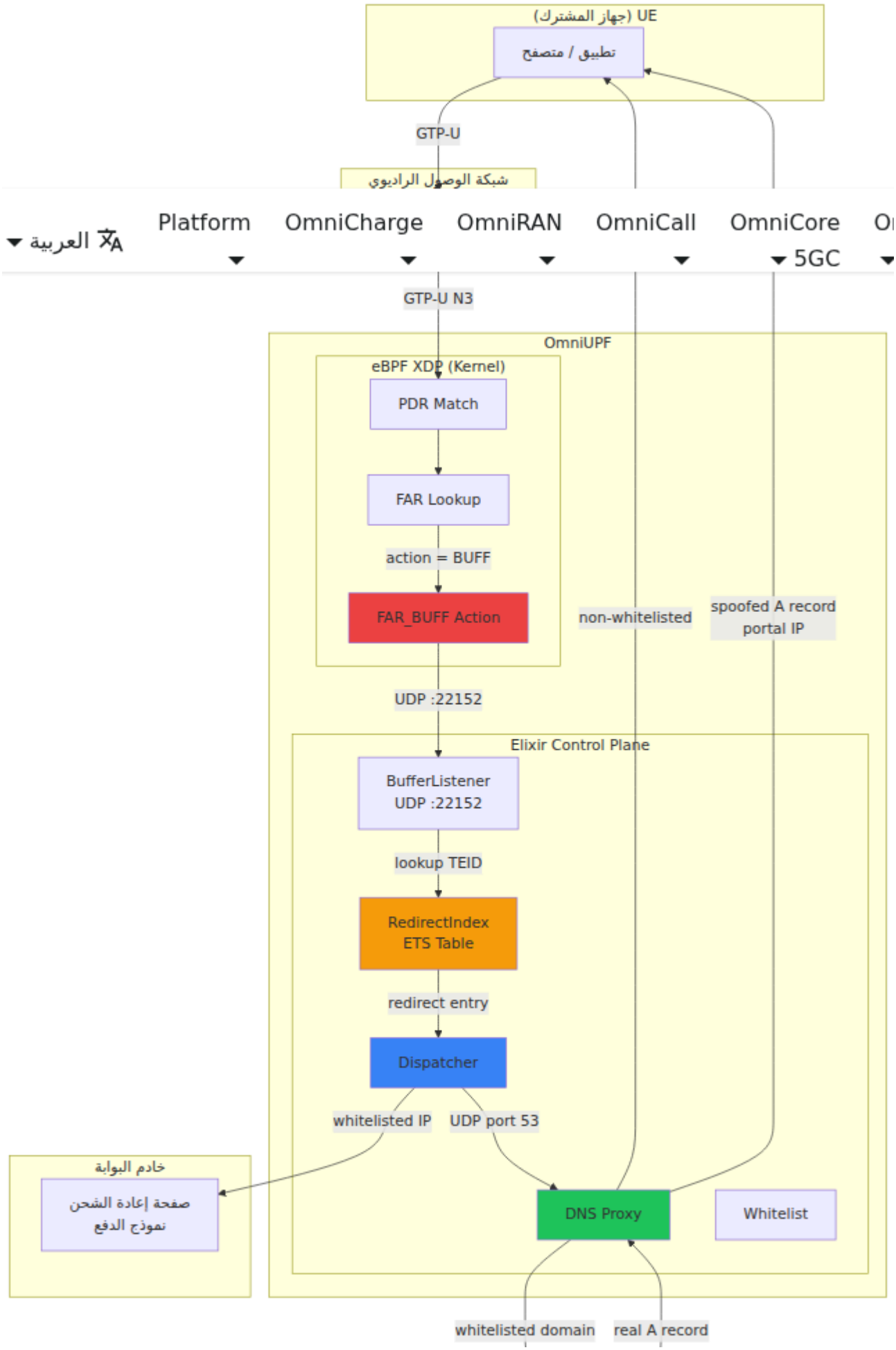
- **استعلامات DNS** يتم تزوير لإرجاع عنوان IP لخادم البوابة، مما يؤدي إلى كشف البوابة الأسرية على جميع الأجهزة الرئيسية (Apple، Android، Windows)
- **الحركة إلى البوابة وعناوين IP المدرجة في القائمة البيضاء** يتم توجيه بشكل طبيعي حتى يتمكن المشترك من إعادة شحن رصيده (CAPTCHA معالجات الدفع، خدمات)
- **تُسقط جميع الحركة الأخرى** بصمت

يرى المشترك موجه بوابة أسرية ويتم توجيهه إلى صفحة إعادة الشحن/الدفع. بمجرد استعادة الرصيد، يقوم SMF بتحديث FAR إلى FORW مرة أخرى على العادية التوجيه عملية وتستأنف الفور.

نقاط التصميم الرئيسية

- **لا حاجة لتغييرات eBPF** -- يعيد استخدام إجراء `FAR_BUFF` الحالي لإعادة توجيه الحزم إلى مساحة المستخدم
- **إعادة توجيه لكل جلسة** -- يمكن أن تحتوي كل جلسة على عنوان IP مختلف للبوابة في SMF `redirect_information` IE ورابط إعادة توجيه، يتم تحديده بواسطة
- **تعيش القائمة البيضاء على UPF** -- يقول SMF؛ فقط "إعادة توجيه هذه الجلسة" يقرر UPF ما هي الحركة التي يُسمح لها بالمرور
- **اعتراض فقط على الرفع** -- تبقى FAR في الاتجاه الهابط FORW حتى تصل استجابات البوابة إلى UE عبر مسار GTP العادي

العمارة


UE (جهاز المشترك)
تطبيق / متصفح
GTP-U
شبكة الوصول الراديوي
العربية
Platform
OmniCharge
OmniRAN
OmniCall
OmniCore
5GC
GTP-U N3
OmniUPF
eBPF XDP (Kernel)
PDR Match
FAR Lookup
action = BUFF
FAR_BUFF Action
non-whitelisted
spoofed A record
portal IP
UDP :22152
Elixir Control Plane
BufferListener
UDP :22152
lookup TEID
RedirectIndex
ETS Table
redirect entry
Dispatcher
whitelisted IP
UDP port 53
خادم البوابة
صفحة إعادة الشحن
نموذج الدفع
DNS Proxy
Whitelist
whitelisted domain
real A record

تدفق الحزم

1. **يرسل UE حزمة رفع** (استعلام DNS، طلب HTTP، إلخ) عبر GTP-U إلى UPF
2. **تطابق eBPF PDR** يجد PDR المطابق، ويبحث عن FAR
3. **إجراء FAR هو BUFF** (عندما يكون إعادة التوجيه نشطًا تم تجاوزها من FORW) -- ترسل eBPF الحزمة إلى BufferListener على منفذ UDP 22152
4. **BufferListener** يستخرج TEID، ويتحقق من جدول RedirectIndex ETS
5. إذا كان TEID في RedirectIndex: **Dispatcher** يعالج حزمة IP الداخلية
6. **شجرة القرار**:
   - استعلام DNS لنطاق غير مدرج في القائمة البيضاء: تزوير سج A مع عنوان IP للبوابة ��
   - استعلام DNS لنطاق مدرج في القائمة البيضاء: توجيه إلى المحلّل الحقيقي، تخزين عناوين IP المحلولة
   - حركة المرور إلى عنوان IP المدرج في القائمة البيضاء: IP للبوابة أو عنوان IP توجيه عبر مقبس خام
   - كل شيء آخر: إسقاط بصمت
7. **استجابات DNS/GTP-U** تُرسل مرة أخرى إلى UE عبر مسار GTP-U الهابط (مغلفة مع DL TEID وإرسالها إلى gNB)

معالجة حلقة N9 (جلسة SGW + PGW مزدوجة)

في نشرات EPC 4G، يعمل OmniUPF غالبًا كـ **SGW-U و PGW-U** على **في نفس الوقت** منفصلة. تحتوي جلسة PFCP بإنشاء جلسة PGW-C و SGW-C نفس العقدة. يقوم SGW على N9 بتنفيذ تطبيق سياسة المشترك الفعلي PGW. تقوم جلسة (N3 حلقة عبر واجهة) الرفع PGW لجلسة TEID التي تعيد توجيه الحزم إلى FAR.

عند تفعيل إعادة توجيه حديقة مسورة عبر `POST /v1/walled_garden`، تكتشف منطق التفعيل هذه الطوبولوجيا تلقائيًا:

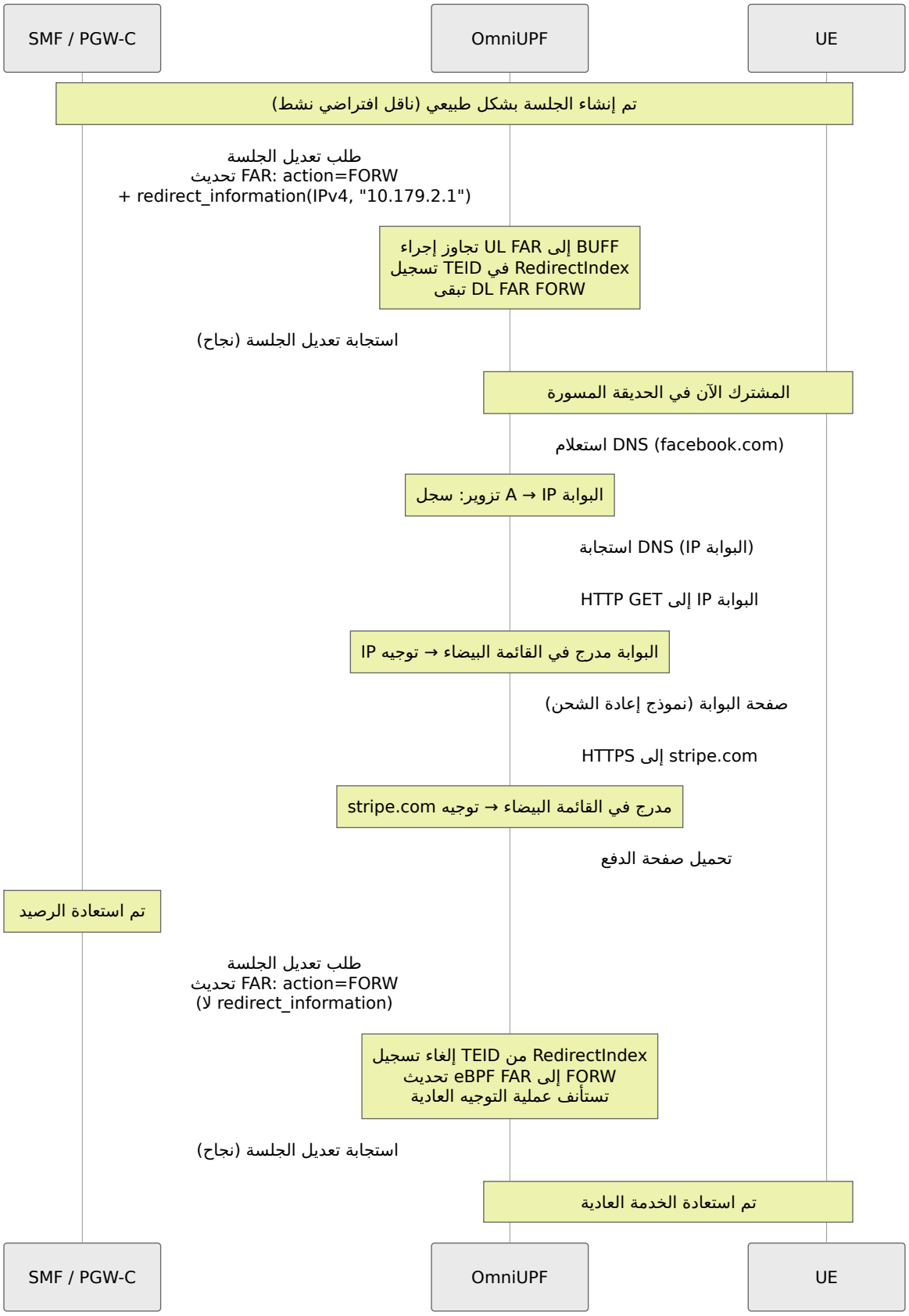
1. يتم تحديد **جلسة PGW TEID** من PDRs الرفع لـ SEID المستهدف.
2. تحتوي (جلسة SGW) للعثور على جلسة PFCP ي��وم الكود بفحص جميع جلسات ولديها PGW UL PDRs الخاص بها TEIDs مطابقًا لأحد `teid` يكون على FAR

`outer_header_creation` تلك. مضبوطة FAR هي N9 forward FAR التي تشير إلى PGW.

3. **كلا** FARs PGW UL و FARs UL التي تواجه gNB لـ SGW يتم تجاوزها إلى `BUFF` في خريطة eBPF. هذا ضروري لأن برنامج eBPF يرى الحزمة كما تصل من eNB مع TEID SGW، وليس TEID PGW.
4. **فقط** يتم تسجيل **PDRs UL التي تواجه gNB لـ SGW** في RedirectIndex (وليس PDRs UL لـ PGW). TEID PDR UL لـ SGW هو ما سيراه eBPF في الحزمة الواردة من eNB.
5. **مسار** بالنسبة لـ **DL** (إرسال الاستجابات مرة أخرى إلى UE)، يستخدم Dispatcher **DL FAR لجلسة SGW** (FAR الذي يوجه نحو eNB، مع `remoteip != n3_address`). تحتوي هذه FAR على TEID eNB وعنوان IP الحالي لـ gNB. يتم البحث عن DL TEID مباشرة من حالة الجلسة في وقت الاستجابة، وليس مخزنة في وقت التفعيل — هذا يتعامل مع الحالة التي يتم فيها تعيين TEID eNB بواسطة تعديل الجلسة اللاحق بعد تفعيل إعادة التوجيه.

مباشرة UPF في طوبولوجيا (لا SGW، SMF يتحدث مباشرة إلى UPF)، يتم استخدام FARs أي شيء وتُسجل SGW الخاصة بالجلسة فقط — لا يجد كود اكتشاف TEIDs UL و PDRs UL للجلسة الرئيسية مباشرة.

تدفق إشارات PFCP



عناصر PFCP المعنية

يحفز SMF إعادة التوجيه عن طريق تضمين `redirect_information` في `forwarding_parameters` لـ FAR:

IE	الوصف
`apply_action`	يتم تعيينه إلى FORW بواسطة SMF (يتجاوز إلى UPF داخليًا BUFF)
`redirect_information`	يحتوي على نوع إعادة التوجيه والعنوان
`forwarding_parameters`	يحتوي على `redirect_information` و `outer_header_creation`

أنواع معلومات إعادة التوجيه (3 وفقًا لـ GPP TS 29.244 1-8.2.20 الجدول):

النوع	القيمة	سلوك UPF
IPv4	0	للبوابة IP المقدمة كعنوان IPv4 استخدم سلسلة عنوان
IPv6	1	للبوابة IP المقدمة كعنوان IPv6 استخدم سلسلة عنوان
URL	2	للبوابة. لا IP ؛ استخدم ذلك كعنوانIP إلى URL حل اسم المضيف للرؤية فقط. يتحمل API يتم تخزينه في -- URL مسار UPF تستخدم خادم الويب الخاص بالبوابة مسؤولية التعامل مع أي توجيه يعتمد على المسار.
SIP URI	3	غير مدعوم حاليًا

كشف البوابة الأسرية

تقوم الحديقة المسورة بتحفيز **كشف تلقائي للبوابة الأسرية** على جميع منصات الأجهزة عندما يتصل جهاز ويحاول التحقق من الاتصال .DNS الرئيسية عن طريق تزوير استجابات

المزورة بإعادة توجيه هذه DNS بالإنترنت، يقوم بالاستعلام عن نطاقات معروفة. تقوم استجابة الفحوصات إلى عنوان IP أسرية بوابة أنه على الجهاز يفسره الذي ،للبوابة.

نطاقات كشف المنصة

المنصة	نطاق الكشف	ة المتوقعة
Apple (iOS/macOS)	`captive.apple.com`	HTTP 200 مع `<HTML><HEAD>&lt` `</HEAD><BODY>نجاح&lt/`
Android	`connectivitycheck.gstatic.com`	HTTP 204
Windows	`www.msftconnecttest.com`	HTTP 200 مع `Microsoft`
Samsung	`connectivitycheck.samsung.com`	HTTP 200

للبوابة بدلاً من عناوينها الحقيقية، يكشف الجهاز عن بوابة IP عندما تحل هذه النطاقات إلى عنوان أسرية ويعرض صفحة البوابة للمستخدم إما كـ:

- إشعار/منبثق (iOS، Android)
- إعادة توجيه تلقائية في المتصفح (Windows)

للبوابة المكونة استجابات مناسبة لهذه عناوين الكشف ثم IP يجب أن يقدم خادم البوابة عند عنوان يعيد التوجيه إلى صفحة إعادة الشحن/الدفع.

التكوين

يعيش تكوين الحديقة المسورة في ملف التكوين أثناء التشغيل. في تثبيت الإنتاج (`.deb` package)، يكون ملف التكوين في `/etc/omniupf/runtime.exs`. يتحقق نص بدء التشغيل من هذا الملف وإذا كان موجودًا، يقوم بتعيين (`rel/env.sh`) للإصدار `RELEASE_CONFIG_DIR=/etc/omniupf` بحيث يستخدم إصدار Erlang هذا بدلاً من `config/runtime.exs` المدمج. يجب إعادة تشغيل UPF بعد تغييرات التكوين.

```
#
================================================================
# حديقة مسورة / إعادة توجيه خارج الرصيد
#
================================================================

# تمكين تطبيق إعادة توجيه الحديقة المسورة
walled_garden_enabled = true

# عنوان IP لخادم البوابة (بوابة أسرية / صفحة إعادة الشحن)
walled_garden_portal_ip = "10.179.2.1"

# محلل DNS العلوي لاستعلامات النطاقات المدرجة في القائمة البيضاء
walled_garden_dns_resolver = "8.8.8.8"

# النطاقات المدرجة في القائمة البيضاء (يمكن للمشتركين الوصول إليها أثناء
إعادة التوجيه)
# يدعم الأحرف البديلة: "*.stripe.com" يتطابق مع "api.stripe.com"
walled_garden_whitelist = [
  "stripe.com",
  "*.stripe.com",
  "js.stripe.com",
  "hcaptcha.com",
  "*.hcaptcha.com",
  "newassets.hcaptcha.com",
]
```

المعلمات

المعلمة	النوع	مطلوب	الافتراضي	
`walled_garden_enabled`	Boolean	لا	`false`	ي لوظيفة ورة. عندما يكون `fal` `redirec` في FARs.
`walled_garden_portal_ip`	String (IPv4)	نعم (إذا تم تمكينه)	`10.179.2.1`	IP / عنوان ن. هذا هو ة IP عنوان ما لا يمكن المضيف لـ `redirec` في SMF، التوجيه هو أنواع .URI IPv4/IPv6 لعنوان من لنسبة لنوع في وقت إنشاء FAR
`walled_garden_dns_resolver`	String (IPv4)	لا	`8.8.8.8`	DNS محلل عند توجيه للنطاقات ة البيضاء. استعلامات مدرجة في ط إلى هذا زوير جميع خرى محليًا
`walled_garden_whitelist`	قائمة من السلاسل	لا	`[]`	التي يمكن وصول إليها

	الافتراضي	مطلوب	النوع	المعلمة
ي الحديقة ارة القائمة وب النمط				

إدارة القائمة البيضاء

التي يمكن للمشتركين (المحلولة الخاصة بها IP وعناوين) تتحكم القائمة البيضاء في النطاقات فقط SMF يقوم -- SMF وليس على ،UPF الوصول إليها أثناء إعادة التوجيه. يتم تكوين ذلك على بتحفيز إعادة التوجيه.

بناء الجملة النمطية

النمط	يتطابق	لا يتطابق
`stripe.com`	`stripe.com`, `api.stripe.com`, `js.stripe.com`	`evilstripe.com`, `notstripe.com`
`*.stripe.com`	`api.stripe.com`, `js.stripe.com`, `dashboard.stripe.com`	`stripe.com` (بالضبط)، `evilstripe.com`
`hcaptcha.com`	`hcaptcha.com`, `newassets.hcaptcha.com`	`evihcaptcha.com`

يتطابق مع النطاق نفسه وأي نطاق `stripe.com` :تستخدم الأنماط **تثبيت النطاق الفرعي** فرعي (`foo.stripe.com`)، ولكن ليس النطاقات التي تحتوي فقط على السلسلة (`evilstripe.com`). المطابقة غير حساسة لحالة الأحرف.

تخزين IP

بإعادة توجيه استعلام لنطاق مدرج في القائمة البيضاء، يتم **تخزين** DNS عندما يقوم وكيل **عناوين IP المحلولة تلقائيًا** في القائمة البيضاء. هذا يعني:

1. يقوم المشترك بالاستعلام عن `api.stripe.com`
2. يقوم وكيل DNS بإعادة توجيه إلى المحلل الحقيقي، ويعود بـ `104.18.7.25`
3. يتم إضافة IP المدرجة في القائمة `104.18.7.25` إلى ذاكرة التخزين المؤقت لعناوين البيضاء
4. يتم توجيه حركة HTTP/HTTPS (لا يتم إسقاطها) اللاحقة إلى `104.18.7.25`

يتم دائمًا إدراج عنوان IP للبوابة في القائمة البيضاء بغض النظر عن التكوين.

النطاقات الموصى بها للقائمة البيضاء

لصفحة إعادة شحن نموذجية مع معالجة الدفع عبر Stripe و hCaptcha:

```
walled_garden_whitelist = [
  # معالج الدفع
  "stripe.com",
  "*.stripe.com",

  # خدمة CAPTCHA
  "hcaptcha.com",
  "*.hcaptcha.com",

  # خطوط Google (إذا كانت البوابة تستخدمها)
  "fonts.googleapis.com",
  "fonts.gstatic.com",

  # CDN (إذا كانت مستضافة خارجيًا) لموارد البوابة
  "cdn.example.com",
]
```

روابط إعادة التوجيه لكل جلسة

يمكن أن تحتوي كل جلسة PFCP في ذلك عبر SMF على **هدف إعادة توجيه مختلف**. يتحكم IE في FAR:

- **نوع IPv4**: يتم تحليل سلسلة IP المقدمة واستخدامها كعنوان IP للبوابة لتلك الجلسة
- **نوع IPv6**: يتم تحليل سلسلة IPv6 المقدمة واستخدامها كعنوان IP للبوابة لتلك الجلسة

- **نوع URL**: يتم استخراج اسم المضيف ويتم حله عبر DNS في وقت إنشاء FAR. يتم استخدام عنوان IP المحلول كعنوان IP للبوابة. لا تستخدم URL -- يتم تخزينه للرؤية في API فقط.

هذا يمكّن السيناريوهات التي تقوم فيها مستويات مختلفة من المشتركين، MVNOs، أو خطط الخدمة بإعادة التوجيه إلى بوابات مختلفة:

الجلسة	SMF redirect_information	عنوان IP المستخدم	ه
الجلسة A	`IPv4: 10.179.2.1`	`10.179.2.1`	`10.179.2.`
الجلسة B	`IPv6: 2001:db8::1`	`2001:db8::1`	`2001:db8:`
الجلسة C	`URL: https://topup.mvno.com/recharge`	عنوان IP المحلول لـ `topup.mvno.com`	`https://t`

يخزن UPF عنوان IP للبوابة ورابط إعادة التوجيه لكل جلسة، والتي يمكن رؤيتها عبر API.

API

مسار القاعدة لجميع نقاط نهاية الحديقة المسورة هو `/v1/walled_garden` (لا يوجد بادئة `/api`). يتم تقديم هذه النقاط النهائية بواسطة خادم HTTP Phoenix على `api_port` المكون (الافتراضي: 8080).

GET /v1/walled_garden

يعيد جميع جلسات إعادة التوجيه النشطة في الحديقة المسورة، وعناوين IP المدرجة في القائمة البيضاء، ونطاقات CIDR المخزنة، وتفاصيل الجلسة.

الاستجابة:

```
{
  "redirect_count": 2,
  "redirects": [
    {
      "teid": "0x4000",
      "session_seid": 1,
      "portal_ip": "10.179.2.1",
      "redirect_url": null,
      "ue_ip": "10.60.0.1",
      "gnb_ip": "10.179.1.21",
      "dl_teid": "0x5000",
      "far_global_id": 42
    },
    {
      "teid": "0x4001",
      "session_seid": 2,
      "portal_ip": "10.179.2.2",
      "redirect_url": "https://topup.mvno.com",
      "ue_ip": "10.60.0.2",
      "gnb_ip": "10.179.1.21",
      "dl_teid": "0x5001",
      "far_global_id": 43
    }
  ],
  "whitelisted_ips": [
    {"ip": "10.179.2.1", "type": "portal"},
    {"ip": "104.18.7.25", "type": "resolved"},
    {"ip": "104.18.6.25", "type": "resolved"}
  ],
  "whitelisted_cidrs": ["192.168.0.0/24"]
}
```


حقول الاستجابة:

الحقل	الوصف
`redirect_count`	عدد جلسات الحديقة المسورة النشطة
`redirects[].teid`	الرفع الذي يتم اعتراضه TEID (hex)
`redirects[].session_seid`	جلسة PFCP SEID
`redirects[].portal_ip`	عنوان IP للبوابة لهذه الجلسة المحددة
`redirects[].redirect_url`	رابط إعادة التوجيه من SMF (إذا كان من نوع URL)، أو null
`redirects[].ue_ip`	عنوان IP لـ UE
`redirects[].gnb_ip`	عنوان IP لـ gNB لاستجابات GTP-U
`redirects[].dl_teid`	TEID الهابط لتغليف GTP-U
`redirects[].far_global_id`	معرف FAR العالمي الداخلي
`whitelisted_ips`	جميع عناوين IP الحالية في القائمة البيضاء (+ البوابة حلول DNS المخزنة)
`whitelisted_cidrs`	نطاقات CIDR المضافة عبر API القائمة البيضاء

POST /v1/walled_garden

تفعيل إعادة توجيه الحديقة المسورة على جلسة PFCP موجودة بواسطة SEID. هذا هو المسار المسار العادي هو عبر تعديل جلسة PFCP — الذي يتم تحفيزه بواسطة المشغل/API مفيدًا للاختبار وللمشغلين الذين يحتاجون API يعد مسار `redirect_information` في FAR. إلى إعادة توجيه جلسة يدويًا دون تدخل SMF.

جسم الطلب:

```
{
  "seid": 1,
  "url": "http://10.179.2.1/"
}
```

الحقل	النوع	مطلوب	الوصف
`seid`	عدد صحيح	نعم	SEID المحلي لجلسة PFCP لإعادة التوجيه
`url`	سلسلة	نعم	عنوان — هدف إعادة التوجيه IPv4/IPv6 أو URL. إذا تم إعطاء URL، يتم حل اسم المضيف ويستخدم عنوان IP الناتج كعنوان URL للبواب��. يتم تخزين سلسلة URL الكاملة للرؤية في API.

الاستجابة (200 OK):

```
{
  "status": "redirect activated",
  "info": {
    "seid": 1,
    "sgw_seid": 7,
    "portal_ip": "10.179.2.1",
    "ue_ip": "10.60.0.1",
    "gnb_ip": "10.179.1.21",
    "ul_teids": ["0x4000", "0x4006"]
  }
}
```

حقل `sgw_seid` ليس فارغًا عندما يتم اكتشاف حلقة N9 (جلسة مزدوجة SGW+PGW) وتم أيضًا تعيين TEID الرفع إلى BUFF. تسرد `ul_teids` جميع TEIDs الرفع التي تم تسجيلها في فهرس إعادة التوجيه.

استجابات الخطأ:

الحالة	المعنى
404	لم يتم العثور على الجلسة (SEID لا يتطابق مع أي جلسة PFCP نشطة)
400	معلمات مطلوبة مفقودة

DELETE /v1/walled_garden/:seid

إلغاء تفعيل إعادة توجيه الحديقة المسورة لجلسة، واستعادة التوجيه العادي. يتم تعيين جميع FARs مرة أخرى إلى (N9 المزدوجة في طوبولوجيا حلقة SGW لجلسة FARs بما في ذلك) الرفع للجلسة `action=FORW` في خريطة eBPF، ويتم إلغاء تسجيل TEIDs من فهرس إعادة التوجيه.

معامل المسار: `:seid` — SEID المحلي للجلسة التي سيتم إلغاء تفعيلها.

الاستجابة (200 OK):

```
{"status": "redirect removed", "info": {"seid": 1}}
```

استجابات الخطأ:

الحالة	المعنى
404	لم يتم العثور على الجلسة

GET /v1/walled_garden/whitelist

يعيد القائمة البيضاء الحالية: عناوين IP الفردية المخزنة (عنوان IP للبوابة وعناوين IP المحلولة عبر DNS) وأي نطاقات CIDR تمت إضافتها عبر API.

الاستجابة:

```
{
  "ips": [
    {"ip": "10.179.2.1", "type": "portal"},
    {"ip": "104.18.7.25", "type": "resolved"}
  ],
  "cidrs": ["192.168.100.0/24"]
}
```

POST /v1/walled_garden/whitelist

إضافة عنوان IP أو نطاق CIDR إلى القائمة البيضاء أثناء التشغيل، دون إعادة تشغيل UPF. التغييرات تكون في الذاكرة فقط ولا تستمر عبر إعادة التشغيل — أضف إدخالات دائمة إلى `walled_garden_whitelist` في `runtime.exs`.

جسم الطلب (إضافة IP):

```
{"ip": "203.0.113.10"}
```

جسم الطلب (إضافة نطاق CIDR):

```
{"cidr": "192.168.100.0/24"}
```

يجب أن يكون واحد فقط من `ip` أو `cidr` موجودًا. يعني إضافة نطاق CIDR أن أي حركة مرور ضمن هذا النطاق يتم توجيهها بواسطة الموزع دون الحاجة إلى إدخال ذاكرة تخزين IP إلى عناوين IP. لكل DNS مؤقت

الاستجابة (200 OK — تم إضافة IP):

```
{"status": "added", "ip": "203.0.113.10"}
```

الاستجابة (200 OK — تم إضافة CIDR):

```
{"status": "added", "cidr": "192.168.100.0/24"}
```

استجابات الخطأ:

الحالة	المعنى
400	غير صالح، أو حقول الجسم مفقودة CIDR ،غير صالح IP

مقاييس بروميثيوس

مقاييس قياس

المقياس: `upf_walled_garden_active_redirects` **النوع**: Gauge **الوصف**: عدد الجلسات الحالية في حالة إعادة توجيه الحديقة المسورة

استعلامات المثال:

```
# عدد إعادة التوجيه الحالي
upf_walled_garden_active_redirects
```

العدادات

المقياس: `upf_walled_garden_packets_intercepted_total` **النوع**: Counter **الوصف**: إجمالي الحزم المعترضة بواسطة الحديقة المسورة (كل حركة المرور الرفع من الجلسات المعاد توجيهها)

المقياس: `upf_walled_garden_packets_dropped_total` **النوع**: Counter **الوصف**: إجمالي الحزم التي تم إسقاطها بواسطة الحديقة المسورة (حركة المرور غير المدرجة في القائمة البيضاء، حركة المرور غير DNS)

المقياس: `upf_walled_garden_packets_forwarded_total` **النوع**: Counter **التسميات**:

- `dst_ip` - عنوان IP الوجهة للحزمة المعاد توجيهها **الوصف**: الحزم التي تم توجيهها عبر الحديقة المسورة إلى الوجهات المدرجة في القائمة البيضاء. مُعلمة حسب عنوان IP الوجهة لرؤية لكل وجهة.

المقياس: `upf_walled_garden_bytes_forwarded_total` **النوع**: Counter **التسميات**:

- `dst_ip` - عنوان IP للحركة المعاد توجيهها **الوصف**: بايتات تم توجيهها عبر الحديقة المسورة لكل عنوان IP في ا��قائمة وجهة. استخدم هذا لتحديد الخدمات المدرجة البيضاء التي يصل إليها المشتركون أثناء إعادة التوجيه.

المقياس: `upf_walled_garden_dns_spoofed_total` **النوع**: Counter **التسميات**:

- `domain` - استعلامات DNS التي تم تزويرها بواسطة النطاق الذي تم تزويره **الوصف**: الحديقة المسورة. مُعلمة حسب النطاق المستعلم عنه.

المقياس: `upf_walled_garden_dns_forwarded_total` **النوع**: Counter **التسميات**:

- `domain` - استعلامات DNS التي تم توجيهها إلى المحلل الحقيقي (النطاقات المدرجة في القائمة البيضاء) **الوصف**: النطاق المدرج في القائمة البيضاء الذي تم توجيهه. مُعلمة حسب النطاق.

استعلامات المثال

```
# جلسات الحديقة المسورة النشطة
upf_walled_garden_active_redirects

# (حزم/ثانية) معدل الاعتراض
rate(upf_walled_garden_packets_intercepted_total[5m])

# (يجب أن يكون غالبية المعترضين) معدل الإسقاط
rate(upf_walled_garden_packets_dropped_total[5m])

# (بايت/ثانية) حركة المرور المعاد توجيهها حسب الوجهة
sum by (dst_ip)
(rate(upf_walled_garden_bytes_forwarded_total[5m]))

# أعلى 5 وجهات مدرجة في القائمة البيضاء حسب حجم الحركة
topk(5, sum by (dst_ip)
(rate(upf_walled_garden_bytes_forwarded_total[5m])))

# أكثر النطاقات المزورة استعلامًا
topk(10, sum by (domain)
(rate(upf_walled_garden_dns_spoofed_total[5m])))

# معدل البحث عن النطاق المدرج في القائمة البيضاء
sum by (domain) (rate(upf_walled_garden_dns_forwarded_total[5m]))

# نسبة الحزم المفقودة مقابل المعاد توجيهها
sum(rate(upf_walled_garden_packets_dropped_total[5m]))
  / sum(rate(upf_walled_garden_packets_intercepted_total[5m]))
```

استكشاف الأخطاء وإصلاحها

عدم ظهور بوابة الأسرية على الجهاز

الأعراض: يتم إعادة توجيه المشترك (مرئي في API) ولكن الجهاز لا يظهر موجه بوابة أسرية.

الأسباب المحتملة:

- المكون IP خادم البوابة لا يستجيب على عنوان
- خادم البوابة لا يتعامل مع عناوين الكشف الخاصة بالمنصة
- استجابة DNS لا تصل إلى UE (تحقق من مسار GTP-U)

الحل:

1. تحقق من أن خادم البوابة يمكن الوصول إليه: `curl http://<portal_ip>/`
2. تأكد من أن البوابة تتعامل مع عناوين الكشف (على سبيل المثال، `GET /hotspot-detect.html` لـ Apple)
3. تحقق من `GET /v1/walled_garden` لتأكيد تسجيل الجلسة
4. تحقق من مقاييس بروميثيوس: يجب أن تزداد `upf_walled_garden_dns_spoofed_total`
5. تحقق من أن مسار GTP-U الهابط يعمل (يجب أن تبقى DL FAR في FORW)

عدم تحميل صفحة الدفع

الأعراض: يرى المشترك بوابة أسرية ولكنه لا يمكنه الوصول إلى صفحة الدفع (Stripe، إلخ).

الأسباب المحتملة:

- نطاق معالج الدفع ليس في القائمة البيضاء
- لم يتم تخزينه IP CDN يستخدم معالج الدفع عنوان
- نمط القائمة البيضاء لا يتطابق مع النطاقات الفرعية بشكل صحيح

الحل:

1. تحقق من `GET /v1/walled_garden` — تحقق من أن `whitelisted_ips` تشمل عناوين IP لمعالجات الدفع
2. تحقق من بروميثيوس: يجب أن تظهر `upf_walled_garden_dns_forwarded_total{domain="stripe.com"}` عمليات البحث
3. أضف النطاقات المفقودة إلى القائمة البيضاء (شائعة: `js.stripe.com`, `m.stripe.network`)
4. تحقق من `upf_walled_garden_bytes_forwarded_total` حسب `dst_ip` لرؤية ما هي الحركة التي تتدفق فعليًا

عدم تفعيل إعادة التوجيه

الأعراض: يرسل SMF تعديل الجلسة مع `redirect_information` ولكن حركة مرور المشترك لا يتم اعتراضها.

الأسباب المحتملة:

- `walled_garden_enabled` هو `false`
- IE `redirect_information` في FAR غير صحيح (في FAR **الرفع** يجب أن يكون)
- لم يتم تجاوز إجراء FAR إلى BUFF

الحل:

1. تحقق من التكوين: `walled_garden_enabled = true`
2. تحقق من `GET /v1/pfcp_sessions` — انظر إلى إجراء FAR للجلسة. يجب أن يظهر FAR الرفع الإجراء `0`x04 (BUFF)
3. تحقق من `GET /v1/walled_garden` — يجب أن يظهر TEID للجلسة في قائمة إعادة التوجيه
4. تحقق من سجلات UPF للرسائل `redirect_info` أثناء تعديل الجلسة

عدم مسح إعادة التوجيه بعد إعادة الشحن

الأعراض: قام المشترك بإعادة شحن رصيده ولكن الحركة لا تزال معترضة.

الأسباب المحتملة:

- لم يرسل SMF تعديل الجلسة لإزالة إعادة التوجيه
- لم يتم اكتشاف الانتقال من BUFF إلى FORW

الحل:

1. تحقق من `GET /v1/walled_garden` — هل لا تزال الجلسة مدرجة؟
2. تحقق من `GET /v1/pfcp_sessions` — تحقق من أن إجراء FAR قد تم تحديثه مرة أخرى إلى `0`x02 (FORW)
3. تحقق من سجلات SMF لتأكيد أنه أرسل تعديل الجلسة
4. تحقق من سجلات UPF للرسالة "تمت إزالة إعادة التوجيه، إلغاء تسجيل الحديقة المسورة"

الهابط غير صحيح (قديم بعد تعديل الجلسة) TEID

الأعراض: استجابات DNS ؛ يتم إرسال حزمUE المزورة أو الحركة المعاد توجيهها لا تصل إلى غير معروف TEID يسقطها كـ eNB ولكن gNB الصحيح لـ IP إلى عنوان GTP-U.

السبب: في طوبولوجيات 4G/SGW+PGW، قد يكون TEID eNB في FAR DL لـ SGW صفرًا أو على سبيل المثال،) عنصر نائب في الوقت الذي يتم فيه تفعيل إعادة توجيه الحديقة المسورة TEID يحدث تبادل `Initial Context Setup` عندما يقوم eNB بتخصيص TEID (بعد إنشاء الجلسة. ولكن — SGW في جلسة FAR DL الخاص به ويرسله مرة أخرى عبر تعديل الجلسة، يتم تحديث غير الصحيح TEID إذا كان الموزع قد خزّن القيمة القديمة، فسوف يستخدم.

الحل: يقوم الموزع بحل FAR DL الحالي **حيًا** في وقت الاستجابة من خلال البحث في TEID DL لجلسة SGW (FAR مع `outer_header_creation` مضبوطة و `remoteip != n3_address`). إنه يتراجع فقط إلى القيمة المخزنة في وقت التفعيل إذا فشلت عملية البحث عن الجلسة. لذلك، يجب أن يصحح DNS القديم نفسه بمجرد إرسال الاستجابة التالية TEID أو المعاد توجيهها. إذا كنت لا تزال ترى TEID غير صحيح:

1. تحقق من `GET /v1/walled_garden` — تحقق من أن `dl_teid` في إدخال إعادة التوجيه يبدو معقولًا (غير صفر).
2. تحقق من `GET /v1/pfcp_sessions` — انظر إلى إدخالات FAR لجلسة SGW؛ يجب أن تحتوي FAR التي تواجه gNB على `teid` و `remoteip` الحاليين.
3. إذا تم حذف جلسة SGW (على سبيل المثال، تسليم أو تحرير)، ستبقى إدخال إعادة التوجيه في RedirectIndex ولكن البحث الحي سيفشل. في هذه الحالة، استخدم `DELETE /v1/walled_garden/:seid` لـ SMF لتنظيف الإدخال القديم والسماح بإعادة الإنشاء.

تعيد البوابة 304 غير معدلة

الأعراض: يظهر متصفح المشترك صفحة فارغة أو يتم تحميل صفحة البوابة الأسرية مرة واحدة ولكن الزيارات اللاحقة تظهر فارغة.

السبب: يتم إرسال استجابات HTTP 304 (غير معدلة) عندما يكون لدى المتصفح إصدار مخزن من الصفحة ويؤكد الخادم أنه لم يتغير شيء. بالنسبة لتدفقات البوابة الأسرية، ترسل بعض خوادم الويب الخاصة بالبوابة 304 استجابةً لطلبات الكشف عن المنصة (`/hotspot-detect.html`, `/generate_204`, إلخ) إذا أرسل المتصفح رؤوس `If-Modified-Since` أو `If-None-Match`. ترسل بعض تطبيقات البوابة الأسرية أيضًا 304 لصفحة إعادة التوجيه نفسها.

الحل:

1. يقوم UPF بإعادة توجيه استجابات HTTP بشكل شفاف من خادم البوابة — لا يعدل رموز الاستجابة. المشكلة في تكوين خادم الويب الخاص بالبوابة.
2. على خادم البوابة، تأكد من أن نقاط نهاية الكشف عن المنصة تعيد الحالة الصحيحة (200، وليس 304) مع محتوى الجسم المتوقع (انظر كشف البوابة الأسرية).
3. قم بتكوين خادم البوابة لإرسال `Cache-Control: no-store` وترك رؤوس `ETag`/`Last-Modified` على نقاط نهاية الكشف حتى لا يقوم المتصفحات بتخزينها.
4. تحقق باستخدام: `curl -v -H "If-None-Match: foo" http://<portal_ip>/hotspot-detect.html` — يجب أن تكون الاستجابة 200، وليس 304.

توجيه مقبس خام

مدرج في القائمة البيضاء، فإنه يستخدم IP عندما يقوم الموزع بإعادة توجيه حزمة إلى عنوان **مقبس خام** (`IPPROTO_RAW`، 255 رقم البروتوكول) عبر وحدة `:socket` في Erlang. يتم فتح مقبس خام جديد لكل حزمة، ويتم إرسال الحزمة الداخلية الكاملة (كما تم استلامها من eBPF) باستخدام `sendto`، ويتم إغلاق المقبس على الفور.

كيف يعمل هذا: تحتوي الحزمة الداخلية من حمولة GTP-U بالفعل على رأس IP صالح مع عنوان كمصدر وعنوان خادم الوجهة كوجهة. من خلال حقن هذه الحزمة عبر مقبس خام لـ IP UE، يقوم النواة بتوجيهها بناءً على عنوان الوجهة باستخدام جدول توجيه `IPPROTO_RAW` باستخدام على مسار إلى خادم البوابة/الوجهة المدرجة UPF الخاصة بـ N6 المضيف. يجب أن تحتوي واجهة في القائمة البيضاء لكي يعمل هذا.

المشاكل الشائعة:

العرض	السبب المحتمل	الحل
فشل التوجيه بصمت	`EPERM` — يتطلب المقبس الخام الجذر أو `CAP_NET_RAW`	تأكد من أن OmniUPF يعمل كجذر أو لديه القدرة `CAP_NET_RAW`
تم توجيه الحزم ولكن لا تصل استجابة إلى UE	مسار إلى البوابة N6 مفقود	أضف مسارًا إلى الشبكة الفرعية للبوابة على مضيف UPF
`Walled garden: raw socket open failed` في السجلات	قدرة مفقودة أو قيود نواة	تحقق من خدمة `systemd` `AmbientCapabilities=CAP_NET_RAW`
يعمل التوجيه ولكن يحصل UE على IP خاطئ كمصدر	يعيد N6 على NAT كتابة المصدر	تأكد من أن استجابات البوابة إلى IP UE يتم توجيهها مرة أخرى عبر UPF

تحقق من سجلات UPF للحصول على رسائل `Walled garden forward failed` و `raw socket open failed`. استخدم مقاييس بروميثيوس `upf_walled_garden_packets_forwarded_total` و `upf_walled_garden_bytes_forwarded_total` لتأكيد تدفق الحركة.

تحذير من الكاردينالية العالية لبروميثيوس

ملاحظة: يمكن أن تؤدي التسميات على مقاييس الحديقة المسورة إلى إنتاج `domain` و `dst_ip` كاردينالية عالية إذا تم استعلام العديد من الوجهات أو النطاقات الفريدة. في النشر الكبير، ضع في اعتبارك استخدام قواعد التسجيل لتجميع هذه المقاييس:

```
# قاعدة تسجيل لتجميع حسب الشبكة الفرعية /24 بدلاً من IP الفردي
sum by (dst_subnet) (
  label_replace(
    rate(upf_walled_garden_bytes_forwarded_total[5m]),
    "dst_subnet", "$1.0/24", "dst_ip", "(\d+\.\d+\.\d+)\.\d+"
  )
)
```

دليل عمليات واجهة المستخدم على الويب

جدول المحتويات

1. نظرة عامة
2. الوصول إلى لوحة التحكم
3. عرض الجلسات
4. إدارة القواعد
5. إدارة المخازن
6. لوحة إحصائيات
7. مراقبة السعة
8. عرض التكوين
9. عرض المسارات
10. عرض قدرات XDP
11. عارض السجلات

نظرة عامة

توفر واجهة المستخدم على الويب الخاصة بـ OmniUPF لوحة تحكم شاملة للمراقبة والإدارة في الوقت الحقيقي لوظيفة مستوى المستخدم. تم بناء الواجهة على Phoenix LiveView وتوفر:

- **رؤية في الوقت الحقيقي** لجلسات PFCP و اتصالات PDU النشطة
- **فحص القواعد** لـ PDR و FAR و QER و URR عبر جميع الجلسات
- **إدارة المخازن** لتخزين الحزم أثناء أحداث التنقل
- **مراقبة الإحصائيات** لمعالجة الحزم والمسارات والواجهات
- **تتبع السعة** لاستخدام خرائط eBPF والحدود
- **عرض السجلات الحية** لاستكشاف الأخطاء وإصلاحها

الهيكلية

تتواصل لوحة التحكم مع عدة مثيلات من OmniUPF عبر واجهة برمجة التطبيقات REST الخاصة بها لـ:

- استعلام جلسات PFCP والارتباطات
- فحص قواعد اكتشاف الحزم وإعادة التوجيه
- مراقبة مخازن الحزم وحالتها
- الوصول إلى الإحصائيات في الوقت الحقيقي ومقاييس الأداء
- تتبع سعة خرائط eBPF واستخدامها

الوصول إلى لوحة التحكم

الوصول الافتراضي

تكون لوحة التحكم متاحة عبر HTTPS على خادم إدارة OmniUPF:

```
https://<upf-server>:443/
```

المنفذ الافتراضي: 443 (HTTPS مع شهادة موقعة ذاتيًا)

التكوين

تتطلب لوحة التحكم تكوين مضيف OmniUPF في `config/config.exs`:

يمكن تكوين عدة مثيلات من UPF لنشر متعدد المثيلات:

يحدد تكوين `upf_hosts` أي مثيلات من OmniUPF متاحة في قائمة تحديد المضيف في جميع أنحاء واجهة المستخدم.

التنقل

توفر لوحة التحكم علامات تبويب للتنقل لكل منطقة تشغيلية:

- **الجلسات** - `/sessions` - جلسات PFCP والارتباطات
- **القواعد** - `/rules` - فحص قواعد URR و QER و FAR و PDR

- **المخازن** - `/buffers` - مراقبة الحزم والتحكم
- **الإحصائيات** - `/statistics` - إحصائيات الحزم والمسارات و XDP والواجهات
- **السعة** - `/capacity` - استخدام خرائط eBPF ومراقبة السعة
- **التكوين** - `/upf_config` - تكوين UPF وعناوين مستوى البيانات
- **المسارات** - `/routes` - مسارات UE وجلسات بروتوكول التوجيه (BGP و OSPF)
- **قدرات XDP** - `/xdp_capabilities` - دعم وضع XDP وقدرات الأداء
- **السجلات** - `/logs` - بث السجلات الحية

عرض الجلسات

الرابط: `/sessions`

الميزات

يعرض عرض الجلسات جميع جلسات PFCP النشطة والارتباطات من مثيلات OmniUPF المحددة.

ملخص ارتباطات PFCP

يعرض جميع ارتباطات PFCP النشطة (من اتصالات التحكم SMF/PGW-C):

العمود	الوصف
معرف العقدة	معرف عقدة SMF أو PGW-C (FQDN أو IP)
العنوان	عنوان IP لـ SMF/PGW-C للتواصل عبر PFCP
معرف الجلسة التالية	معرف جلسة PFCP المتاحة التالية لهذا الارتباط

الغرض:

- التحقق من اتصال SMF بـ UPF
- مراقبة عدد اتصالات مستوى التحكم
- تتبع تخصيص معرف الجلسة لكل ارتباط

جدول الجلسات النشطة

يعرض جميع جلسات PFCP التي تمثل جلسات PDU النشطة لـ UE:

العمود	الوصف
SEID المحلي	معرف نقطة نهاية الجلسة المعينة من UPF
SEID البعيد	معرف نقطة نهاية الجلسة المعينة من SMF
عنوان UE	عنوان IPv4 أو IPv6 لمعدات المستخدم
TEID	معرف نقطة نهاية نفق GTP-U لحركة المرور الصاعدة
PDRs	عدد قواعد اكتشاف الحزم في الجلسة
FARs	عدد قواعد إجراء إعادة التوجيه في الجلسة
QERs	عدد قواعد تنفيذ QoS في الجلسة
URRs	عدد قواعد تقارير الاستخدام في الجلسة
الإجراءات	زر التوسيع لعرض معلومات القاعدة التفصيلية

الميزات:

- **تصفية بواسطة IP**: العثور على الجلسات لعنوان IP محدد لـ UE
- **تصفية بواسطة TEID**: العثور على الجلسات بواسطة معرف نقطة النهاية للنفق
- **توسيع الجلسة**: عرض تفاصيل JSON الكامل بتنسيق PDR/FAR/QER
- **تحديث تلقائي**: يتم التحديث كل 10 ثوانٍ

عرض الجلسة الموسعة:

عند النقر على "توسيع" في جلسة، يظهر العرض:

- **قواعد اكتشاف الحزم (PDRs)**: JSON كامل مع TEID، عنوان UE، معرف FAR، معرف QER، مرشحات SDF
  - **قابلة للنقر** **معرفات PDR** - انقر للتنقل إلى علامة التبويب القواعد الكاملة وعرض تفاصيل PDR
  - ترتبط PDRs الصاعدة (TEID ≠ 0) بالبحث عن PDR الصاعدة

    - ترتبط PDRs النازلة (IPv4) بالبحث عن PDR النازلة
    - ترتبط PDRs النازلة (IPv6) بالبحث عن PDR النازلة IPv6
- **قواعد إجراء إعادة التوجيه (FARs)**: أعلام الإجراءات، إنشاء رأس خارجي، نقاط النهاية الوجهة
- **قواعد تنفيذ QoS (QERs)**: QFI ومعلمات QoS الأخرى و GBR و MBR
- **قواعد تقارير الاستخدام (URRs)**: عدادات الحجم (الصاعدة والنازلة وإجمالي البايتات)

عرض الجلسة الموسعة يظهر تفاصيل PDRs و FARs و QERs لجلسة معينة.

حالات الاستخدام

تحقق من اتصال UE:

1. انتقل إلى عرض الجلسات
2. أدخل عنوان IP لـ UE في الفلتر
3. تأكد من وجود الجلسة مع TEID الصحيح
4. قم بالتوسيع للتحقق من تكوين PDR/FAR

مراقبة عدد الجلسات:

- تحقق من إجمالي عدد الجلسات في الرأس
- قارن عبر عدة مثيلات UPF
- تتبع نمو الجلسة بمرور الوقت

استكشاف مشكلات الجلسة:

- ابحث عن عنوان IP أو TEID لـ UE محدد
- قم بتوسيع الجلسة لفحص تكوين القاعدة
- تحقق من معلمات إعادة توجيه FAR
- تحقق من إعدادات QoS لـ QER

التحديثات في الوقت الحقيقي

يقوم عرض الجلسات بتحديث تلقائي كل 10 ثوانٍ. يظهر مؤشر صحة حالة اتصال UPF:

- **صحي** (أخضر): UPF قابل للوصول ويستجيب
- **غير صحي** (أحمر): UPF غير قابل للوصول أو لا يستجيب
- **غير معروف** (رمادي): حالة الصحة لم تحدد بعد

إدارة القواعد

الرابط: `/rules`

يوفر عرض القواعد فحصًا شاملاً لجميع قواعد اكتشاف الحزم وإعادة التوجيه و QoS وتقارير الاستخدام عبر جميع الجلسات.

قواعد اكتشاف الحزم - PDR علامة تبويب

عرض وفحص جميع PDRs في UPF مع **نماذج البحث** و **التنقل القابل للنقر**:

PDRs الصاعدة (N3 → N6):

- **نموذج البحث**: البحث بواسطة TEID لعرض تفاصيل PDR الصاعدة المحددة
- **TEID**: معرف نقطة نهاية نفق GTP-U من gNB (قابل للنقر - ينتقل إلى البحث)

- **معرف FAR**: قاعدة إجراء إعادة التوجيه المرتبطة (قابلة للنقر - تنتقل إلى علامة تبويب FAR)
- **معرف QER**: قاعدة تنفي QoS المرتبطة (قابلة للنقر - تنتقل إلى علامة تبويب QER)
- **معرفات URR**: قواعد تقارير الاستخدام المرتبطة (قابلة للنقر - تنتقل إلى علامة تبويب URR)
- **إزالة الرأس الخارجي** GTP-U: علامة إزالة التفكيك
- **مرشحات SDF**: قواعد تصنيف تدفق البيانات الخدمية

PDRs النازلة (N6 → N3):

- **نموذج البحث**: البحث بواسطة عنوان IPv4 لـ UE لعرض تفاصيل PDR النازلة المحددة
- **عنوان UE**: عنوان IPv4 لمعدات المستخدم (معروض في نتائج البحث)
- **معرف FAR**: قاعدة إجراء إعادة التوجيه المرتبطة (قابلة للنقر - تنتقل إلى علامة تبويب FAR)
- **معرف QER**: قاعدة تنفيذ QoS المرتبطة (قابلة للنقر - تنتقل إلى علامة تبويب QER)
- **معرفات URR**: قواعد تقارير الاستخدام المرتبطة (قابلة للنقر - تنتقل إلى علامة تبويب URR)
- **وضع SDF**: وضع مرشح تدفق البيانات الخدمية (افتراضي + sdf، فقط sdf، لا شيء)
- **التصفح**: تصفح PDRs لكل 100 افت��اضي (مع عناصر التحكم في الصفحات صفحة، الحد الأقصى 1000)

PDRs النازلة IPv6:

- تدعم واجهة برمجة التطبيقات التصفح لـ PDRs النازلة IPv6
- نفس الهيكل مثل IPv4 ولكن مفاتيح بواسطة عناوين IPv6
- يمكن إضافة علامة تبويب واجهة المستخدم الكاملة إذا لزم الأمر

قواعد إجراء إعادة التوجيه - FAR علامة تبويب

عرض جميع FARs ومعلماتها مع إجراءاتها:

الميزات:

- **نموذج البحث**: البحث بواسطة معرف FAR لعرض تفاصيل FAR المحددة
- **البحث التلقائي**: النقر على معرفات FAR من تفاصيل PDR يملأ البحث تلقائيًا

- حالة التخزين الحالية FAR **التحديثات في الوقت الحقيقي**: تعكس حالة

العمود	الوصف
معرف FAR	معرف قاعدة إعادة التوجيه الفريد
الإجراء	أعلام إجراءات إعادة التوجيه (FORWARD و DROP و BUFFER و NOTIFY و DUPLICATE)
التخزين	حالة التخزين الحالية (مفعل/معطل)
الوجهة	معلمات إنشاء الرأس الخارجي (TEID، عنوان IP)

أعلام إجراء FAR:

- **FORWARD (1)**: إعادة توجيه الحزمة إلى الوجهة
- **DROP (2)**: تجاهل الحزمة
- **BUFFER (4)**: تخزين الحزمة في المخزن
- **NOTIFY (8)**: إرسال إشعار إلى مستوى التحكم
- **DUPLICATE (16)**: تكرار الحزمة إلى وجهات متعددة

تبديل التخزين:

- انقر على "تمكين التخزين" أو "تعطيل التخزين" لتبديل علامة التخزين
- مفيد لاستكشاف مشكلات الانتقال
- تغييرات إجراء FAR على الفور في خريطة eBPF

قواعد تنفيذ QoS - علامة تبويب QER

عرض قواعد QoS المطبقة على تدفقات الحركة:

الميزات:

- **التنقل القابل للنقر**: انقر على معرفات QER من تفاصيل PDR للتنقل وتسليط الضوء على QER محدد
- **التسليط التلقائي**: يتم تسليط الضوء على صف QER عند التنقل من PDR

- مع عناصر التحكم في الصفحات (افتراضي 100 لكل صفحة، QERs **التصفح**: تصفح الحد الأقصى 1000)

العمود	الوصف
معرف QER	معرف قاعدة QoS الفريد (PDRs قابل للنقر عند الإشارة إليه من)
MBR (الصاعدة)	الحد الأقصى لمعدل البت لحركة المرور الصاعدة (kbps)
MBR (النازلة)	الحد الأقصى لمعدل البت لحركة المرور النازلة (kbps)
GBR (الصاعدة)	معدل البت المضمون لحركة المرور الصاعدة (kbps)
GBR (النازلة)	معدل البت المضمون لحركة المرور النازلة (kbps)
QFI	معرف تدفق QoS (5 علامةG)

تفسير QoS:

- **MBR = 0**: لا يوجد حد للمعدل
- **GBR = 0**: أفضل جهد (لا يوجد عرض نطاق مضمون)
- **GBR > 0**: تدفق بمعدل بت مضمون (مفضل)

قواعد تقارير الاستخدام - URR علامة تبويب

عرض قواعد تتبع الاستخدام وعدد الحجم:

الميزات:

- محدد وتسليط الضوء URR للعثور على URR **نموذج البحث**: البحث بواسطة معرف عليه
- للتنقل وتسليط PDR من تفاصيل URR **التنقل القابل للنقر**: انقر على معرفات محدد URR الضوء على
- باللون الأزرق عند التنقل من URR **التسليط التلقائي**: يتم تسليط الضوء على صف أو البحث عبر البحث PDR
- مع عناصر التحكم في الصفحات (افتراضي 100 لكل صفحة، URRs **التصفح**: تصفح الحد الأقصى 1000)

العمود	الوصف
معرف URR	قابل للنقر عند الإشارة إليه من) معرف قاعدة تقارير الاستخدام الفريد PDRs)
حجم الصاعدة	إلى الشبكة البيانات UE بايتات أرسلت من
حجم النازلة	UE بايتات أرسلت من الشبكة البيانات إلى
إجمالي الحجم	إجمالي البايتات في كلا الاتجاهين
الإجراءات	URR زر الحذف لإعادة تعيين العدادات لهذه

عرض الحجم:

- يتم تنسيقه تلقائيًا (B و KB و MB و GB و TB)
- يتم تحديث العدادات في الوقت الحقيقي مع كل تحديث
- يستخدم للفوترة والتحليلات

التصفية:

- يظهر فقط URRs ذات الحجم غير الصفري
- يتم تصفية URRs غير النشطة (جميع العدادات عند 0) لأغراض الأداء

حالات الاستخدام

فحص تصنيف الحركة:

1. انتقل إلى القواعد ← علامة تبويب PDR
2. ابحث عن TEID أو عنوان IP لـ UE محدد
3. تحقق من أن PDR يرتبط بـ FAR و QER الصحيحين

استكشاف مشكلات إعادة التوجيه:

1. انتقل إلى القواعد ← علامة تبويب FAR
2. حدد معرف FAR من PDR للجلسة

3. تحقق من أن الإجراء هو FORWARD (ليس DROP أو BUFFER)
4. تحقق من معلمات إنشاء الرأس الخارجي

مراقبة تنفيذ QoS:

1. انتقل إلى القواعد → علامة تبويب QER
2. تحقق من أن قيم MBR و GBR تتطابق مع السياسة
3. تحقق من علامة QFI 5G لتدفقات

تتبع استخدام البيانات:

1. انتقل إلى القواعد → علامة تبويب URR
2. قم بفرز حسب الحجم الإجمالي للعثور على أعلى المستخدمين
3. راقب نمو الحجم بمرور الوقت
4. تحقق من تكامل الفوترة

إدارة المخازن

الرابط: `/buffers`

الميزات

يعرض عرض المخازن مخازن الحزم التي تحتفظ بها UPF أثناء أحداث التنقل أو تبديل المسارات.

الإحصائيات الإجمالية

تعرض لوحة المعلومات إحصائيات المخازن المجمع:

- **إجمالي الحزم** FARs: عدد الحزم المخزنة عبر جميع
- **إجمالي البايتات**: إجمالي حجم البيانات المخزنة
- **إجمالي FARs**: عدد FARs التي تحتوي على حزم مخزنة
- **الحد الأقصى لكل FAR**: الحد الأقصى للحزم المسموح بها لكل FAR
- **الحد الأقصى الإجمالي**: الحد الأقصى للحزم المخزنة
- **الحزمة TTL** (بالثواني): الوقت حتى انتهاء صلاحية الحزم المخزنة

المخازن حسب FAR

جدول لجميع FARs مع الحزم المخزنة:

العمود	الوصف
معرف FAR	معرف قاعدة إجراء إعادة التوجيه
عدد الحزم	عدد الحزم المخزنة لهذا FAR
عدد البايتات	إجمالي البايتات المخزنة لهذا FAR
أقدم حزمة	الطابع الزمني لأقدم حزمة مخزنة
أحدث حزمة	الطابع الزمني لأحدث حزمة مخزنة
الإجراءات	أزرار التحكم في المخزن (على شكل حبوب)

إجراءات التحكم في المخازن

لكل FAR مع حزم مخزنة، تتوفر الأزرار التالية على شكل حبوب:

تحكم التخزين:

- FAR يحدث علامة إجراء) FAR **تعطيل التخزين** (أحمر): إيقاف التخزين لهذا
- FAR **تمكين التخزين** (أرجواني): تشغيل التخزين لهذا

عمليات المخازن:

- الحالية FAR **تفريغ** (أزرق): إعادة تشغيل جميع الحزم المخزنة باستخدام قواعد
- **مسح** (رمادي): حذف جميع الحزم المخزنة دون إعادة توجيه

مسح جميع المخازن:

- زر "مسح جميع" الأحمر في الرأس
- يمسح المخازن لجميع FARs
- يتطلب تأكيد

حالات الاستخدام

مراقبة تخزين الانتقال:

1. أثناء الانتقال، تحقق من أن الحزم يتم تخزينها
2. تحقق من حالة تخزين FAR (يجب أن تكون مفعلّة)
3. راقب عدد الحزم وعمرها

إكمال الانتقال:

1. بعد تبديل المسار، انقر على "تفريغ" لإعادة تشغيل الحزم المخزنة
2. تحقق من أن الحزم قد تم إعادة توجيهها إلى المسار الجديد
3. انقر على "تعطيل التخزين" لإيقاف التخزين

مسح المخازن العالقة:

1. حدد FARs (تحقق من أقدم طابع زمني) مع حزم مخزنة قديمة
2. انقر على "مسح" للتخلص من الحزم القديمة
3. أو انقر على "تعطيل التخزين" لمنع المزيد من التخزين

استكشاف مشكلات تجاوز السعة:

1. تحقق من إجمالي عدد الحزم مقابل الحد الأقصى الإجمالي
2. حدد FARs مع تخزين مفرط
3. تحقق من أن SMF قد أرسل تعديل الجلسة لتعطيل التخزين
4. قم بتعطيل التخزين يدويًا إذا تم تفويت أمر SMF

التحديثات في الوقت الحقيقي

يقوم عرض المخازن بتحديث تلقائي كل 5 ثوانٍ لعرض حالة المخزن الحالية.

لوحة إحصائيات

الرابط: `/statistics`

الميزات

يوفر عرض الإحصائيات مقاييس الأداء في الوقت الحقيقي من مسار بيانات OmniUPF. لمزيد من المعلومات التفصيلية حول مقاييس Prometheus، راجع **مرجع المقاييس**.

إحصائيات الحزم

عدادات معالجة الحزم المجمعة:

- **حزم RX**: إجمالي الحزم المستلمة على ج��يع الواجهات
- **حزم TX**: إجمالي الحزم المرسلة على جميع الواجهات
- **حزم تم إسقاطها**: الحزم التي تم تجاهلها بسبب الأخطاء أو السياسة
- **حزم GTP-U**: الحزم المعالجة مع تغليف GTP-U

ومعدل إسقاط الحزم UPF **الاستخدام**: مراقبة الحمل العام لحركة مرور

إحصائيات المسارات

مقاييس إعادة التوجيه لكل مسار (إذا كانت متاحة):

- **ضربات المسار**: الحزم المطابقة لكل قاعدة توجيه
- **نجاح إعادة التوجيه**: عدد الحزم التي تم إعادة توجيهها بنجاح
- **أخطاء إعادة التوجيه**: محاولات إعادة التوجيه الفاشلة

الاستخدام: تحديد المسارات المزدحمة وأخطاء إعادة التوجيه

إحصائيات XDP

مقاييس أداء eXpress Data Path:

- **XDP المعالج**: إجمالي الحزم المعالجة في طبقة XDP
- **XDP المرسلة**: الحزم المرسلة إلى كومة الشبكة
- **XDP المرفوضة**: الحزم المرفوضة في طبقة XDP
- **XDP الملغاة**: أخطاء المعالجة في برنامج XDP

واكتشاف أخطاء المعالجة XDP **الاستخدام**: مراقبة أداء

أسباب إسقاط XDP:

- تنسيق حزمة غير صالح
- فشل البحث في خريطة eBPF
- إسقاطات قائمة على السياسة
- استنفاد الموارد

إحصائيات واجهة N3/N6

عدادات حركة المرور لكل واجهة:

واجهة N3 (اتصال RAN):

- **RX N3**: الحزم المستلمة من gNB/eNodeB
- **TX N3**: الحزم المرسلة إلى gNB/eNodeB

واجهة N6 (اتصال الشبكة البيانات):

- **RX N6**: الحزم المستلمة من الشبكة البيانات (الإنترنت/IMS)
- **TX N6**: الحزم المرسلة إلى الشبكة البيانات

الإجمالي: إجمالي عدد الحزم عبر الواجهات

الاستخدام: مراقبة توازن الحركة ومشكلات محددة بالواجهة

حالات الاستخدام

مراقبة حمل الحركة:

1. تحقق من معدلات RX/TX للحزم
2. تحقق من أن الحركة تتدفق في كلا الاتجاهين
3. قارن حركة N3 مقابل N6 (يجب أن تكون متساوية تقريبًا)

اكتشاف إسقاط الحزم:

1. تحقق من عداد الحزم المرفوضة
2. راجع عداد الحزم المرفوضة XDP
3. تحقق من السبب في السجلات إذا كانت الإسقاطات مرتفعة

تحليل الأداء:

1. راقب نسبة الحزم المعالجة إلى المرسلة XDP
2. تحقق من إسقاطات XDP (تشير إلى الأخطاء)
3. تحقق من توزيع حركة المرور على واجهات N3/N6

تخطيط السعة:

1. تتبع معدل الحزم بمرور الوقت
2. قارن مع حدود سعة UPF
3. خطط للتوسع إذا اقتربت من الحدود

التحديثات في الوقت الحقيقي

تقوم الإحصائيات بالتحديث تلقائيًا كل 5 ثوانٍ.

مراقبة السعة

الرابط: `/capacity`

الميزات

يعرض عرض السعة استخدام خريطة eBPF في مسار بيانات لجميع الخرائط وحدود السعة UPF.

جدول استخدام خريطة eBPF

جدول لجميع خرائط eBPF مع معلومات الاستخدام:

العمود	الوصف
اسم الخريطة	اسم خريطة eBPF (مثل `uplink_pdr_map`، `far_map`)
المستخدم	عدد الإدخالات الموجودة حاليًا في الخريطة
السعة	الحد الأقصى للإدخالات المسموح بها في الخريطة
الاستخدام	شريط تقدم مرئي مع النسبة المئوية
حجم المفتاح	حجم مفاتيح الخريطة بالبايتات
حجم القيمة	حجم قيم الخريطة بالبايتات

مؤشرات الاستخدام الملونة

يتم تلوين شريط تقدم الاستخدام بناءً على الاستخدام:

- **أخضر (<50%)**: تشغيل طبيعي، سعة كافية
- **أصفر (50-70%)**: تحذير، راقب النمو
- **كهرماني (70-90%)**: تحذير، خطط لزيادة السعة
- **أحمر (>90%)**: حرجة، يتطلب اتخاذ إجراء فوري

الخرائط الحرجة للمراقبة

uplink_pdr_map:

- يخزن PDRs الصاعدة المفاتيح بواسطة TEID
- إدخال واحد لكل تدفق حركة مرور صاعدة
- **حرجة**: الاستنفاد يمنع إنشاء جلسات جديدة

downlink_pdr_map / **downlink_pdr_map_ip6**:

- يخزن PDRs النازلة المفاتيح بواسطة عنوان IP لـ UE
- إدخال واحد لكل عنوان IPv4/IPv6 لـ UE
- **حرجة**: الاستنفاد يمنع إنشاء جلسات جديدة

far_map:

- يخزن قواعد إجر��ء إعادة التوجيه المفاتيح بواسطة معرف FAR
- مشتركة عبر عدة PDRs
- **أولوية عالية**: تؤثر على قرارات إعادة التوجيه

qer_map:

- يخزن قواعد تنفيذ QoS المفاتيح بواسطة معرف QER
- ولكن ليس على الاتصال الأساسي QoS **أولوية متوسطة**: تؤثر على

urr_map:

- يخزن قواعد تقارير الاستخدام المفاتيح بواسطة معرف URR
- **أولوية منخفضة**: تؤثر على الفوترة ولكن ليس على الاتصال

حالات الاستخدام

تخطيط السعة:

1. راقب اتجاهات استخدام الخريطة بمرور الوقت
2. حدد أي الخرائط تنمو بسرعة أكبر
3. خطط لزيادة السعة قبل الوصول إلى الحدود

منع فشل إنشاء الجلسات:

1. قبل زيادة حركة المرور المتوقعة PDR تحقق من استخدام خريطة
2. زيادة سعة الخريطة إذا اقتربت من الحدود
3. راقب بعد زيادة السعة للتحقق

استكشاف مشكلات الجلسات:

1. عندما يفشل إنشاء الجلسة، تحقق من عرض السعة
2. حمراء (>90%)، فإن السعة مستنفدة PDR إذا كانت خرائط
3. زيادة سعة الخريطة أو مسح الجلسات القديمة

تحسين تكوين الخريطة:

1. راجع أحجام المفاتيح والقيم

2. احسب استخدام الذاكرة لكل خريطة
3. تحسين أحجام الخرائط بناءً على أنماط الاستخدام الفعلية

تكوين السعة

يتم تكوين سعات خريطة eBPF عند بدء تشغيل UPF في ملف تكوين UPF. القيم النموذجية:

- نشر صغير: 10,000 - 100,000 إدخال لكل خريطة
- نشر متوسط: 100,000 - 1,000,000 إدخال لكل خريطة
- نشر كبير: 1,000,000+ إدخال لكل خريطة

حساب الذاكرة:

```
ذاكرة الخريطة = (حجم المفتاح + حجم القيمة) × السعة
```

على سبيل المثال، تستخدم خريطة PDR 64 مع 1 مليون إدخال وقيم بحجم 64 بايت حوالي 64 ميجابايت من ذاكرة النواة.

التحديثات في الوقت الحقيقي

يقوم عرض السعة بتحديث تلقائي كل 10 ثوانٍ.

عرض التكوين

الرابط: `/upf_config`

الميزات

يعرض عرض التكوين معلمات تشغيل UPF وتكوين مستوى البيانات.

تكوين UPF

يعرض تكوين UPF الثابت:

- **واجهة PFCP**: عنوان IP والمنفذ للاتصال بـ SMF/PGW-C
- **واجهة N3**: عنوان IP للاتصال بـ RAN (gNB/eNodeB)

- **واجهة N6**: عنوان IP للاتصال بالشبكة البيانات
- **واجهة N9**: عنوان IP للتواصل بين UPFs (اختياري)
- **منفذ API**: منفذ استماع واجهة برمجة التطبيقات REST
- **الإصدار**: إصدار برنامج OmniUPF

تكوين مستوى البيانات (eBPF)

يعرض معلمات مستوى البيانات النشطة في وقت التشغيل:

- **النشط** N3: ربط واجهة في وقت التشغيل **عنوان N3**
- **النشط** N9: ربط واجهة في وقت التشغيل (إذا تم تمكينه) **عنوان N9**

تعكس هذه القيم تكوين مسار eBPF الفعلي وقد تختلف عن التكوين الثابت إذا تم تغيير الواجهات.

حالات الاستخدام

تحقق من اتصال UPF:

1. تحقق من أن عنوان واجهة N3 يتطابق مع تكوين gNB
2. تحقق من أن واجهة N6 يمكنها التوجيه إلى الشبكة البيانات
3. تأكد من أن واجهة PFCP قابلة للوصول من SMF

استكشاف مشكلات الواجهة:

1. قارن التكوين الثابت مع العناوين النشطة في مستوى البيانات
2. تحقق من أن الواجهات مرتبطة بشكل صحيح
3. تحقق من تغييرات تكوين الواجهة

التوثيق والتدقيق:

1. سجل تكوين UPF للتوثيق
2. تحقق من أن النشر يتطابق مع مواصفات التصميم
3. تدقيق تعيينات الواجهة

عرض المسارات

الرابط: `/routes`

الميزات

يوفر عرض المسارات مراقبة شاملة لمسارات IP لمعدات المستخدم (UE) وجلسات بروتوكول التوجيه (BGP و OSPF).

نظرة عامة على حالة المسار

تعرض لوحة المعلومات إحصائيات المسار المجمعة:

- **الحالة**: تمكين أو تعطيل التوجيه
- **إجمالي المسارات**: إجمالي عدد مسارات IP لمعدات UE
- **متزامن**: عدد المسارات المتزامنة بنجاح
- **فشل**: عدد المسارات التي فشلت في التزامن

مسارات UE IP النشطة

جدول يعرض جميع مسارات IP النشطة لمعدات المستخدم:

العمود	الوصف
الفهرس	رقم فهرس المسار
عنوان UE IP	عنوان IPv4 أو IPv6 لـ UE المعين

الغرض:

- عرض جميع عناوين IP لـ UE التي تم تكوين مسارات لها
- التحقق من توزيع المسارات إلى بروتوكولات التوجيه
- مراقبة حالة تزامن المسارات

جيران OSPF

جدول لجيران بروتوكول OSPF (Open Shortest Path First):

العمود	الوصف
معرف الجار	معرف جهاز توجيه OSPF
العنوان	عنوان IP لجار OSPF
الواجهة	الواجهة المستخدمة لجوار OSPF
الحالة	حالة جوار OSPF (كاملة، أولية، إلخ)
الأولوية	قيمة أولوية OSPF
مدة التشغيل	المدة التي كان فيها الجار نشطًا
مدة الموت	الوقت حتى يعتبر الجار ميتًا

حالات OSPF:

- **كاملة** (أخضر): متجاورة بالكامل وتبادل معلومات التوجيه
- **حالات أخرى** (أصفر): تشكيل الجوار أو غير مكتمل

أقران BGP

جدول لأقران بروتوكول BGP (Border Gateway Protocol):

العمود	الوصف
عنوان الجار	عنوان IP للجار BGP
ASN	رقم النظام المستقل للجار
الحالة	حالة جلسة BGP (تأسست، خاملة، إلخ)
المدة	مدة الحالة الحالية
البادئات المستلمة	عدد بادئات المسار المستلمة من الجار
الرسائل المرسلة	إجمالي رسائل BGP المرسلة إلى الجار
الرسائل المستلمة	إجمالي رسائل BGP المستلمة من الجار

حالات BGP:

- نشطة، تتبادل المسارات BGP **تأسست** (أخضر): جلسة
- **حالات أخرى** (أحمر): الجلسة متوقفة أو في مرحلة التأسيس

يعرض الرأس أيضًا معرف جهاز توجيه BGP و ASN المحلي عندما يكون BGP مكونًا.

المسارات المعاد توزيعها OSPF

جدول يظهر LSAs الخارجية OSPF (إعلانات حالة الربط) المعاد توزيعها لـ UE للمسارات:

العمود	الوصف
معرف حالة الربط	معرف LSA (عادةً عنوان الشبكة)
القناع	قناع الشبكة للمسار
جهاز التوجيه المعلن	معرف جهاز التوجيه الذي يعلن عن هذا المسار الخارجي
نوع المقياس	نوع مقياس OSPF الخارجي (E1 أو E2)
المقياس	مقياس تكلفة OSPF للمسار
العمر	الوقت منذ أن تم إنشاء LSA (بالثواني)
رقم التسلسل	رقم تسلسل LSA للتحديثات

الغرض:

- التحقق من أن مسارات OSPF يتم إعادة توزيعها في UE
- مراقبة أي جهاز توجيه يعلن عن المسارات الخارجية
- تتبع عمر LSA والتحديثات

إجراءات التحكم في المسارات

زر مزامنة المسارات:

- يحفز FRR (Free Range Routing) مزامنة المسارات إلى UE يدويًا
- يجبر تحديث بروتوكول التوجيه مع المسارات الحالية لـ
- مفيد بعد تغييرات التكوين أو لاستعادة من فشل المزامنة

زر التحديث:

- يقوم بتحديث جميع معلومات المسار يدويًا
- يحدث جيران OSPF، أقران BGP، وجداول المسارات

حالات الاستخدام

مراقبة صحة بروتوكول التوجيه:

1. انتقل إلى عرض المسارات
2. تحقق من حالات جيران OSPF ("يجب أن تكون "كاملة)
3. تحقق من أن أقران BGP "تأسست"
4. تأكد من العدد المتوقع من الجيران/الأقران

التحقق من توزيع مسارات UE:

1. تحقق من جدول مسارات UE IP لـ UE النشطة محدد
2. انتقل إلى قسم المسارات المعاد توزيعها OSPF
3. تحقق من ظهور مسار UE في LSAs الخارجية
4. تأكد من أن جهاز التوجيه المعلن يتطابق مع UPF المتوقع

استكشاف مشكلات مزامنة المسارات:

1. تحقق من العدادات المتزامنة مقابل الفاشلة في نظرة الحالة
2. إذا كانت المسارات تفشل، انقر على زر "مزامنة المسارات"
3. راقب رسائل الخطأ في الشريط الأحمر إذا فشلت المزامنة
4. تحقق من رسائل الخطأ OSPF/BGP في الأقسام المعنية

التحقق من نشر متعدد UPF:

1. حدد مثيلات UPF المختلفة من القائمة المنسدلة
2. قارن عدد المسارات عبر المثيلات
3. تحقق من أن جيران OSPF يرون بعضهم البعض
4. تحقق من علاقات الاقتران BGP

مراقبة توسيع المسارات:

1. تتبع العدد الإجمالي للمسارات مع زيادة جلسات UE
2. تحقق من توزيع المسارات إلى بروتوكولات التوجيه
3. راقب نمو عدد LSAs OSPF
4. تحقق من عدد بادئات BGP المستلمة من الأقران

التحديثات في الوقت الحقيقي

يقوم عرض المسارات بتحديث تلقائي كل 10 ثوانٍ لعرض حالة بروتوكول التوجيه الحالية ومسارات UE.

تكامل التوجيه

يتكامل عرض المسارات مع FRR (Free Range Routing) الذي يعمل على UPF:

- **OSPF**: يتم إعادة توزيع المسارات كـ LSAs 2 خارجية من النوع
- **BGP**: يتم الإعلان عن المسارات إلى أقران BGP المكونة
- **آلية المزامنة**: تستدعي مكالمات واجهة برمجة التطبيقات REST أوامر vtysh لتحديث FRR

عرض قدرات XDP

الرابط: `/xdp_capabilities`

الميزات

يعرض عرض قدرات XDP دعم وضع eXpress Data Path (XDP) وقدرات الأداء وحسابات الإنتاجية لمسار بيانات UPF.

تكوين الواجهة

يعرض معلومات واجهة الشبكة والسائق:

الحقل	الوصف
اسم الواجهة	واجهة الشبكة المستخدمة لـ XDP (مثل eth0، ens1f0)
السائق	اسم سائق الشبكة (مثل i40e، ixgbe، virtio_net)
إصدار السائق	سلسلة إصدار السائق
الوضع الحالي	وضع XDP النشط (DRV أو SKB أو NONE)
عدد الطوابير المتعددة	عدد أزواج طوابير NIC للمعالجة المتوازية

أوضاع XDP

يعرض العرض جميع أوضاع XDP مع حالة دعمها وخصائص الأداء:

XDP_DRV (وضع السائق):

- **الأداء**: ~10-5 Mpps (ملايين الحزم في الثانية)
- **الوصف**: دعم XDP الأصلي في السائق، أعلى أداء
- **يتطلب**: سائق XDP الأصلي (i40e، ixgbe، mlx5، إلخ) مع دعم NIC
- **الحالة**: مدعوم إذا كان لدى السائق خطافات XDP
- **المؤشر**: علامة تحقق خضراء (✓) إذا كان مدعومًا، علامة X حمراء (✗) إذا لم يكن كذلك

XDP_SKB (الوضع العام):

- **الأداء**: ~1-2 Mpps
- **الوصف**: وضع احتياطي يستخدم كومة الشبكة في النواة
- **يتطلب**: أي واجهة شبكة
- **الحالة**: مدعوم دائمًا
- **المؤشر**: علامة تحقق خضراء (✓)

مؤشر الوضع الحالي:

- نقطة زرقاء بجوار وضع XDP النشط حاليًا
- يظهر أي وضع قيد الاستخدام فعليًا

أسباب عدم دعم الوضع:

- إذا كان الوضع غير مدعوم، يوضح حقل "السبب" السبب
- عدم توافق نوع الواجهة، XDP الأسباب الشائعة: السائق يفتقر إلى دعم

عرض قدرات XDP التفاعلية وحاسبة الإنتاجية، الأوضاع المدعومة، تكوين الواجهة يظهر Mpps

التوصيات

يعرض العرض شريط توصية ملون بناءً على التكوين الحالي:

أخضر (مثالي):

- "✓ مع دعم السائق الأصلي XDP_DRV مثالي: تم تمكين وضع"
- الوضع الأعلى أداءً نشط

أصفر (تحذير):

- "⚠ يُعتبر الترقية إلى وضع XDP_DRV للحصول على أداء أفضل"
- يعمل في الوضع العام عندما يكون وضع السائق متاحًا
- "⚠ تحذير: وضع XDP_DRV لسائق غير مدعوم من قبل هذا"

- القيود على الأجهزة تمنع الأداء الأمثل

أزرق (معلومات):

- معلومات عامة حول تكوين XDP

حاسبة أداء Mpps

حاسبة تفاعلية لتحويل معدل الحزم (Mpps) إلى الإنتاجية (Gbps):

معلمات الإدخال

معدل الحزم (Mpps):

- النطاق: 0.1 - 100 Mpps
- الافتراضي: الحد الأقصى لـ XDP الحالي لوضع Mpps
- يمثل ملايين الحزم المعالجة في الثانية

متوسط حجم الحزمة (بايت):

- النطاق: 64 - 9000 بايت
- الافتراضي: 1200 بايت (نموذجية حزمة GTP)
- يتضمن الحزمة الكاملة مع تغليف GTP

أزرار الإعداد السريع:

- **64B (الحد الأدنى)**: الحد الأدنى لحجم إطار Ethernet
- **128B**: حزم صغيرة
- **256B**: مستوى التحكم أو الإشارات
- **512B**: حزم متوسطة الحجم
- **1024B**: حزم كبيرة
- **1518B (الحد الأقصى)**: الحد الأقصى لحجم إطار Ethernet بدون إطارات ضخمة

نتائج الحساب

إجمالي الإنتاجية (Gbps):

- الإنتاجية بمعدل السلك بما في ذلك جميع الرؤوس
- الصي��ة: `Gbps = Mpps × Packet_Size × 8 / 1000`

- يتضمن رؤوس Ethernet وIP وUDP وGTP

معدل بيانات المستخدم (Gbps):

- الإنتاجية الفعلية للحمولة المستخدم
- يستبعد GTP ~50 بايت من تكلفة تغليف
- الصيغة: `Gbps = Mpps × (Packet_Size - 50) / 1000`

معدل الحزم:

- يعرض Mpps والحزم/الثانية مع فاصل الآلاف
- مثال: 10 Mpps = 10,000,000 حزمة/ثانية

عرض الصيغة:

- يظهر تفاصيل الحساب خطوة بخطوة
- مثال: `10 Mpps × 1200 بايت/بت × 8 ÷ 1000 = 96 Gbps`

فهم Mpps

يتضمن العرض قسم تفسير يغطي:

ما هو Mpps:

- ملايين الحزم في الثانية
- مقياس رئيسي لأداء معالجة الحزم
- مستقل عن حجم الحزمة

العلاقة بالإنتاجية:

- نفس Mpps مع حزم أكبر = Gbps أعلى
- نفس Mpps مع حزم أصغر = Gbps أقل
- تعتمد الإنتاجية على كل من المعدل وحجم الحزمة

تكلفة تغليف GTP:

- رأس Ethernet: 14 بايت
- رأس IP: 20 بايت (IPv4) أو 40 بايت (IPv6)

- رأس UDP: 8 بايت
- رأس GTP: 8 بايت (الحد الأدنى)
- إجمالي التكلفة النموذجية: ~50 بايت لكل حزمة

حالات الاستخدام

تقييم أداء XDP:

1. انتقل إلى عرض قدرات XDP
2. تحقق من وضع XDP الحالي (يجب أن يكون DRV للحصول على أفضل أداء)
3. لاحظ نطاق أداء Mpps
4. راجع شريط التوصية

حساب الإنتاجية المتوقعة:

1. أدخل معدل الحزمة المتوقع في Mpps
2. أدخل متوسط حجم الحزمة لملف حركة المرور الخاص بك
3. راجع الإنتاجية المحسوبة في Gbps
4. قارن مع سعة الرابط أو متطلبات الأداء

تحسين تكوين XDP:

1. تحقق مما إذا كان وضع XDP_DRV مدعومًا ولكن غير نشط
2. راجع إصدار السائق والتوافق
3. اتبع التوصية لترقية الوضع إلى وضع السائق إذا كان متاحًا
4. تحقق من أن عدد الطوابير المتعددة يتوافق مع أنوية CPU

تخطيط السعة:

1. استخدم الحاسبة لتحديد Mpps المطلوب للإنتاجية المستهدفة
2. قارن مع قدرات وضع XDP الحالي
3. حدد ما إذا كانت هناك حاجة لترقية الأجهزة
4. خطط لاختيار الواجهة والسائق للن��ر الجديد

استكشاف مشكلات الأداء:

1. تحقق من أن وضع XDP هو DRV، وليس SKB
2. تحقق من إصدار السائق لأية مشكلات أداء معروفة

3. تحقق من أن عدد الطوابير المتعددة كافٍ
4. احسب ما إذا كان الوضع الحالي يدعم الإنتاجية المطلوبة

نصائح لتحسين الأداء

وضع السائق (XDP_DRV):

- استخدم NICs مع دعم XDP الأصلي (Intel i40e/ixgbe، Mellanox mlx5)
- تحديث برامج تشغيل NIC إلى أحدث إصدار
- تمكين الطوابير المتعددة (RSS) للمعالجة المتوازية
- ضبط أحجام مخازن NIC

الوضع العام (XDP_SKB):

- مقبول للتطوير والاختبار
- غير موصى به للإنتاج عالي الإنتاجية
- النظر في ترقية الأجهزة للنشر الإنتاجي

تكوين الطوابير المتعددة:

- يجب أن يتطابق عدد الطوابير مع أو يتجاوز عدد أنوية CPU
- يمكّن المعالجة المتوازية للحزم عبر الأنوية
- يوزع الحمل عبر RSS (توزيع الحمل على جانب الاستقبال)

التحديثات في الوقت الحقيقي

يقوم عرض قدرات XDP بالتحديث كل 30 ثانية لتحديث حالة الواجهة ومعلومات الوضع.

عارض السجلات

الرابط: `/logs`

الميزات

عرض سجلات تطبيق OmniUPF في الوقت الحقيقي من لوحة التحكم.

الميزات:

- بث سجلات حية عبر Phoenix LiveView
- تحديثات في الوقت الحقيقي عند إنشاء السجلات
- تاريخ سجلات قابل للتمرير
- مفيد لاستكشاف الأخطاء وإصلاحها أثناء الجلسات النشطة

مستويات السجل

تستخدم سجلات OmniUPF مستويات Logger القياسية في Elixir:

- **DEBUG**: معلومات تشخيصية مفصلة
- **INFO**: رسائل معلومات عامة (افتراضي)
- **WARNING**: رسائل تحذير لمشكلات غير حرجة
- **ERROR**: رسائل خطأ للفشل

حالات الاستخدام

استكشاف مشكلات إنشاء الجلسة:

1. افتح عرض السجلات
2. ابدأ إنشاء الجلسة من SMF
3. راقب سجلات رسائل PFCP وأي أخطاء

مراقبة الاتصال PFCP:

1. عرض رسائل إعداد ارتباط PFCP
2. تتبع إنشاء/تعديل/حذف الجلسات
3. تحقق من رسائل نبض القلب

تصحيح مشكلات إعادة التوجيه:

1. ابحث عن أخطاء معالجة الحزم
2. تحقق من سجلات تشغيل خريطة eBPF
3. حدد مشكلات تكوين FAR/PDR

أفضل الممارسات

إرشادات التشغيل

المراقبة:

- تحقق بانتظام من عرض السعة لمنع استنفاد الخريطة
- راقب الإحصائيات للأنماط غير العادية أو الإسقاطات
- تتبع نمو عدد الجلسات بمرور الوقت
- راقب أخطاء معالجة XDP

إدارة المخازن:

- راقب المخازن أثناء سيناريوهات الانتقال
- امسح المخازن العالقة إذا تجاوزت الحزم TTL
- تحقق من تعطيل التخزين بعد إكمال الانتقال
- استخدم "تفريغ" بدلاً من "مسح" لتجنب فقدان الحزم

إدارة الجلسات:

- استخدم الفلاتر للعثور بسرعة على جلسات UE محددة
- قم بتوسيع الجلسات للتحقق من تكوين القاعدة
- قارن الجلسات عبر عدة مثيلات UPF
- تحقق من مؤشر الصحة قبل استكشاف الأخطاء

استكشاف الأخطاء:

- استخدم السجلا�� لاستكشاف الأخطاء في الوقت الحقيقي
- تحقق من عرض الجلسات للتحقق من اتصال UE
- تحقق من تكوين القواعد لتدفقات الحركة
- راقب الإحصائيات لإسقاط الحزم أو أخطاء إعادة التوجيه

الأداء

- يتم تحديث لوحة التحكم تلقائيًا كل 5-10 ثوانٍ حسب العرض
- قد تستغرق قوائم الجلسات الكبيرة وقتًا للتحميل

- يقوم عرض القواعد بتصفية الإدخالات النشطة فقط (URRs حجوم غير صفرية لـ)
- يتم تنفيذ عمليات المخازن على الفور على UPF المحدد

الوثائق ذات الصلة

- تكوين - **دليل إدارة القواعد** URR و QER و FAR و PDR
- **دليل المراقبة** - الإحصائيات والمقاييس وتخطيط السعة
- مرجع كامل لمقاييس Prometheus - **مرجع المقاييس**
- **رموز أسباب PFCP** - رموز أخطاء PFCP وتشخيص الجلسات
- مرجع واجهة برمجة التطبيقات - **توثيق واجهة برمجة التطبيقات** REST والتصفح
- تفاصيل توجيه - **دليل المس��رات** UE وتكامل FRR
- **دليل أوضاع XDP** - توثيق مفصل لوضع XDP ومعلومات eBPF
- مشكلات شائعة وتشخيصات - **دليل استكشاف الأخطاء**
- **دليل عمليات UPF** - عمليات UPF العامة والهندسة المعمارية

أوضاع إرفاق XDP لـ OmniUPF

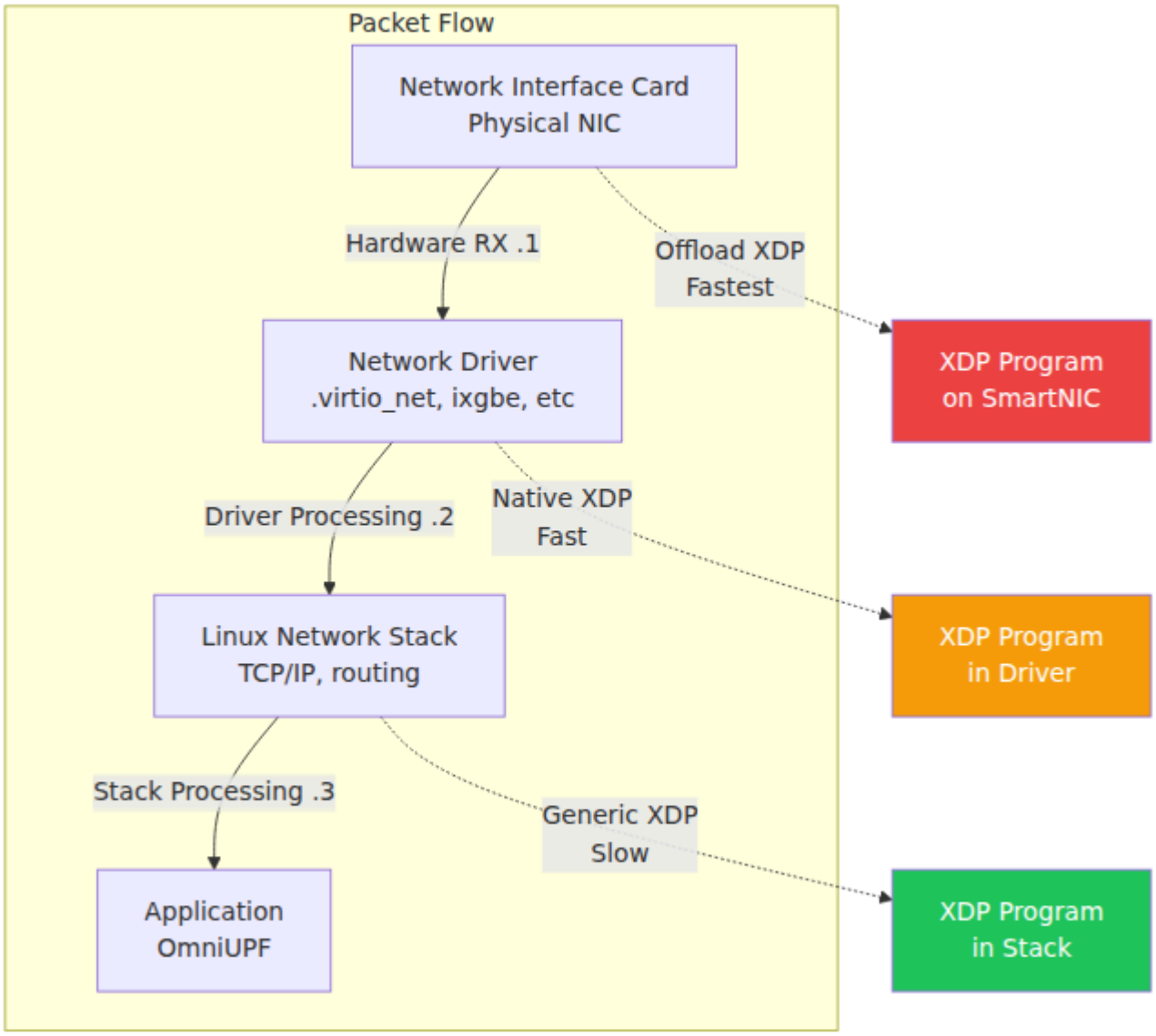
جدول المحتويات

1. نظرة عامة
2. مقارنة أوضاع XDP
3. الوضع العام (افتراضي)
4. الوضع الأصلي (موصى به للإنتاج)
5. وضع التحميل (SmartNIC)
6. تمكين XDP الأصلي على Proxmox VE
7. تمكين XDP الأصلي على Hypervisors الأخرى
8. التحقق من وضع XDP
9. استكشاف مشكلات XDP

نظرة عامة

يستخدم OmniUPF **XDP (مسار البيانات السريع)** لمعالجة الحزم عالية الأداء. XDP هي تقنية بالتشغيل في أقرب نقطة ممكنة في كومة نواة لينكس تسمح لبرامج معالجة الحزم (eBPF) الشبكة، مما يوفر زمن انتقال بمستوى الميكروثانية ومعدل نقل يصل إلى ملايين الحزم في الثانية.

يحدد وضع إرفاق XDP **أين** يتم تنفيذ برنامج eBPF في مسار الحزمة:



ويحدد ما إذا كنت تستطيع تحقيق OmniUPF الصحيح بشكل كبير على أداء XDP يؤثر اختيار وضع معالجة حزم بمستوى الإنتاج.

مقارنة أوضاع XDP

الجانب	الوضع العام	الوضع الأصلي	وضع التحميل
نقطة الإرفاق	كومة الشبكة في لينكس	رنامج التشغيل�� الشبكي	أجهزة NIC
الأداء	~1-2 Mpps	~5-10 Mpps	~10-40 Mpps
زمن الانتقال	~100 µs	~10 µs	~1 µs
استخدام وحدة المعالجة المركزية	مرتفع	متوسط	منخفض
متطلبات NIC	أي NIC	برنامج تشغيل يدعم XDP	مع دعم SmartNIC XDP
دعم Hypervisor	جميع Hypervisors	معظمها (يتطلب تعدد الطوابير)	نادر (PCI passthrough)
حالة الاستخدام	الاختبار، التطوير	**الإنتاج (موصى به)**	مواقع الحافة ذات النطاق الترددي العالي
التكوين	`xdp_attach_mode: generic`	`xdp_attach_mode: native`	`xdp_attach_mode: offload`

التوصية: استخدم **الوضع الأصلي** للنشر في الإنتاج. الوضع العام مناسب فقط للاختبار.

الوضع العام (افتراضي)

الوصف

في كومة الشبكة في لينكس **بعد** أن يقوم برنامج eBPF العام على تشغيل برنامج XDP يعمل ولكنه يعمل مع أي واجهة شبكة XDP التشغيل بمعالجة الحزمة. هذا هو أبطأ وضع.

خصائص الأداء

- **معدل النقل** (Mpps): ~1-2 مليون حزمة في الثانية
- **زمن الانتقال**: ~100 ميكروثانية لكل حزمة
- **عبء وحدة المعالجة المركزية**: مرتفع (الحزمة XDP تُنسخ إلى كومة النواة قبل)

متى تستخدم

- **التطوير والاختبار** فقط
- **بيئات المختبر** حيث لا تهم الأداء
- **النشر الأولي** للتحقق من الوظائف قبل تحسينها

التكوين

```yaml
# config.yaml
interface_name: [eth0]
xdp_attach_mode: generic  # الوضع الافتراضي
```

تحذير: الوضع العام **غير مناسب للإنتاج**. سيصبح عنق الزجاجة عند معدلات الحزم العالية ويهدر موارد وحدة المعالجة المركزية.

الوضع الأصلي (موصى به للإنتاج)

الوصف

يعمل XDP **داخل برنامج التشغيل الشبكي**، قبل أن تصل eBPF الأصلي على تشغيل برنامج الحزم إلى كومة الشبكة في لينكس. يوفر هذا أداءً قريبًا من الأجهزة مع الحفاظ على مرونة مستوى النواة.

خصائص الأداء

- **معدل النقل**: ~5-10 مليون حزمة في الثانية (Mpps) لكل نواة
- **زمن الانتقال**: ~10 ميكروثانية لكل حزمة
- **عبء وحدة المعالجة المركزية**: منخفض (الحزمة تُعالج على مستوى برنامج التشغيل)
- **التوسع** NIC: توسيع خطي مع أنوية وحدة المعالجة المركزية وطوابير

متى تستخدم

- **نشر الإنتاج** (موصى به)
- **الشبكات ذات الدرجة الناقلة** التي تتطلب نطاق ترددي عالي
- **سيناريوهات الحوسبة الحافة** مع متطلبات الأداء
- **أي نشر** حيث تهم الأداء

متطلبات برنامج تشغيل NIC

يتطلب XDP الأصلي برنامج تشغيل شبكة مع دعم XDP. تدعم معظم NICs الحديثة XDP الأصلي:

NICs الفيزيائية (bare metal):

- Intel: `ixgbe` (10G)، `i40e` (40G)، `ice` (100G)
- Broadcom: `bnxt_en`
- Mellanox: `mlx4_en`، `mlx5_core`
- Netronome: `nfp` (مع دعم التحميل)
- Marvell: `mvneta`، `mvpp2`

NICs الافتراضية (hypervisors):

- VirtIO: `virtio_net` (KVM، Proxmox، OpenStack) ✓
- VMware: `vmxnet3` ✓
- Microsoft: `hv_netvsc` (Hyper-V) ✓
- Amazon: `ena` (AWS) ✓
- SR-IOV: `ixgbevf`، `i40evf` (PCI passthrough) ✓

ملاحظة: VirtualBox لا يدعم **XDP** (استخدم الوضع العام فقط) **الأصلي**.

التكوين

```
# config.yaml
interface_name: [eth0]
xdp_attach_mode: native
```

متطلبات تعدد الطوابير: للحصول على أداء مثالي، قم بتمكين تعدد الطوابير على NICs الافتراضية (انظر قسم Proxmox أدناه).

وضع التحميل (SmartNIC)

الوصف

يعمل XDP المحمل على تشغيل برنامج eBPF **مباشرة على أجهزة NIC** (SmartNIC)، متجاوزًا وحدة المعالجة المركزية تمامًا لمعالجة الحزم. يوفر هذا أعلى أداء ولكنه يتطلب أجهزة متخصصة.

خصائص الأداء

- **معدل النقل** (Mpps): ~10-40 مليون حزمة في الثانية
- **زمن الانتقال**: ~1 ميكروثانية لكل حزمة
- **عبء وحدة المعالجة المركزية** (المعالجة على NIC): قريب من الصفر

متى تستخدم

- **نشر ذو نطاق ترددي فائق** (10G+ لكل مثيل UPF)

- **مواقع الحافة** مع تسريع الأجهزة
- **نشر حساس للتكلفة** (تقليل متطلبات وحدة المعالجة المركزية)

متطلبات الأجه��ة

تدعم فقط SmartNICs Agilio من Netronome حاليًا XDP المحمل:

- Netronome Agilio CX 10G/25G/40G/100G

ملاحظة: يتطلب وضع التحميل **PCI passthrough** أو **bare metal** - غير متاح في تكوينات VM القياسية.

التكوين

```
# config.yaml
interface_name: [eth0]
xdp_attach_mode: offload
```

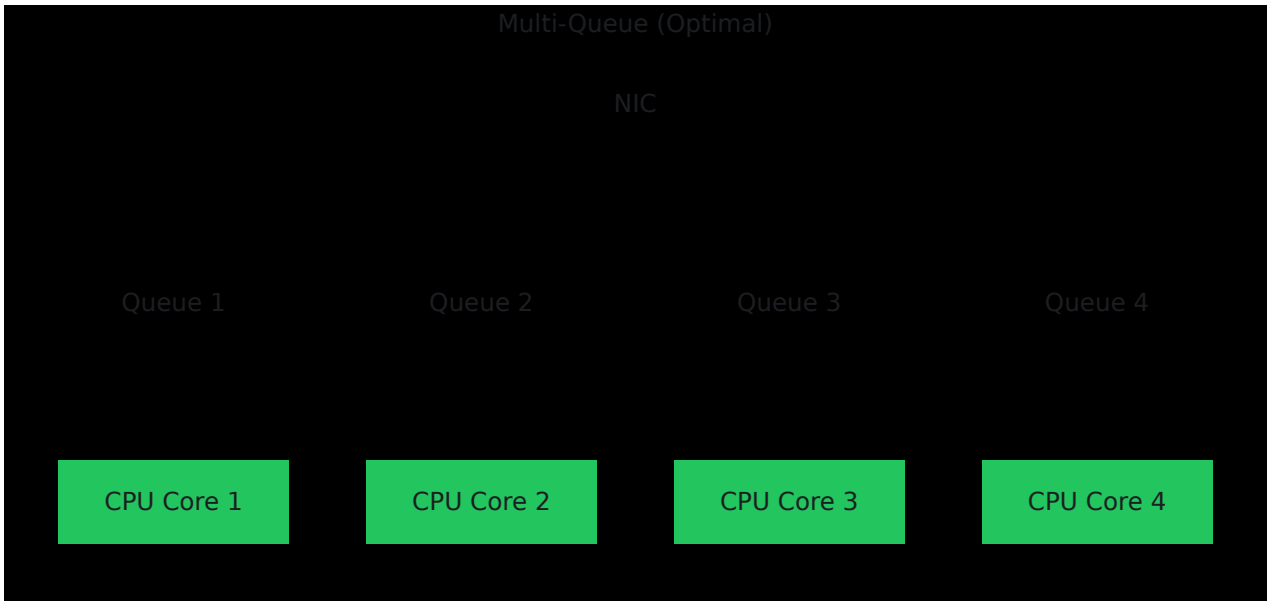
تمكين XDP الأصلي على Proxmox VE

يستخدم Proxmox VE أجهزة الشبكة **VirtIO** للـ VMs، والتي تدعم XDP الأصلي عبر برنامج التشغيل `virtio_net`. للحصول على أداء مثالي، يجب عليك تمكين **تعدد الطوابير**.

الخطوة 1: فهم المتطلبات

لماذا تهم تعدد الطوابير:

- **طابور واحد** (افتراضي): تتم معالجة جميع حركة المرور الشبكية بواسطة نواة وحدة معالجة مركزية واحدة → عنق الزجاجة
- **تعدد الطوابير**: يتم توزيع الحركة عبر عدة أنوية وحدة معالجة مركزية → توسيع خطي



الخطوة 2: تمكين تعدد الطوابير في Proxmox

الخيار أ: عبر واجهة ويب Proxmox

1. **قم بإيقاف تشغيل VM** (ليس فقط إعادة التشغيل) **تمامًا**
    - حدد VM الخاص بك في واجهة Proxmox
    - انقر على **إيقاف التشغيل**
2. **تحرير جهاز الشبكة**

- انتقل إلى علامة التبويب **الأجهزة**
- انقر على جهاز الشبكة الخاص بك (مثل `net0`)
- انقر على **تحرير**

3. **تعيين تعدد الطوابير**

- ابحث عن حقل **"تعدد الطوابير"**
- اضبطه على **8** (أو تطابق عدد vCPU 16 الخاص بك، بحد أقصى)
- انقر على **موافق**

4. **بدء VM**

- انقر على **بدء**

الخيار ب: عبر سطر أوامر Proxmox

```
# SSH إلى مضيف Proxmox الخاص بك

# ابحث عن معرف VM الخاص بك
qm list

# تعيين تعدد الطوابير (استبدل XXX بمعرف VM الخاص بك)
qm set XXX -net0 virtio=XX:XX:XX:XX:XX:XX,bridge=vmbr0,queues=8

# مثال لـ VM 191 مع MAC BC:24:11:1D:BA:00
qm set 191 -net0 virtio=BC:24:11:1D:BA:00,bridge=vmbr0,queues=8

# قم بإيقاف تشغيل VM
qm shutdown XXX

# انتظر حتى يتم إيقاف التشغيل، ثم ابدأ
qm start XXX
```

توصيات عدد الطوابير:

- **4** **طوابير**: الحد الأدنى للإنتاج (جيد لـ 2-4 VCPU VMs)
- **8** **طوابير**: موصى به لمعظم النشر (8-4 VCPU VMs)
- **16** **طابور**: الحد الأقصى للأداء العالي (8+ VCPU VMs)

الخطوة 3: تحقق من تعدد الطوابير داخل VM

بعد إعادة تشغيل VM، قم بتسجيل الدخول عبر SSH إلى VM وتحقق:

```
# تحقق من تكوين الطابور
ethtool -l eth0

# المخرجات المتوقعة:
# Channel parameters for eth0:
# Combined:     8         <-- يجب أن تتطابق مع القيمة التي قمت بتكوينها

# احسب الطوابير الفعلية
ls -1d /sys/class/net/eth0/queues/rx-* | wc -l
ls -1d /sys/class/net/eth0/queues/tx-* | wc -l

# يجب أن تظهر كلاهما 8 (أو القيمة التي قمت بتكوينها)
```

الخطوة 4: تمكين XDP الأصلي في OmniUPF

قم بتحرير تكوين OmniUPF:

```
# تحرير ملف التكوين
sudo nano /config.yaml
```

قم بتغيير وضع XDP:

```
# قبل
xdp_attach_mode: generic

# بعد
xdp_attach_mode: native
```

أعد تشغيل OmniUPF:

```
sudo systemctl restart omniupf
```

الخطوة 5: تحقق من أن XDP الأصلي نشط

تحقق من السجلات:

```
# عرض سجلات بدء التشغيل
journalctl -u omniupf --since "1 minute ago" | grep -i
"xdp\|attach"

# المخرجات المتوقعة:
# xdp_attach_mode:native
# XDPAttachMode:native
# Attached XDP program to iface "eth0" (index 2)
```

تحقق عبر API:

```
# استعلام التكوين
curl -s http://localhost:8080/api/v1/config | grep xdp_attach_mode

# المخرجات المتوقعة:
# "xdp_attach_mode": "native",
```

مشكلات Proxmox الشائعة

المشكلة: "فشل في إرفاق برنامج XDP"

الحل:

- تحقق من تمكين تعدد الطوابير (`ethtool -l eth0`)
- تحقق من إصدار النواة: `uname -r` (يجب أن يكون ≥ 5.15)
- تأكد من تحميل برنامج تشغيل VirtIO: `lsmod | grep virtio_net`

المشكلة: طابور واحد فقط على الرغم من التكوين

الحل:

- يجب أن يكون VM **مغلقًا تمامًا** (ليس معاد تشغيله) لتغييرات الطابور
- استخدم `qm shutdown XXX && sleep 5 && qm start XXX`
- تحقق في تكوين Proxmox: `grep net0 /etc/pve/qemu-server/XXX.conf`

المشكلة: الأداء لا يتحسن مع الوضع الأصلي

الحل:

- تحقق من تثبيت وحدة المعالجة المركزية (تجنب الإفراط في الاشتراك)
- يجب أن ينتشر استخدام وحدة المعالجة المركزية عبر الأنوية - راقب `top`
- تحقق من إحصائيات XDP: `curl http://localhost:8080/api/v1/xdp_stats`

تمكين XDP الأصلي على Hypervisors الأخرى

VMware ESXi / vSphere

تستخدم VMware برنامج التشغيل `vmxnet3` الذي يدعم XDP الأصلي.

المتطلبات:

- ESXi 6.7 أو أحدث
- إصدار برنامج تشغيل vmxnet3 1.4.16+ في VM
- إصدار الأجهزة VM 14 أو أحدث

تمكين تعدد الطوابير:

1. **قم بإيقاف تشغيل VM**
2. **تحرير إعدادات VM**:
   - انقر بزر الماوس الأيمن على VM → تحرير الإعدادات
   - محول الشبكة → متقدم
   - اضبط **تمكين توزيع جانب الاستقبال** على **مفعل**
3. **تحرير ملف .vmx** (اختياري، لمزيد من الطوابير):

```
ethernet0.pnicFeatures = "4"
ethernet0.multiqueue = "8"
```

4. **ابدأ VM وتحقق**:

```
ethtool -l ens192  # تحقق من عدد الطوابير
```

تكوين OmniUPF:

```
interface_name: [ens192]  # تستخدم VMware عادة ens192
xdp_attach_mode: native
```

KVM / libvirt (Raw)

تمكين تعدد الطوابير عبر virsh:

```
# تحرير تكوين VM
virsh edit your-vm-name
```

أضف إلى قسم واجهة الشبكة:

```
<interface type='network'>
  <source network='default'/>
  <model type='virtio'/>
  <driver name='vhost' queues='8'/>
</interface>
```

أعد تشغيل VM وتحقق:

```
ethtool -l eth0
```

Microsoft Hyper-V

تستخدم Hyper-V برنامج التشغيل `hv_netvsc` الذي يدعم XDP الأصلي.

المتطلبات:

- Windows Server 2016 أو أحدث
- خدمات تكامل لينكس 4.3+ في VM
- من الجيل الثاني VM

تمكين تعدد الطوابير:

PowerShell على مضيف Hyper-V:

```
# تعيين VMQ (طابور الآلة الافتراضية) - تعدد الطوابير في Hyper-V
Set-VMNetworkAdapter -VMName "YourVM" -VrssEnabled $true -
VmmqEnabled $true
```

تكوين OmniUPF:

```
interface_name: [eth0]
xdp_attach_mode: native
```

VirtualBox

تحذير: لا يدعم VirtualBox **XDP الأصلي**.

السبب: لا تقوم برامج تشغيل الشبكة في VirtualBox (e1000، virtio-net) بتنفيذ روابط XDP.

الحل: استخدم الوضع العام فقط:

```
xdp_attach_mode: generic  # الخيار الوحيد لـ VirtualBox
```

التحقق من وضع XDP

بعد تكوين XDP الأصلي، تحقق من أنه يعمل بشكل صحيح:

1. تحقق من سجلات OmniUPF

```
# عرض السجلات الأخيرة
journalctl -u omniupf --since "5 minutes ago" | grep -i xdp

# ابحث عن:
# ✓ "xdp_attach_mode:native"
# ✓ "Attached XDP program to iface"
# ✗ "Failed to attach" أو "falling back to generic"
```

2. تحقق عبر API

```
# استعلام نقطة النهاية للتكوين
curl -s http://localhost:8080/api/v1/config | jq .xdp_attach_mode

# المخرجات المتوقعة:
# "native"
```

3. ت��قق من إحصائيات XDP

```
# عرض إحصائيات معالجة XDP
curl -s http://localhost:8080/api/v1/xdp_stats | jq

# مثال على المخرجات:
{
  "xdp_aborted": 0,      # يجب أن تكون 0 (أخطاء)
  "xdp_drop": 1234,      # الحزم المفقودة
  "xdp_pass": 5678,      # الحزم المرسلة إلى الكومة
  "xdp_redirect": 9012,  # الحزم المعاد توجيهها
  "xdp_tx": 3456         # الحزم المرسلة
}
```

4. تحقق من دعم برنامج التشغيل

```
# تحقق مما إذا كان برنامج التشغيل يدعم XDP
ethtool -i eth0 | grep driver

# بالنسبة لـ Proxmox/KVM: يجب أن يظهر "virtio_net"
# بالنسبة لـ VMware: يجب أن يظهر "vmxnet3"
# بالنسبة لـ Hyper-V: يجب أن يظهر "hv_netvsc"
```

5. اختبار الأداء

قارن معالجة الحزم قبل وبعد:

```
# راقب معدل الحزم
watch -n 1 'curl -s http://localhost:8080/api/v1/packet_stats | jq
.rx_packets'

# الوضع العام: ~1-2 Mpps
# الوضع الأصلي: ~5-10 Mpps (تحسين 5-10x)
```

استكشاف مشكلات XDP

المشكلة: "فشل في إرفاق برنامج XDP" عند بدء التشغيل

الأع��اض:

```
خطأ: فشل في إرفاق برنامج XDP على الواجهة eth0
```

التشخيص:

1. **تحقق من دعم برنامج التشغيل**:

```
ethtool -i eth0 | grep driver

# إذا لم يكن برنامج التشغيل هو
virtio_net/vmxnet3/hv_netvsc، فلن يعمل XDP الأصلي
```

2. **تحقق من إصدار النواة**:

```
uname -r

# يجب أن يكون XDP بشكل موثوق ≥ 5.15 لدعم
```

3. **تحقق من وجود برامج XDP موجودة**:

```
ip link show eth0 | grep xdp

# إذا كان هناك برنامج XDP آخر متصل، قم بإلغاء تحميله أولاً
ip link set dev eth0 xdp off
```

الحل:

- قم بتحديث النواة إلى 5.15+ إذا كانت أقدم
- تأكد من تحميل برنامج تشغيل virtio_net: `modprobe virtio_net`
- الرجوع إلى الوضع العام إذا لم يدعم برنامج التشغيل XDP الأصلي

المشكلة: الوضع الأصلي يتراجع إلى الوضع العام

الأعراض:

```
تحذير: يتراجع إلى وضع XDP العام
```

التشخيص:

تحقق من `dmesg` لأخطاء برنامج التشغيل:

```
dmesg | grep -i xdp | tail -20
```

الأسباب الشائعة:

1. **برنامج التشغيل لا يدعم XDP الأصلي**:
   - برامج تشغيل VirtualBox (لا دعم لـ XDP الأصلي)
   - برامج تشغيل NICs القديمة
2. **تعدد الطوابير غير مفعل**:
   - تحقق: `ethtool -l eth0`
   - يجب أن يظهر > 1 طابور مشترك
3. **دعم XDP معطل في نواة لينكس**:

```
# تحقق مما إذا كان XDP مفعلًا في النواة
grep XDP /boot/config-$(uname -r)

# يجب أن يظهر:
# CONFIG_XDP_SOCKETS=y
# CONFIG_BPF=y
```

الحل:

- قم بتمكين تعدد الطوابير (انظر قسم Proxmox)
- قم بتحديث برنامج التشغيل المدعوم
- إعادة بناء النواة مع دعم XDP إذا لزم الأمر

المشكلة: الأداء لا يتحسن مع الوضع الأصلي

الأعراض: تم تمكين الوضع الأصلي ولكن معدل الحزم هو نفسه الوضع العام

التشخيص:

1. **تحقق من توزيع تعدد الطوابير**:

```
# تحقق من إحصائيات كل طابور
ethtool -S eth0 | grep rx_queue

# يجب أن يتم توزيع الحركة عبر عدة طوابير
```

2. **تحقق من استخدام وحدة المعالجة المركزية**:

```
# راقب استخدام وحدة المعالجة المركزية لكل نواة
mpstat -P ALL 1

# يجب أن ترى الحمل موزعًا عبر عدة أنوية
```

3. **تحقق من أن XDP يعمل فعليًا في الوضع الأصلي**:

```
# تحقق من bpftool (إذا كان متاحًا)
sudo bpftool net list

# يجب أن يظهر XDP متصلًا بالواجهة
```

الحل:

- زيادة عدد الطوابير (8-16 طابور)
- تمكين تثبيت وحدة المعالجة المركزية لتجنب هجرة النواة
- تحقق من الإفراط في الاشتراك في وحدة المعالجة المركزية على hypervisor

---

المشكلة: تم إلغاء برنامج XDP (xdp_aborted > 0)

الأعراض:

```
curl http://localhost:8080/api/v1/xdp_stats
{
  "xdp_aborted": 1234,  # غير صفر يدل على الأخطاء
  ...
}
```

التشخيص:

يعني إلغاء XDP أن برنامج eBPF واجه خطأ أثناء التنفيذ.

1. **تحقق من سجلات المدقق eBPF**:

```
dmesg | grep -i bpf | tail -20
```

2. **تحقق من حدود حجم الخريطة**:

```
# تكون خرائط eBPF ممتلئة ق
curl http://localhost:8080/api/v1/map_info

# ابحث عن الخرائط عند 100% من السعة
```

الحل:

- زيادة أحجام خريطة eBPF في التكوين
- تحقق من الحزم التالفة التي تسبب أخطاء eBPF
- تحقق من أن دعم eBPF في نواة لينكس كامل

المشكلة: تعدد الطوابير لا يعمل على Proxmox

الأعراض: يظهر طابور واحد فقط على الرغم من التكوين `ethtool -l eth0`

التشخيص:

1. **تحقق من تكوين VM في Proxmox**:

```
# على مضيف Proxmox
grep net0 /etc/pve/qemu-server/YOUR_VM_ID.conf

# يجب أن يظهر: queues=8
```

2. **تحقق من أن VM كان مغلقًا تمامًا**:

```
# على مضيف Proxmox
qm status YOUR_VM_ID

# يجب أن يظهر "status: stopped" قبل البدء
```

الحل:

```
# على مضيف Proxmox
# قم بإيقاف التشغيل بالقوة وأعد التشغيل
qm shutdown YOUR_VM_ID
sleep 10
qm start YOUR_VM_ID

# ثم تحقق داخل VM
ethtool -l eth0
```

إيقاف تشغيل بالكامل، وليس مجر** إعادة **VM مهم**: تتطلب تغييرات عدد الطوابير التشغيل من داخل VM.

المشكلة: تم رفض الإذن عند إرفاق XDP

الأعراض:

```
خطأ XDP: تم رفض الإذن عند إرفاق برنامج
```

التشخيص:

تتطلب عمليات XDP قدرات `CAP_NET_ADMIN` و `CAP_SYS_ADMIN`.

الحل:

1. **قم بتشغيل OmniUPF كـ root** (أو مع القدرات):

   ```
   sudo systemctl restart omniupf
   ```

2. **إذا كنت تستخدم systemd**، تحقق من أن ملف الخدمة يحتوي على القدرات:

```
# /lib/systemd/system/omniupf.service
[Service]
CapabilityBoundingSet=CAP_NET_ADMIN CAP_SYS_ADMIN
CAP_NET_RAW
AmbientCapabilities=CAP_NET_ADMIN CAP_SYS_ADMIN
CAP_NET_RAW
```

3. **إذا كنت تستخدم Docker**، قم بالتشغيل مع `--privileged`:

```
docker run --privileged -v /sys/fs/bpf:/sys/fs/bpf ...
```

ملخص تأثير الأداء

مقارنة الأداء في العالم الحقيقي لمعالجة الحزم في OmniUPF:

السيناريو	الوضع العام	الوضع الأصلي	التحسين
معدل الحزم	1.5 Mpps	8.2 Mpps	**أسرع 5.5x**
زمن الانتقال	95 µs	12 µs	**أقل 8x**
استخدام وحدة المعالجة المركزية (1 Gbps)	85% (نواة واحدة)	15% (موزعة)	**أكثر 5x كفاءة**
الحد الأقصى للنطاق الترددي	~1.2 Gbps	~10 Gbps	**أعلى 8x**

التوصية: استخدم دائمًا **الوضع الأصلي** مع **تمكين تعدد الطوابير** للنشر في الإنتاج.

توصيات الأجهزة لـ XDP

⚠ مهم: قبل شراء أي أجهزة، استشر دعم Omnitouch %100 للتأكد من توافقها مع تكوينك ومتطلبات النشر المحددة.

NICs المعروفة الجيدة لـ XDP الأصلي

هذه NICs تم التحقق من دعمها لوضع XDP الأصلي مع OmniUPF:

NICs من Intel (موصى بها لـ Bare Metal)

النموذج	السرعة	برنامج التشغيل	دعم XDP	الملاحظات
Intel X520	10GbE	ixgbe	✓ أصلي	مثبت، متاح على نطاق واسع، سعر/أداء جيد
Intel X710	10/40GbE	i40e	✓ أصلي	دعم ممتاز لتعدد الطوابير
Intel E810	100GbE	ice	✓ أصلي	الجيل الأحدث، أفضل أداء
Intel i350	1GbE	igb	✓ أصلي (النواة +5.10)	جيد لاحتياجات النطاق الترددي المنخفض

NICs من Mellanox/NVIDIA (أداء عالي)

النموذج	السرعة	برنامج التشغيل	دعم XDP	الملاحظات
ConnectX-4	25/50/100GbE	mlx5	أصلي ✓	نطاق ترددي عالي، جيد للحوسبة الحافة
ConnectX-5	25/50/100GbE	mlx5	أصلي ✓	أداء ممتاز، تسريع الأجهزة
ConnectX-6	50/100/200GbE	mlx5	أصلي ✓	الجيل الأحدث، أفضل للنطاق الترددي الفائق
BlueField-2	100/200GbE	mlx5	أصلي ✓	SmartNIC مع قدرات DPU

NICs من Broadcom

النموذج	السرعة	برنامج التشغيل	دعم XDP	الملاحظات
BCM57xxx series	10/25/50GbE	bnxt_en	✓ أصلي	شائع في خوادم Dell/HP

NICs الافتراضية (نشر VMs)

المنصة	نوع NIC	برنامج التشغيل	دعم XDP	تعدد الطوابير	الملاحظات
Proxmox/KVM	VirtIO	virtio_net	أصلي ✓	نعم (قابل للتكوين)	**أفضل لـ VMs**
VMware ESXi	vmxnet3	vmxnet3	أصلي ✓	نعم	يتطلب ESXi 6.7+
Hyper-V	NIC صناعي	hv_netvsc	أصلي ✓	نعم	Windows Server 2016+
AWS	ENA	ena	أصلي ✓	نعم	EC2 metal instances
VirtualBox	أي	مختلف	فقط عام	لا	غير موصى به للإنتاج

مع دعم التحميل للأجهزة NICs

التحميل الحقيقي لـ XDP (يعمل على NIC):

البائع	النموذج	السرعة	الملاحظات
Netronome	Agilio CX 10G	10GbE	الدعم الوحيد المؤكد لـ XDP المحمل
Netronome	Agilio CX 25G	25GbE	يتطلب برنامج ثابت خاص
Netronome	Agilio CX 40G	40GbE	باهظ الثمن جدًا (~$2,500-5,000)
Netronome	Agilio CX 100G	100GbE	فقط للدرجة المؤسسية

ملاحظة: NICs مع التحميل للأجهزة نادرة، باهظة الثمن، وتتطلب نشرًا على bare metal. يجب أن تستخدم معظم النشر XDP الأصلي بدلاً من ذلك.

التكوينات المختبرة

تم التحقق من هذه التكوين��ت مع OmniUPF في الإنتاج:

خيار الميزانية (1-10 Gbps)

- **NIC**: Intel X520 (10GbE مزدوج المنفذ)
- **الوضع**: XDP الأصلي
- **معدل النقل**: ~8-10 Gbps لكل مثيل UPF
- **التكلفة**: ~100-200$ (مستعمل/معاد تجديده)

النطاق المتوسط (10-50 Gbps)

- **NIC**: Intel X710 (40GbE) أو Mellanox ConnectX-4 (25GbE)
- **الوضع**: XDP الأصلي
- **معدل النقل**: ~25-40 Gbps لكل مثيل UPF
- **التكلفة**: ~300-800$

عالي الأداء (50+ Gbps)

- **NIC**: Mellanox ConnectX-5/6 (100GbE)
- **الوضع**: XDP الأصلي
- **معدل النقل**: ~80-100 Gbps لكل مثيل UPF
- **التكلفة**: ~1,000-2,500$

نشر VMs (Proxmox/KVM)

- **NIC**: VirtIO مع 8-16 طابور
- **الوضع**: XDP الأصلي
- **معدل النقل**: ~5-10 Gbps لكل مثيل UPF
- **التكلفة**: لا تكلفة إضافية على الأجهزة

ما يجب عدم شرائه

تجنب هذه للأجهزة OmniUPF في الإنتاج:

NIC/المنصة	السبب	البديل
NICs من Realtek	لا دعم لـ XDP، برامج تشغيل لينكس ضعيفة	Intel i350 أو أفضل
VirtualBox	لا دعم لـ XDP الأ��لي	الانتقال إلى Proxmox/KVM
NICs من الدرجة الاستهلاكية	دعم محدود للطوابير، غير موثوق	NICs من الدرجة الخادمة من Intel/Mellanox
NICs القديمة جدًا (2014>)	لا دعم لبرنامج تشغيل XDP	Intel X520 أو أحدث

قائمة التحقق قبل الشراء

قبل شراء الأجهزة، تحقق من:

1. ☐ **دعم برنامج التشغيل** XDP: تحقق مما إذا كان برنامج تشغيل لينكس يدعم

   ```
   # على نظام مشابه
   modinfo <driver_name> | grep -i xdp
   ```

2. ☐ **إصدار النواة**: تأكد من أن النواة ≥ 5.15 لدعم XDP بشكل موثوق

   ```
   uname -r
   ```

3. ☐ **تعدد الطوابير**: تحقق مما إذا كان NIC يدعم الطوابير المتعددة (RSS/VMDq)

4. ☐ **عرض PCI**: تأكد من أن فتحة PCIe تحتوي على عدد كافٍ من الخطوط

   - 10GbE: PCIe 2.0 x4 الحد الأدنى
   - 40GbE: PCIe 3.0 x8 الحد الأدنى
   - 100GbE: PCIe 3.0 x16 أو PCIe 4.0 x8

5. ☐ **نوع النشر**:

- Bare metal: يتطلب NIC الفيزيائية
- VM: دعم VirtIO أو SR-IOV مطلوب
- المضيف NIC حاوية: يتم وراثة تكوين

⚠ أولاً Omnitouch لا تشترِ الأجهزة بناءً على هذا الدليل فقط - تأكد دائمًا مع دعم!

موارد إضافية

- **دليل التكوين**: CONFIGURATION.md - مرجع تكوين كامل
- **دليل استكشاف الأخطاء**: TROUBLESHOOTING.md - تشخيص شامل للمشكلات
- **دليل العمارة**: ARCHITECTURE.md - تفاصيل عمارة XDPو eBPF
- **دليل المراقبة**: MONITORING.md - مراقبة الأداء والإحصائيات

مرجع سريع

إعداد XDP الأصلي على Proxmox (TL;DR)

```
# على مضيف Proxmox:
qm set <VM_ID> -net0 virtio=<MAC>,bridge=vmbr0,queues=8
qm shutdown <VM_ID> && sleep 10 && qm start <VM_ID>

# داخل VM:
ethtool -l eth0  # تحقق من 8 طوابير
sudo nano /etc/omniupf/config.yaml  # تعيين: xdp_attach_mode:
native
sudo systemctl restart omniupf
journalctl -u omniupf --since "1 min ago" | grep xdp  # تحقق من
الوضع الأصلي
```

تحقق من أن وضع XDP نشط

```
# تحقق من التكوين
curl -s http://localhost:8080/api/v1/config | grep xdp_attach_mode

# تحقق من الإحصائيات
curl -s http://localhost:8080/api/v1/xdp_stats | jq

# تحقق من الطوابير
ethtool -l eth0
```

وثائق واجهة برمجة تطبيقات OmniUPF

نظرة عامة

توفر واجهة برمجة تطبيقات OmniUPF واجهة RESTful شاملة لإدارة ومراقبة وظيفة مستوى المستخدم المعتمدة على eBPF. تتيح واجهة برمجة التطبيقات التحكم في الوقت الحقيقي والرؤية لجميع مكونات UPF.

قدرات واجهة برمجة التطبيقات

إدارة الجلسات:

- **جلسات PFCP**: استعلام عن الجلسات النشطة، عرض تفاصيل الجلسة، تصفية حسب عنوان IP لـ UE أو TEID
- **جمعيات PFCP**: مراقبة جمعيات عقدة التحكم وحالتها

قواعد المرور:

- **قواعد اكتشاف الحزم (PDR)**: فحص مصنفات حركة المرور في الاتجاه الصاعد والهابط (IPv4/IPv6)
- **قواعد إجراءات التوجيه (FAR)**: عرض سياسات التوجيه والتخزين والإسقاط
- **قواعد تنفيذ QoS (QER)**: مراقبة تحديد المعدل وسياسات QoS
- **قواعد تقارير الاستخدام (URR)**: تتبع عدادات حجم البيانات لكل جلسة

تخزين الحزم:

- **حالة التخزين**: عرض الحزم المخزنة لكل FAR (`GET /buffer`, `GET /buffer/:far_id`)
- **عمليات التخزين**: تفريغ أو مسح الحزم المخزنة (`POST /buffer/:far_id/flush`, `DELETE /buffer/:far_id`, `DELETE /buffer`)
- **تحكم التخزين**: تنبيه يدوي (`POST /buffer/:far_id/notify`)
- **حالة الإشعارات**: عرض حالة إشعار DLDR (`GET /buffer/notifications`)

المراقبة والإحصائيات:

- **إحصائيات الحزم**: عدادات الحزم في الوقت الحقيقي حسب البروتوكول (GTP، IP، TCP، UDP، ICMP، ARP)
- **إحصائيات XDP**: مقاييس أداء مسار البيانات (تمرير، إسقاط، إعادة توجيه، إلغاء)
- **إحصائيات واجهة N3/N6**: توزيع حركة المرور لشبكة RAN والشبكة البيانات
- **إحصائيات التوجيه**: أداء البحث في FIB (ضربات التخزين المؤقت، عمليات البحث، الأخطاء)

إدارة التوجيه:

- **طرق UE**: استعلام عن جدول توجيه عنوان IP لـ UE (`GET /routes`)
- **تكامل FRR**: مزامنة الطرق مع خادم توجيه Free Range (`POST /routes/sync`)
- **جلسات التوجيه**: عرض جلسات بروتوكول التوجيه (`GET /routing/sessions`)
- **قاعدة بيانات OSPF**: استعلام عن قاعدة بيانات الطرق الخارجية لـ OSPF (`GET /ospf/database/external`)

التكوين:

- **تكوين UPF**: استرجاع وتحرير التكوين (`GET /config`, `POST /config`)
- **تكوين مسار البيانات**: استعلام عن تكوين خاص بمسار البيانات (`GET /dataplane_config`)
- **قدرات XDP**: استعلام عن دعم وضع XDP وقدرات الواجهة (`GET /xdp_capabilities`)
- **سعة خريطة eBPF**: مراقبة استخدام الموارد والسعة (`GET /map_info`)

تكامل واجهة المستخدم على الويب

تم بناء واجهة المستخدم على الويب لـ OmniUPF على هذه الواجهة وتوفر لوحة معلومات تفاعلية لجميع وظائف واجهة برمجة التطبيقات. راجع دليل واجهة المستخدم على الويب للحصول على لقطات شاشة وأمثلة على الاستخدام.

وثائق واجهة برمجة التطبيقات Swagger

تم توثيق واجهة ب��مجة التطبيقات بالكامل باستخدام مواصفة **OpenAPI 3.0 (Swagger)**. توفر واجهة Swagger التفاعلية:

- توثيق شامل لنقاط النهاية مع مخططات الطلب/الاستجابة
- وظيفة تجربة الاستخدام لاختبار استدعاءات واجهة برمجة التطبيقات مباشرة من المتصفح
- تعريفات المخططات لجميع نماذج البيانات
- رموز حالة HTTP واستجابات الأخطاء

واجهة Swagger التفاعلية تعرض نقاط نهاية واجهة برمجة تطبيقات OmniUPF مع توثيق مفصل.

الوصول إلى واجهة Swagger

تتوفر وثائق Swagger على:

```
http://<upf-host>:8080/swagger/index.html
```

على سبيل المثال: `http://10.98.0.20:8080/swagger/index.html`

مسار قاعدة واجهة برمجة التطبيقات

تبدأ جميع نقاط نهاية واجهة برمجة التطبيقات بـ:

```
/api/v1
```

```
## ميزات واجهة برمجة التطبيقات

### الترقيم

الترقيم لنقاط النهاية التي تعي OmniUPF تدعم واجهة برمجة تطبيقات
مجموعات بيانات كبيرة. يمنع الترقيم انتهاء المهلة ويقلل من استخدام
الذاكرة عند الاستعلام عن آلاف الجلسات، PDRs، أو URRs.

**أنماط الترقيم المدعومة**:

1. **الترقيم القائم على الصفحات** (موصى به):
   - `page`: رقم الصفحة (يبدأ من 1)
   - `page_size`: العناصر لكل صفحة (افتراضي: 100، الحد الأقصى: 1000)

2. **الترقيم القائم على الإزاحة**:
   - `offset`: عدد العناصر التي يجب تخطيها
   - `limit`: عدد العناصر التي يجب إرجاعها (الحد الأقصى: 1000)

**طلبات المثال**:

```bash
# الترقيم القائم على الصفحات: الحصول على الصفحة الثانية مع 50
عنصرًا لكل صفحة
GET /api/v1/pfcp_sessions?page=2&page_size=50

# الترقيم القائم على الإزاحة: تخطي أول 100 عنصر، إرجاع 50 التالية
GET /api/v1/pfcp_sessions?offset=100&limit=50

# السلوك الافتراضي (بدون معلمات ترقيم): أول 100 عنصر
GET /api/v1/pfcp_sessions
```

تنسيق الاستجابة:
```

```
&#123;
  "data": [
    &#123; /* session object */ &#125;,
    &#123; /* session object */ &#125;,
    ...
  ],
  "pagination": &#123;
    "total": 5432,
    "page": 2,
    "page_size": 50,
    "total_pages": 109
  &#125;
&#125;
```

نقاط النهاية المرقمة:

- `/api/v1/pfcp_sessions` - قائمة جلسات PFCP
- `/api/v1/pfcp_associations` - قائمة جمعيات PFCP
- `/api/v1/routes` - طرق UE IP
- `/api/v1/uplink_pdr_map` - PDRs (معلومات أساسية) في الاتجاه الصاعد
- `/api/v1/uplink_pdr_map/full` - PDRs في الاتجاه الصاعد مع تفاصيل فلتر SDF الكاملة
- `/api/v1/downlink_pdr_map` - PDRs في الاتجاه الهابط IPv4 (معلومات أساسية)
- `/api/v1/downlink_pdr_map/full` - PDRs في الاتجاه الهابط IPv4 مع تفاصيل فلتر SDF الكاملة
- `/api/v1/downlink_pdr_map_ip6` - PDRs في الاتجاه الهابط IPv6 (معلومات أساسية)
- `/api/v1/downlink_pdr_map_ip6/full` - PDRs في الاتجاه الهابط IPv6 مع تفاصيل فلتر SDF الكاملة
- `/api/v1/far_map` - قواعد إجراءات التوجيه
- `/api/v1/qer_map` - قواعد تنفيذ QoS
- `/api/v1/urr_map` - قواعد تقارير الاستخدام

نقاط نهاية إدارة التخزين:

- `GET /api/v1/buffer` - قائمة بجميع مخازن FAR مع الإحصائيات
- `GET /api/v1/buffer/:far_id` - الحصول على حالة التخزين لـ FAR محدد

- `GET /api/v1/buffer/notifications` - قائمة بحالة إشعارات DLDR
- `DELETE /api/v1/buffer` - مسح جميع الحزم المخزنة
- `DELETE /api/v1/buffer/:far_id` - مسح التخزين لـ FAR محدد
- `POST /api/v1/buffer/:far_id/flush` - تفريغ (إعادة تشغيل) الحزم المخزنة
- `POST /api/v1/buffer/:far_id/notify` - إرسال إشعار DLDR يدويًا

نقاط نهاية التكوين:

- `GET /api/v1/config` - الحصول على تكوين UPF الحالي
- `POST /api/v1/config` - تحديث تكوين UPF (حقول قابلة للتحرير في وقت التشغيل)
- `GET /api/v1/dataplane_config` - الحصول على تكوين خاص بمسار البيانات

نقاط نهاية تكامل التوجيه:

- `GET /api/v1/routes` - قائمة طرق UE
- `POST /api/v1/routes/sync` - تفعيل مزامنة الطرق مع FRR
- `GET /api/v1/routing/sessions` - الحصول على جلسات بروتوكول التوجيه
- `GET /api/v1/ospf/database/external` - الحصول على قاعدة بيانات OSPF الخارجية LSA

أفضل الممارسات:

- استخدم `page_size=100` لعرض واجهة المستخدم على الويب
- استخدم `page_size=1000` للتصديرات الكبيرة (الحد الأقصى)
- استعلام `pagination.total_pages` لتحديد عدد التكرارات
- زيادة `page_size` لتحسين أداء واجهة برمجة التطبيقات (طلبات أقل)

دعم CORS

تم تفعيل مشاركة الموارد عبر الحدود (CORS) بشكل افتراضي لجميع نقاط نهاية واجهة برمجة التطبيقات، مما يسمح لواجهة المستخدم على الويب والتطبيقات الخارجية باستهلاك واجهة برمجة التطبيقات من أصول مختلفة.

مقاييس Prometheus

يعرض OmniUPF مقاييس Prometheus على REST، بالإضافة إلى واجهة برمجة التطبيقات نقطة النهاية `/metrics` (`9090:` المنفذ الافتراضي).

توفر المقاييس:

- عدادات رسائل PFCP والكمون لكل نظير
- إحصائيات الحزم حسب نوع البروتوكول
- أحكام إجراء XDP
- إحصائيات التخزين
- استخدام سعة خريطة eBPF
- تتبع حجم URR

راجع مرجع المقاييس للحصول على الوثائق الكاملة.

الوثائق ذات الصلة

- **دليل واجهة المستخ��م على الويب** - لوحة معلومات تفاعلية مبنية على هذه الواجهة
- وثائق مقاييس Prometheus - **مرجع المقاييس**
- **رموز أسباب PFCP** - رموز أخطاء PFCP واستكشاف الأخطاء وإصلاحها
- تكوين - **دليل إدارة القواعد** PDR، FAR، QER، URR
- تكامل - **دليل إدارة الطرق** UE وتوجيه FRR
- مراقبة الإحصائيات وتخطيط السعة - **دليل المراقبة**
- خيارات تكوين - **دليل التكوين** UPF
- **Swagger UI** - وثائق واجهة برمجة التطبيقات التفاعلية (استبدل `localhost` بمضيف UPF الخاص بك)

إدارة مسارات UE

الوثائق ذات الصلة:

- مرجع API - وثائق API الكامل بما في ذلك نقاط نهاية إدارة المسارات
- عمليات واجهة المستخدم على الويب والمراقبة - دليل العمليات

نظرة عامة

يتكامل UPF (وظيفة مستوى المستخدم) مع **FRR (التوجيه الحر)** لإدارة مسارات IP لمعدات المستخدم (UE). ديناميكيًا UE، يتكيف بنية التوجيه. يضمن هذا التكامل أنه مع إنشاء أو إنهاء جلسات UE، تلقائيًا لتعكس الطوبولوجيا الحالية للشبكة.

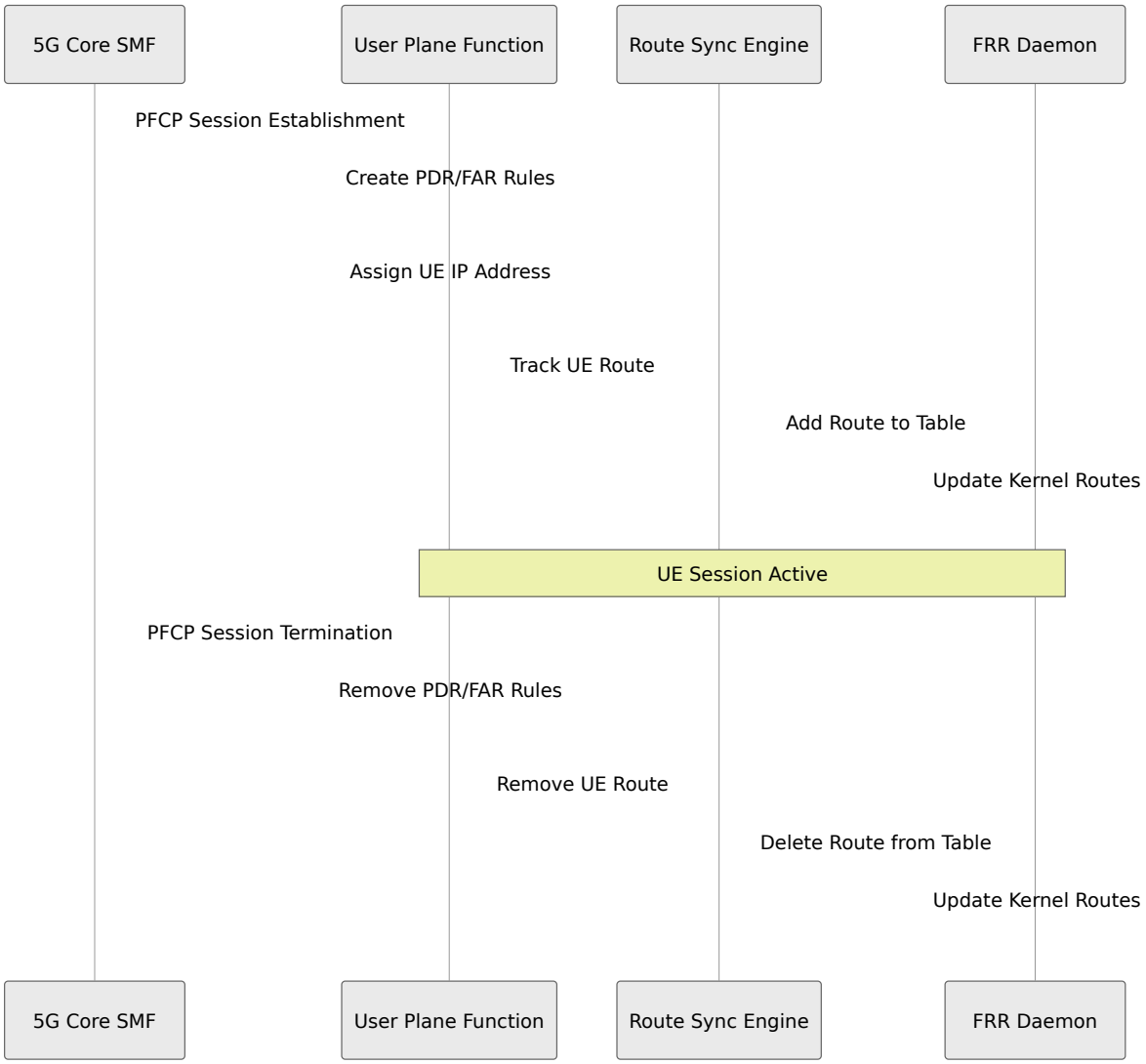
ما هو FRR؟

FRR (التوجيه الحر) هو مجموعة بروتوكولات توجيه قوية ومفتوحة المصدر لأنظمة Linux وUnix. يقوم BGP وOSPF وRIP في ذلك بتنفيذ بروتوكولات توجيه مختلفة بما في و��يرها. كخادم توجيه يحافظ على جدول توجيه النواة ويمكنه إعادة توزيع المسارات إلى FRR نشرنا، يعمل عناصر الشبكة الأخرى.

الهيكلية

كيف تعمل مزامنة المسارات

دورة حياة المسار



المزامنة التلقائية

يحافظ UPF النشطة لمعدات المستخدم. عند تمكينها، يقوم IP على سجل داخلي لجميع عناوين نظام مزامنة المسارات بـ:

1. **مراقبة جلسات UE**: تتبع جميع جلسات PFCP وعناوين IP النشطة المرتبطة بها
2. **الحفاظ على قائمة المسارات**: الاحتفاظ بقائمة محدثة من المسارات التي تحتاج إلى أن تكون في جدول التوجيه
3. **المزامنة مع FRR**: دفع تحديثات المسار تلقائيًا إلى خادم FRR عبر واجهته البرمجية

4. **التعامل مع الفشل**: تتبع حالة المزامنة (مزامنة/فشلت) لكل مسار وإعادة المحاولة حسب الحاجة

إعداد FRR

التكوين

يتم نشر FRR **قوالب** وتكوينه باستخدام **Ansible** لتحديد معلمات التوجيه الأساسية. تقوم بتعريف Ansible الخاص بك، وتقوم Ansible في كتاب تشغيل **Jinja2** مرة واحدة كقالب FRR تكوين الخاصة بك أثناء النشر UPF تلقائيًا بنشره إلى جميع مثيلات.

يتضمن قالب تكوين FRR Jinja2 النموذجي:

```
frr version 7.2.1
frr defaults traditional
hostname pgw02
log syslog informational
service integrated-vtysh-config
!
ip route {{ hostvars[inventory_hostname]['ansible_default_ipv4']
['gateway'] }}/32 {{ ansible_default_ipv4['interface'] }}
!
interface {{ ansible_default_ipv4['interface'] }}
 ip address ospf router-id {{hostvars[inventory_hostname]
['ansible_host']}}
 ip ospf authentication null
!
router ospf
 ospf router-id {{hostvars[inventory_hostname]['ansible_host']}}
 redistribute static
 network {{ hostvars[inventory_hostname]['ansible_default_ipv4']
['network'] }}/{{ mask_cidr }} area 0
 area 0 authentication message-digest
!
line vty
!
end
```

نموذج النشر:

1. **تعريف مرة واحدة**: إنشاء قالب FRR Jinja2 في دور Ansible الخاص بك (على سبيل المثال، `roles/frr/templates/frr.conf.j2`)
2. **تكوين المعلمات** Ansible الخاص بك لكل مضيف UPF: تعيين المتغيرات في جرد
3. **نشر في كل مكان** Ansible لنشر تكوين FRR إلى جميع عقد UPF: تشغيل كتاب تشغيل
4. **تخصيص تلقائي** Ansible المتغيرات الخاصة بالمضيف (عناوين IP، معرفات UPF لكل FRR لتخصيص تكوين (الموجه، إلخ: تستخدم

المعلمات القابلة للتخصيص في قالب Jinja2:

- **معلمات OSPF**: معرف الموجه، تكوين المنطقة، طرق المصادقة، إعلانات الشبكة
- **تكوين BGP**: ASN، علاقات الجيران، سياسات المسار، المجتمعات
- **إعادة توزيع المسارات**: أي المسارات لإعادة توزيعها (على سبيل المثال، `redistribute static` لمسارات UE)
- **تصفية المسارات**: خرائط المسار، قوائم البادئات، قوائم الوصول
- **إعدادات الواجهة**: معلمات واجهة OSPF/BGP

تكامل UPF: بمجرد نشر تكوين FRR الأساسي إلى كل مثيل UPF، يضيف UPF ديناميكيًا عناوين IP **مسارات مضيف** كـ (/32 لـ IPv4، /128 لـ IPv6) لمعدات المستخدم عبر واجهة FRR vtysh النشطة. ثم يتم PFCP بناءً على جلسات:

1. **إضافتها كمسارات ثابتة في FRR** بواسطة محرك مزامنة المسار UPF (عبر vtysh)
2. **التقاطها بواسطة FRR** عبر توجيه `redistribute static`
3. **الإعلان عنها لبروتوكولات التوجيه** (OSPF، BGP) وفقًا لتكوين FRR الخاص بك
4. **نشرها إلى الشبكة** بحيث يمكن توجيه حركة مرور UE إلى هذا المثيل من UPF

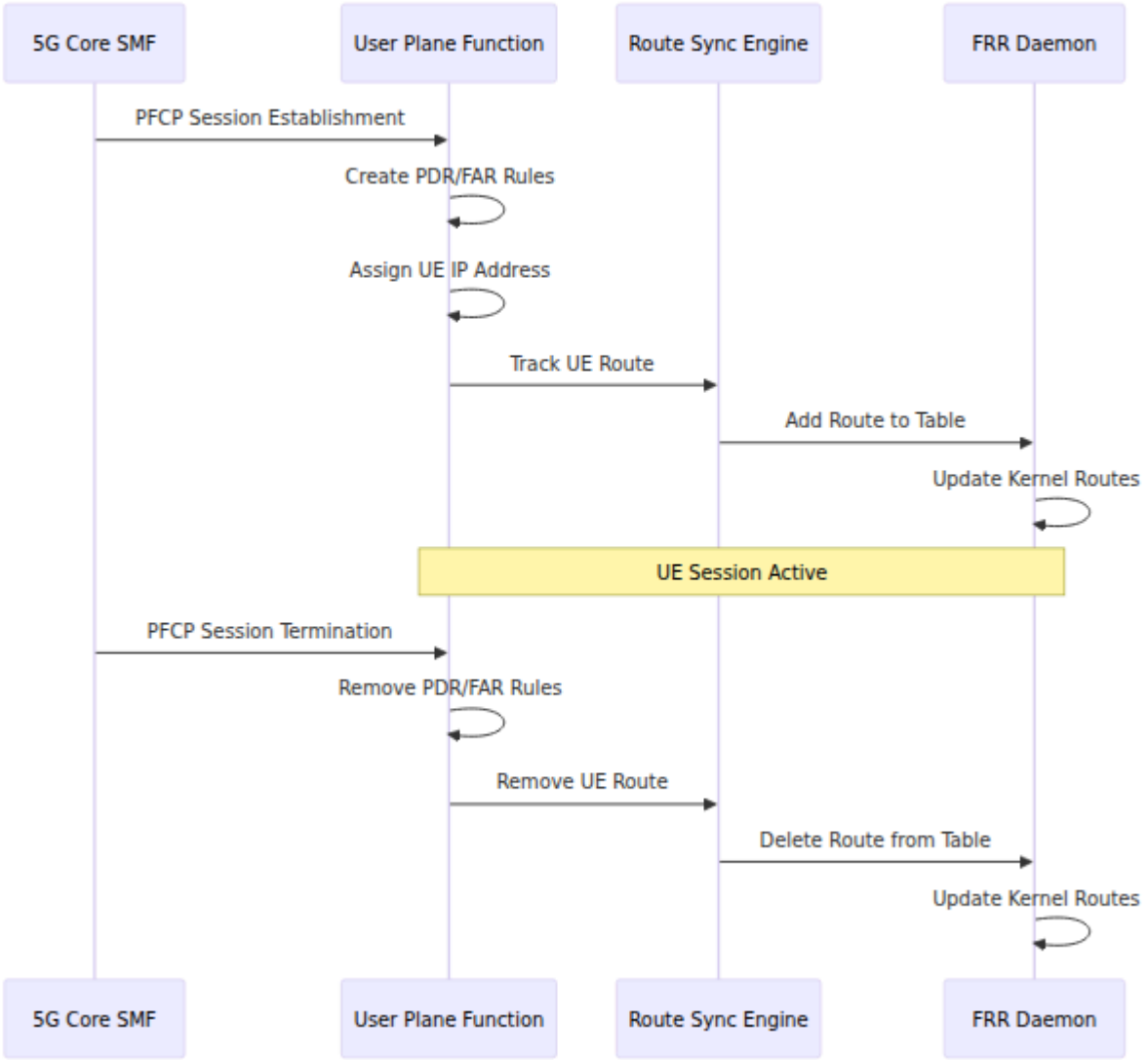
> **مهم**: يضيف UPF المسارات من خلال واجهة vtysh الخاصة بـ FRR، مما يجعلها مسارات ثا��تة في FRR (وليس مسارات نواة). يجب عليك استخدام `redistribute static` في تكوين OSPF/BGP الخاص بك، وليس `redistribute kernel`.

نقاط رئيسية:

- **تحديد مرة واحدة، نشر في كل مكان**: قم بتعريف قالب FRR Jinja2 مرة واحدة في Ansible، وسيتم نشره تلقائيًا إلى جميع مثيلات UPF
- **تتعامل Ansible مع التكوين الثابت**: يقوم قالب Jinja2 بإعداد جميع معلمات بروتوكول التوجيه (OSPF، جيران BGP، مناطق المصادقة، سياسات المسار، إلخ)

- **يتعامل UPF مع المسارات الديناميكية**: يدير كل مثيل UPF ديناميكيًا فقط مسارات UE IP /32 بناءً على جلسات PFCP النشطة
- **الإعلان التلقائي عن المسارات**: يقوم FRR على كل UPF بإعادة توزيع المسارات المحلية لمعدات المستخدم تلقائيًا وفقًا لسياساتك المكونة
- **إدارة مركزية**: تحديث قالب Ansible وإعادة تشغيل كتاب التشغيل لتغيير تكوين التوجيه عبر جميع UPFs في وقت واحد

الإعلان عن المسارات



المراقبة والإدارة

تكامل واجهة المستخدم على الويب

يوفر لوحة تحكم UPF التي تعرض صفحة **المسارات**:

- **حالة المسار**: ما إذا كانت مزامنة المسار مفعلة أو معطلة
- **إجمالي المسارات**: عدد عناوين IP لمعدات المستخدم التي يتم تتبعها
- **إحصائيات المزامنة**: عدد المسارات التي تمت مزامنتها بنجاح وأي فشل
- **المسارات النشطة**: قائمة في الوقت الحقيقي بجميع عناوين IP لمعدات المستخدم الموجودة حاليًا في جدول التوجيه
- **جيران OSPF**: حالة حية للجوار OSPF مع تفاصيل الجيران
- **أقران BGP**: حالة جلسة BGP (عند التكوين) وإحصائيات البادئات
- **المسارات المعاد توزيعها من OSPF**: عرض كامل لـ LSAs الخارجية التي توضح كيف يتم الإعلان عن مسارات UE

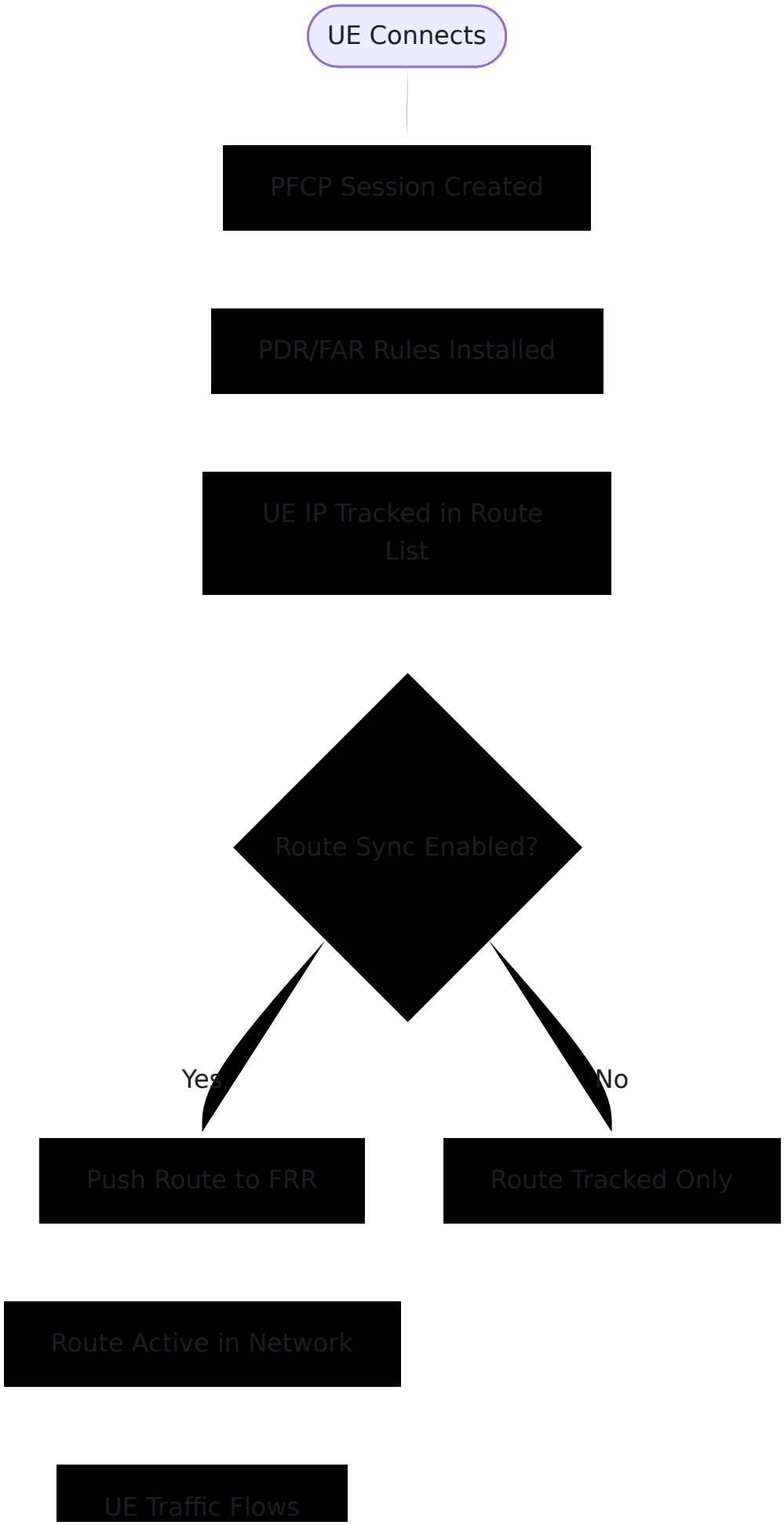
توفر صفحة المسارات رؤية شاملة لمزامنة مسارات UE، وإعلانات بروتوكول التوجيه، وجيران المسارات المعاد توزيعها.

عملية المزامنة اليدوية

يمكن للمسؤولين تشغيل مزامنة المسار اليدوية من خلال واجهة المستخدم على الويب باستخدام زر **مزامنة المسارات**. تقوم هذه العملية بـ:

1. إعادة قراءة القائمة الحالية لجلسات UE النشطة من UPF
2. المقارنة مع جدول توجيه FRR
3. إضافة أي مسارات مفقودة
4. إزالة أي مسارات قديمة
5. إرجاع إحصائيات المزامنة المحدثة

تدفق المسار


UE Connects
PFCP Session Created
PDR/FAR Rules Installed
UE IP Tracked in Route List
Route Sync Enabled?
Yes
No
Push Route to FRR
Route Tracked Only
Route Active in Network
UE Traffic Flows

الفوائد

- **توفير بدون لمس**: يتم إدارة المسارات تلقائيًا دون تدخل يدوي
- **التكيف الديناميكي**: يتكيف توجيه الشبكة في الوقت الحقيقي مع تنقل UE وتغييرات الجلسة
- **قابلية التوسع**: يدعم آلاف المسارات المتزامنة لمعدات المستخدم
- **المرونة**: يتم تتبع عمليات المزامنة الفاشلة ويمكن إعادة المحاولة
- **الرؤية**: رؤية كاملة لحالة المسار من خلال واجهة المستخدم على الويب

التفاصيل الفنية

نقاط نهاية API

يعرض UPF نقاط نهاية إدارة المسارات التالية:

- `GET /api/v1/routes` - قائمة بجميع مسارات UE المتعقبة دون المزامنة

- `POST /api/v1/routes/sync` - مزامنة المسارات مع FRR وإرجاع القائمة المحدثة
- `GET /api/v1/route_stats` - الحصول على إحصائيات التوجيه التفصيلية
- `GET /api/v1/routing/sessions` - الحصول على جلسات بروتوكول التوجيه (أقران BGP، جيران OSPF)
- `GET /api/v1/ospf/database/external` - الحصول على قاعد�ة بيانات OSPF AS-External LSA (المسارات المعاد توزيعها)

> **انظر أيضًا**: وثائق إدارة المسارات للحصول على تفاصيل كاملة عن النقاط النهائية API - والأمثلة

تنسيق المسار

يتم تخزين وإدارة المسارات كعناوين IP بسيطة (على سبيل المثال، `100.64.18.5`). يتعامل خادم التوجيه مع تفاصيل إدخال المسار الكامل بما في ذلك:

- بادئة الوجهة/القناع
- البوابة/الخطوة التالية
- ربط الواجهة
- المقياس والمسافة الإدارية

دعم IPv6

يدعم مدير المسار كل من عناوين UE IPv4 وIPv6:

نوع العنوان	طول البادئة	مثال
IPv4	/32	`100.64.18.5/32`
IPv6	/128	`2001:db8::1/128`

بالنسبة لـ IPv6، تأكد من أن تكوين FRR الخاص بك يتضمن إعادة توزيع BGP IPv6 أو OSPFv3 المناسبة:

```
router ospf6
 redistribute static
```

أو بالنسبة لـ BGP:

```
router bgp <asn>
 address-family ipv6 unicast
  redistribute static
```

تحقق من FRR

قاعدة بيانات OSPF LSA الخارجية

يتم إعادة توزيعها بشكل صحيح في UE يمكنك التحقق من أن مسارات OSPF من خلال فحص قاعدة بيانات حالة رابط OSPF الخاصة بـ FRR. تُظهر LSAs المسارات التي تم حقنها في OSPF من مصادر خارجية.

قاعدة بيانات OSPF لـ FRR تظهر LSAs الخارجية بما في ذلك مسار UE 100.64.18.5/32 الذي يتم الإعلان عنه كمسار E2 (2 النوع الخارجي).

في المثال أعلاه، يمكنك رؤية:

- **Network LSA (10.98.0.20)**: إعلان الشبكة الخاص بـ UPF
- **Router LSA (192.168.1.1)**: إعلان موجه OSPF
- **LSAs الخارجية**: بما في ذلك مسار UE `100.64.18.5` المعاد توزيعه في OSPF مع نوع المقياس E2 (2 النوع الخارجي)

يؤكد هذا التحقق أن:

1. UPF يتتبع بنجاح عنوان IP لمعدات المستخدم

2. لقد دفع محرك مزامنة المسار المسار إلى FRR
3. لقد أعاد FRR توزيع المسار في OSPF
4. يتلقى جيران OSPF إعلانات المسار