Guía de Operaciones de OmniUPF

Tabla de Contenidos

1. Descripción General
2. Entendiendo la Arquitectura del Plano de Usuario 5G
3. Componentes del UPF
4. Protocolo PFCP e Integración con SMF
5. Operaciones Comunes
6. Resolución de Problemas
7. Documentación Adicional
8. Glosario

Descripción General

OmniUPF (Función de Plano de Usuario basada en eBPF) es una Función de Plano de Usuario 5G/LTE de alto rendimiento que proporciona reenvío de paquetes de calidad de operador, aplicación de QoS y gestión de tráfico para redes móviles. Construido sobre la tecnología eBPF de Linux (Filtro de Paquetes de Berkeley extendido) y mejorado con capacidades de gestión integrales, OmniUPF ofrece la infraestructura central de procesamiento de paquetes requerida para redes 5G SA, 5G NSA y LTE.

¿Qué es una Función de Plano de Usuario?

La Función de Plano de Usuario (UPF) es el elemento de red estandarizado por 3GPP responsable del procesamiento y reenvío de paquetes en redes 5G y LTE. Proporciona:

- **Reenvío de paquetes de alta velocidad** entre dispositivos móviles y redes de datos
- **Aplicación de Calidad de Servicio (QoS)** para diferentes tipos de tráfico
- **Detección y enrutamiento de tráfico** basado en filtros y reglas de paquetes
- **Informe de uso** para facturación y análisis
- **Almacenamiento en búfer de paquetes** para escenarios de movilidad y gestión de sesiones
- **Soporte para interceptación legal** para cumplimiento normativo

OmniUPF implementa la funcionalidad completa de UPF definida en 3GPP TS 23.501 (5G) y TS 23.401 (LTE), proporcionando una solución de plano de usuario completa y lista para producción utilizando la tecnología eBPF del núcleo de Linux para un rendimiento máximo.

Capacidades Clave de OmniUPF

Procesamiento de Paquetes:

- Procesamiento de paquetes de plano de usuario completamente compatible con 3GPP
- Ruta de datos basada en eBPF para rendimiento a nivel de núcleo
- Encapsulación y desencapsulación de GTP-U (Protocolo de Túnel GPRS)
- Soporte para IPv4 e IPv6 tanto para redes de acceso como de datos
- XDP (Ruta de Datos eXpress) para procesamiento de latencia ultra-baja
- Procesamiento de paquetes multihilo

QoS y Gestión de Tráfico:

- Reglas de Aplicación de QoS (QER) para gestión de ancho de banda
- Reglas de Detección de Paquetes (PDR) para clasificación de tráfico
- Reglas de Acción de Reenvío (FAR) para decisiones de enrutamiento
- Filtrado de Flujo de Datos de Servicio (SDF) para enrutamiento específico de aplicaciones
- Reglas de Informe de Uso (URR) para seguimiento de volumen y facturación

Control y Gestión:

- Interfaz PFCP (Protocolo de Control de Reenvío de Paquetes) a SMF/PGW-C
- API RESTful para monitoreo y diagnóstico
- Estadísticas y métricas en tiempo real
- Monitoreo de capacidad de mapas eBPF
- Panel de control basado en web

Características de Rendimiento:

- Procesamiento de paquetes sin copia a través de eBPF
- Reenvío de paquetes a nivel de núcleo (sin sobrecarga de espacio de usuario)
- Escalabilidad multinúcleo
- Capaz de descarga para aceleración de hardware
- Optimizado para implementaciones nativas en la nube

Para detalles sobre el uso del panel de control, consulte Operaciones de la Interfaz Web.

Entendiendo la Arquitectura del Plano de Usuario

OmniUPF es una solución de plano de usuario unificada que proporciona reenvío de paquetes de calidad de operador para redes 5G Autónomas (SA), 5G NSA y 4G LTE/EPC. **OmniUPF es un solo producto** que puede funcionar simultáneamente como:

- **UPF (Función de Plano de Usuario)** - plano de usuario 5G/NSA (controlado por OmniSMF a través de N4/PFCP)
- **PGW-U (Puerta de Enlace de PDN de Plano de Usuario)** - puerta de enlace EPC 4G a redes externas (controlado por OmniPGW-C a través de Sxc/PFCP)
- **SGW-U (Puerta de Enlace de Servicio de Plano de Usuario)** - puerta de enlace de servicio EPC 4G (controlado por OmniSGW-C a través de Sxb/PFCP)

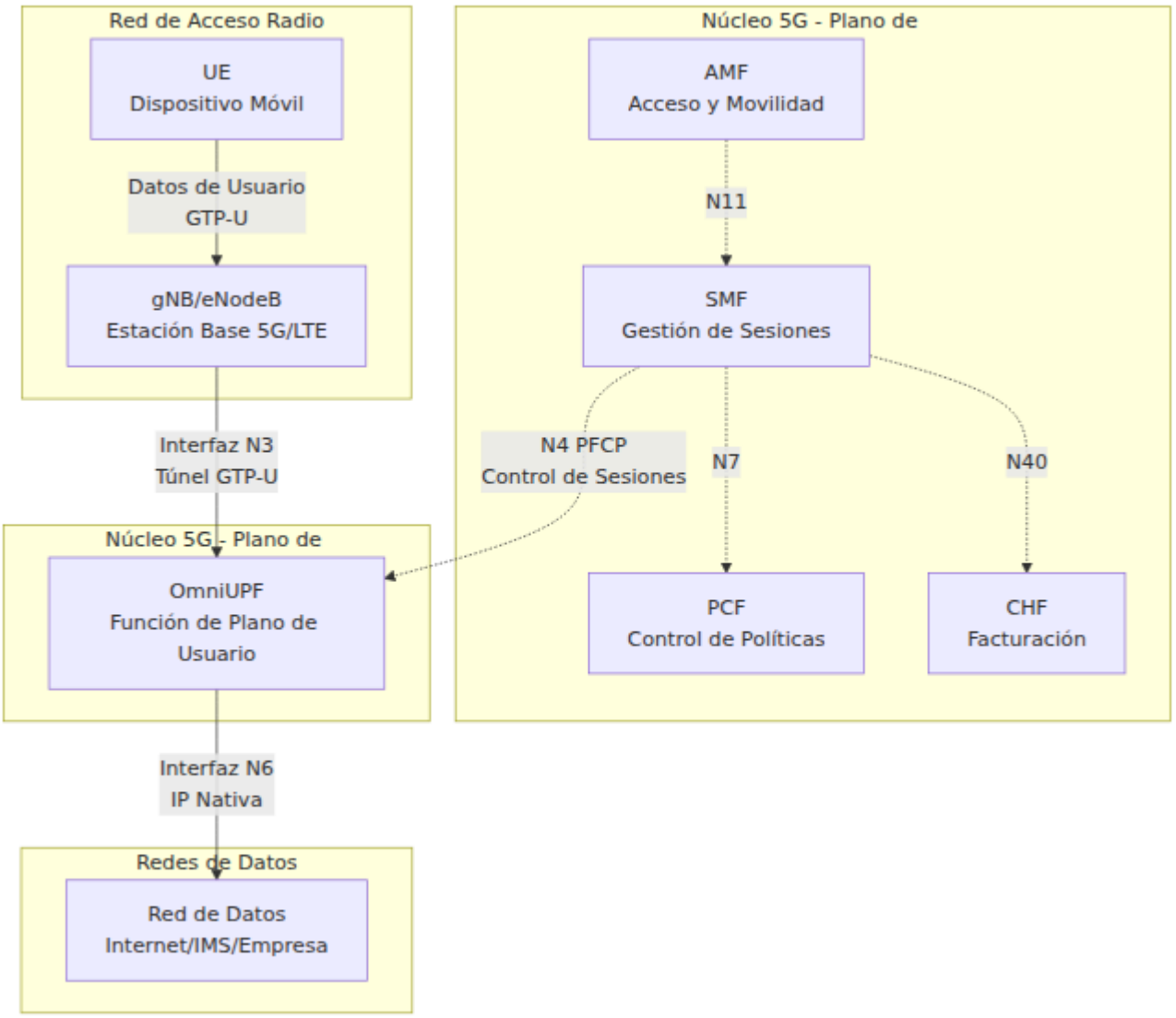
OmniUPF puede operar en **cualquier combinación** de estos modos:

- **Solo UPF**: Implementación pura de 5G
- **PGW-U + SGW-U**: Puerta de enlace 4G combinada (implementación típica de EPC)
- **UPF + PGW-U + SGW-U**: Soporte simultáneo para 4G y 5G (escenario de migración)

Todos los modos utilizan el mismo motor de procesamiento de paquetes basado en eBPF y el protocolo PFCP, proporcionando un alto rendimiento constante, ya sea operando como UPF, PGW-U, SGW-U o los tres simultáneamente.

Arquitectura de Red 5G (Modo SA)

La solución OmniUPF se sitúa en el plano de datos de las redes 5G, proporcionando la capa de reenvío de paquetes de alta velocidad que conecta dispositivos móviles a redes y servicios de datos.



Arquitectura de Red 4G LTE/EPC

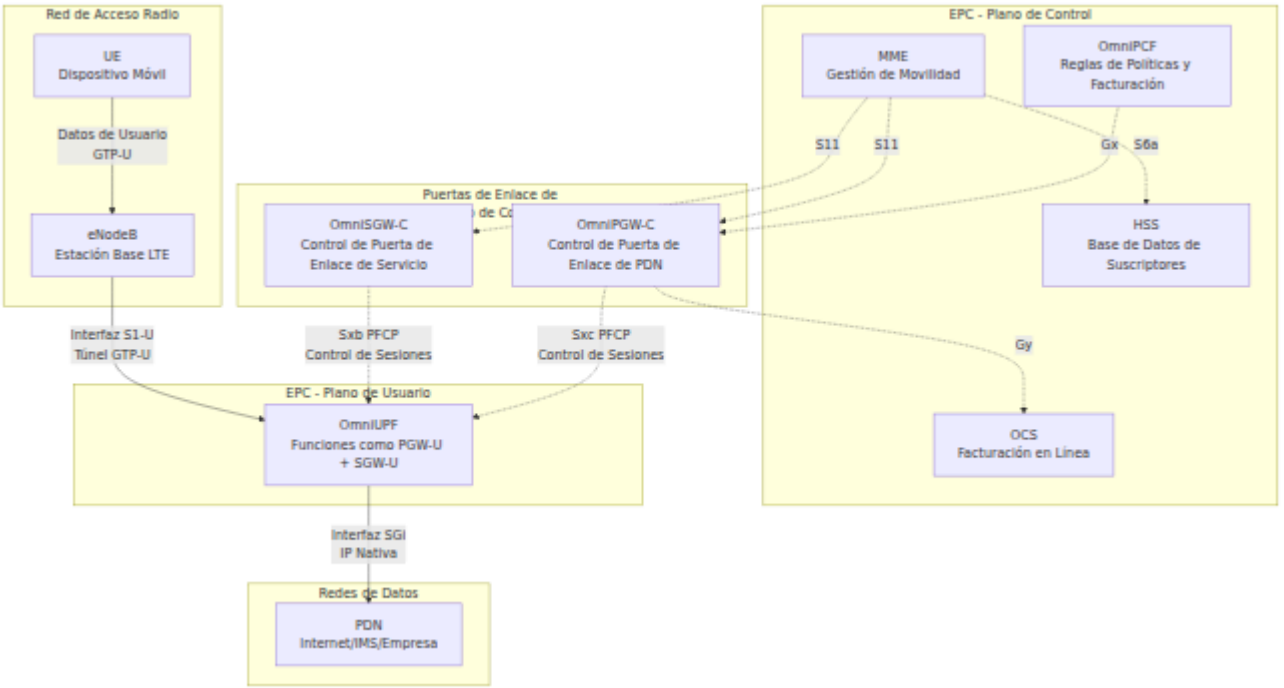
OmniUPF también soporta implementaciones de 4G LTE y EPC (Núcleo de Paquetes Evolucionado), funcionando como OmniPGW-U o OmniSGW-U dependiendo de la arquitectura de la red.

Modo Combinado PGW-U/SGW-U (Implementación Típica de 4G)

En este modo, OmniUPF actúa como SGW-U y PGW-U, controlado por funciones de plano de control separadas.

Modo Separado SGW-U y PGW-U (Roaming/Multi-Sitio)

En implementaciones de roaming o multi-sitio, se pueden desplegar dos instancias separadas de OmniUPF: una como SGW-U y otra como PGW-U.



Modo de Bucle N9 (Instancia Única SGWU+PGWU)

Para implementaciones simplificadas, OmniUPF puede ejecutar **tanto los roles de SGWU como de PGWU en una sola instancia** con procesamiento de

bucle N9 completamente en eBPF.

Características Clave:

- **Latencia N9 sub-microsegundo** - Procesado completamente en eBPF, nunca toca la red
- **Reducción del 40-50% en CPU** - Un solo pase XDP frente a dos instancias separadas
- **Implementación simplificada** - Una instancia, un archivo de configuración
- **Detección automática** - Cuando `n3_address` = `n9_address`, el bucle se habilita
- **Cumplimiento total de 3GPP** - Protocolos estándar PFCP y GTP-U

Configuración:

```
# /etc/omniupf/runtime.exs
xdp_interfaces = "eth0"
n3_address = "10.0.1.10"        # IP de la interfaz S1-U
n9_address = n3_address          # La misma IP habilita el bucle
N9
pfcp_address = "10.0.1.10"      # Tanto SGWU-C como PGWU-C se
conectan aquí
pfcp_port = 8805
```

Cuándo usar:

- Implementaciones de computación en el borde (minimizar latencia)
- Entornos con restricciones de costos (servidor único)
- Laboratorio/pruebas (configuración simplificada)
- Implementaciones pequeñas a medianas (< 100K suscriptores)

Cuándo NO usar:

- Se requiere redundancia geográfica (SGWU y PGWU en diferentes ubicaciones)
- Mandatos regulatorios para puertas de enlace separadas
- Escala masiva (> 1M suscriptores)

Para detalles completos, ejemplos de configuración, resolución de problemas y métricas de rendimiento, consulte **Guía de Operaciones de Bucle N9**.

Cómo Funcionan las Funciones de Plano de Usuario en la Red

La función de plano de usuario (OmniUPF, OmniPGW-U o OmniSGW-U) opera como el plano de reenvío controlado por el respectivo plano de control:

1. **Establecimiento de Sesión**
    - **5G**: OmniSMF establece la asociación PFCP a través de la interfaz N4 con OmniUPF

- **4G**: OmniPGW-C o OmniSGW-C establece la asociación PFCP a través de Sxb/Sxc con OmniPGW-U/OmniSGW-U
- El plano de control crea sesiones PFCP para cada sesión PDU de UE (5G) o contexto PDP (4G)
- El plano de usuario recibe reglas PDR, FAR, QER y URR a través de PFCP
- Los mapas eBPF se poblan con reglas de reenvío

2. **Procesamiento de Paquetes de Subida** (UE → Red de Datos)

- **5G**: Los paquetes llegan a la interfaz N3 desde gNB con encapsulación GTP-U
- **4G**: Los paquetes llegan a la interfaz S1-U (SGW-U) o S5/S8 (PGW-U) desde eNodeB con encapsulación GTP-U
- El plano de usuario compara los paquetes con PDRs de subida basándose en TEID
- El programa eBPF aplica QER (limitación de tasa, marcado)
- FAR determina la acción de reenvío (reenviar, descartar, almacenar en búfer, duplicar)
- Se elimina el túnel GTP-U, los paquetes se reenvían a la interfaz N6 (5G) o SGi (4G)
- URR rastrea el conteo de paquetes y bytes para la facturación

3. **Procesamiento de Paquetes de Bajada** (Red de Datos → UE)

- **5G**: Los paquetes llegan a la interfaz N6 como IP nativa
- **4G**: Los paquetes llegan a la interfaz SGi como IP nativa
- El plano de usuario compara los paquetes con PDRs de bajada basándose en la dirección IP de UE
- Los filtros SDF pueden clasificar aún más el tráfico por puerto, protocolo o aplicación
- FAR determina el túnel GTP-U y los parámetros de reenvío
- Se añade la encapsulación GTP-U con el TEID apropiado
- **5G**: Los paquetes se reenvían a la interfaz N3 hacia gNB
- **4G**: Los paquetes se reenvían a S1-U (SGW-U) o S5/S8 (PGW-U) hacia eNodeB

4. **Movilidad y Transferencia**

- **5G**: OmniSMF actualiza las reglas PDR/FAR durante escenarios de transferencia
- **4G**: OmniSGW-C/OmniPGW-C actualiza las reglas durante la transferencia inter-eNodeB o TAU (Actualización de Área de Seguimiento)
- El plano de usuario puede almacenar paquetes durante el cambio de ruta
- Transición sin problemas entre estaciones base sin pérdida de paquetes

Integración con el Plano de Control (4G y 5G)

OmniUPF se integra con funciones de plano de control tanto de 5G como de 4G a través de interfaces estándar de 3GPP:

Interfaces 5G

Interfaz	De → A	Propósito	Especificación 3GPP
N4	OmniSMF ↔ OmniUPF	Establecimiento, modificación, eliminación de sesión PFCP	TS 29.244
N3	gNB → OmniUPF	Tráfico de plano de usuario desde RAN (GTP-U)	TS 29.281
N6	OmniUPF → Red de Datos	Tráfico de plano de usuario hacia DN (IP nativa)	TS 23.501
N9	OmniUPF ↔ OmniUPF	Comunicación inter-UPF para roaming/borde	TS 23.501

Interfaces 4G/EPC

Interfaz	De → A	Propósito	Especificación 3GPP
Sxb	OmniSGW-C ↔ OmniUPF (modo SGW-U)	Control de sesión PFCP para puerta de enlace de servicio	TS 29.244
Sxc	OmniPGW-C ↔ OmniUPF (modo PGW-U)	Control de sesión PFCP para puerta de enlace de PDN	TS 29.244
S1-U	eNodeB → OmniUPF (modo SGW-U)	Tráfico de plano de usuario desde RAN (GTP-U)	TS 29.281
S5/S8	OmniUPF (SGW-U) ↔ OmniUPF (PGW-U)	Plano de usuario interpuerta (GTP-U)	TS 29.281
SGi	OmniUPF (modo PGW-U) → PDN	Tráfico de plano de usuario hacia la red de datos (IP nativa)	TS 23.401

Nota: Todas las interfaces PFCP (N4, Sxb, Sxc) utilizan el mismo protocolo PFCP definido en TS 29.244. Los nombres de las interfaces difieren, pero el protocolo y los formatos de mensaje son idénticos.

Componentes del UPF

Ruta de Datos eBPF

La **ruta de datos eBPF** es el motor central de procesamiento de paquetes que se ejecuta en el núcleo de Linux para un rendimiento máximo.

Funciones Principales:

- **Procesamiento de GTP-U**: Encapsulación y desencapsulación de túneles GTP-U
- **Clasificación de Paquetes**: Comparar paquetes con reglas PDR utilizando TEID, IP de UE o filtros SDF
- **Aplicación de QoS**: Aplicar limitación de tasa y marcado de paquetes según las reglas QER
- **Decisiones de Reenvío**: Ejecutar acciones FAR (reenviar, descartar, almacenar en búfer, duplicar, notificar)
- **Seguimiento de Uso**: Incrementar contadores URR para facturación basada en volumen

Mapas eBPF: La ruta de datos utiliza mapas eBPF (tablas hash en la memoria del núcleo) para el almacenamiento de reglas:

Nombre del Mapa	Propósito	Clave	Valor
`uplink_pdr_map`	PDRs de subida	TEID (32 bits)	Información PDR (ID FAR, ID QER, IDs URR)
`downlink_pdr_map`	PDRs de bajada (IPv4)	Dirección IP de UE	Información PDR
`downlink_pdr_map_ip6`	PDRs de bajada (IPv6)	Dirección IPv6 de UE	Información PDR
`far_map`	Reglas de reenvío	ID FAR	Parámetros de reenvío (acción, información del túnel)
`qer_map`	Reglas de QoS	ID QER	Parámetros de QoS (MBR, GBR, marcado)
`urr_map`	Seguimiento de uso	ID URR	Contadores de volumen (subida, bajada, total)
`sdf_filter_map`	Filtros SDF	ID PDR	Filtros de aplicación (puertos, protocolos)

Características de Rendimiento:

- **Sin copia**: Paquetes procesados completamente en espacio de núcleo
- **Soporte XDP**: Adjuntar a nivel de controlador de red para latencia sub-microsegundo
- **Multinúcleo**: Escala a través de núcleos de CPU con soporte de mapa por CPU

- **Capacidad**: Millones de PDRs/FARs en mapas eBPF (limitados por la memoria del núcleo)

Para el monitoreo de capacidad, consulte Gestión de Capacidad.

Manejador de Interfaz PFCP

La **interfaz PFCP** implementa 3GPP TS 29.244 para la comunicación con SMF o PGW-C.

Funciones Principales:

- **Gestión de Asociación**: Latido PFCP y configuración/liberación de asociación
- **Ciclo de Vida de Sesión**: Crear, modificar y eliminar sesiones PFCP
- **Instalación de Reglas**: Traducir IEs PFCP en entradas de mapa eBPF
- **Informe de Eventos**: Notificar a SMF sobre umbrales de uso, errores o eventos de sesión

Soporte de Mensajes PFCP:

Tipo de Mensaje	Dirección	Propósito
Establecimiento de Asociación	SMF → UPF	Establecer asociación de control PFCP
Liberación de Asociación	SMF → UPF	Destruir asociación PFCP
Latido	Bidireccional	Mantener viva la asociación
Establecimiento de Sesión	SMF → UPF	Crear nueva sesión PDU con PDR/FAR/QER/URR
Modificación de Sesión	SMF → UPF	Actualizar reglas para movilidad, cambios de QoS
Eliminación de Sesión	SMF → UPF	Eliminar sesión y todas las reglas asociadas
Informe de Sesión	UPF → SMF	Informar uso, errores o eventos

Elementos de Información (IE) Soportados:

- Crear PDR, FAR, QER, URR
- Actualizar PDR, FAR, QER, URR
- Eliminar PDR, FAR, QER, URR
- Información de Detección de Paquetes (IP de UE, F-TEID, filtro SDF)
- Parámetros de Reenvío (instancia de red, creación de encabezado externo)
- Parámetros de QoS (MBR, GBR, QFI)
- Disparadores de Informe de Uso (umbral de volumen, umbral de tiempo)

Servidor API REST

La **API REST** proporciona acceso programático al estado y operaciones del UPF.

Funciones Principales:

- **Monitoreo de Sesiones**: Consultar sesiones PFCP activas y asociaciones
- **Inspección de Reglas**: Ver configuraciones de PDR, FAR, QER, URR
- **Estadísticas**: Recuperar contadores de paquetes, estadísticas de rutas, estadísticas de XDP
- **Gestión de Buffers**: Ver y controlar buffers de paquetes
- **Información de Mapas**: Monitorear uso y capacidad de mapas eBPF

Puntos de Acceso de API: (34 puntos de acceso en total)

Categoría	Puntos de Acceso	Descripción
Salud	`/health`	Verificación de salud y estado
Configuración	`/config`	Configuración del UPF
Sesiones	`/pfcp_sessions`, `/pfcp_associations`	Datos de sesiones/asociaciones PFCP
PDRs	`/uplink_pdr_map`, `/downlink_pdr_map`, `/downlink_pdr_map_ip6`, `/uplink_pdr_map_ip6`	Reglas de detección de paquetes
FARs	`/far_map`	Reglas de acción de reenvío
QERs	`/qer_map`	Reglas de aplicación de QoS
URRs	`/urr_map`	Reglas de informe de uso
Buffers	`/buffer`	Estado y control del buffer de paquetes
Estadísticas	`/packet_stats`, `/route_stats`, `/xdp_stats`, `/n3n6_stats`	Métricas de rendimiento
Capacidad	`/map_info`	Capacidad y uso de mapas eBPF
Dataplane	`/dataplane_config`	Direcciones de interfaz N3/N9

Para detalles de la API y uso, consulte **Guía de Monitoreo**.

Panel de Control Web

El **Panel de Control Web** proporciona un tablero en tiempo real para el monitoreo y gestión del UPF.

Características:

- **Vista de Sesiones**: Navegar por sesiones PFCP activas con IP de UE, TEID y conteos de reglas
- **Gestión de Reglas**: Ver y gestionar PDRs, FARs, QERs y URRs a través de todas las sesiones
- **Monitoreo de Buffers**: Rastrear paquetes almacenados en búfer y controlar el almacenamiento en búfer por FAR
- **Tablero de Estadísticas**: Estadísticas en tiempo real de paquetes, rutas, XDP y estadísticas de interfaces N3/N6
- **Monitoreo de Capacidad**: Uso de mapas eBPF con indicadores de capacidad codificados por colores
- **Vista de Configuración**: Mostrar configuración del UPF y direcciones del dataplane
- **Visor de Registros**: Transmisión de registros en vivo para resolución de problemas

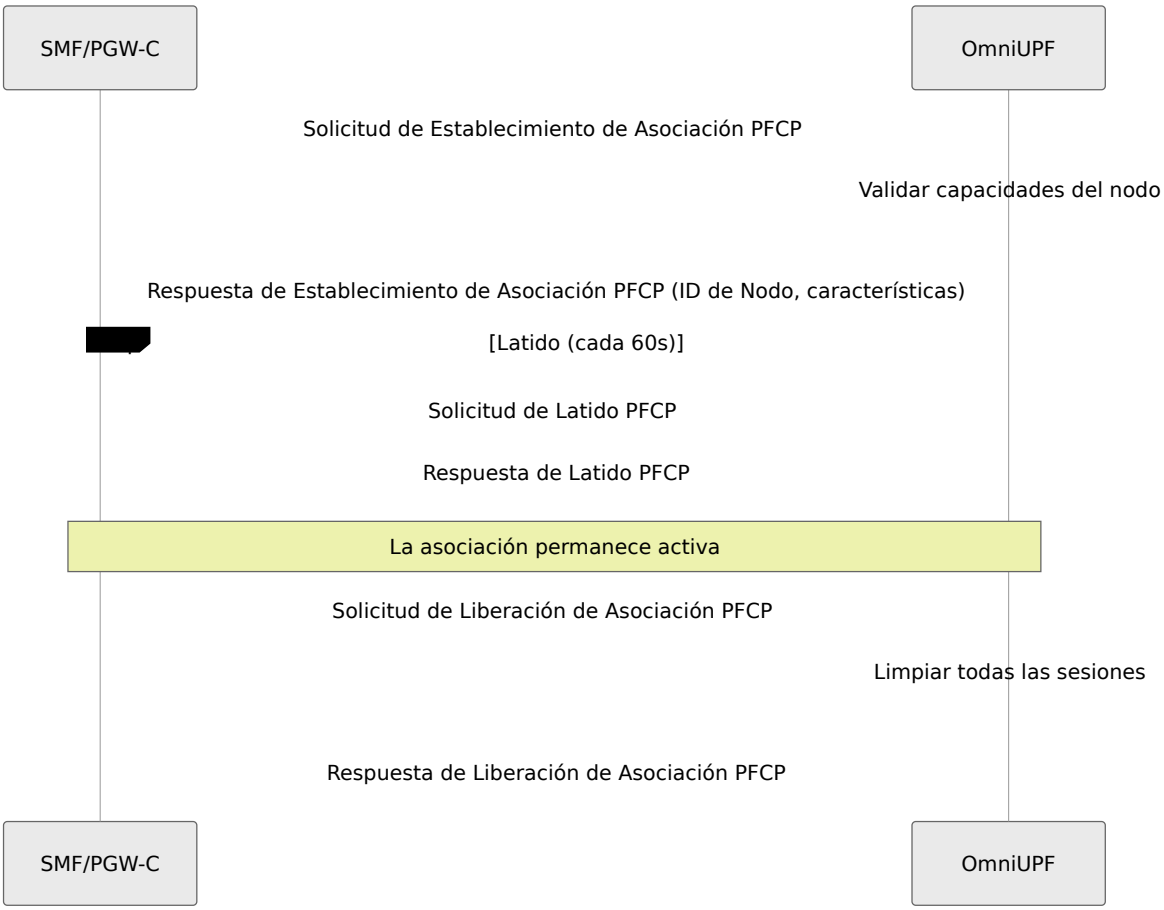
Para operaciones detalladas de la interfaz de usuario, consulte **Guía de Operaciones de la Interfaz Web**.

Protocolo PFCP e Integración con SMF

Asociación PFCP

Antes de que se puedan crear sesiones, el SMF debe establecer una asociación PFCP con el UPF.

Ciclo de Vida de la Asociación:



Puntos Clave:

- Cada SMF establece una asociación con el UPF
- UPF rastrea la asociación por ID de Nodo (FQDN o dirección IP)
- Los mensajes de latido mantienen la vivacidad de la asociación
- Todas las sesiones bajo una asociación se eliminan si se libera la asociación

Para ver asociaciones, consulte Vista de Sesiones.

Detección de Reinicio de SMF y Limpieza de Sesiones Huérfanas

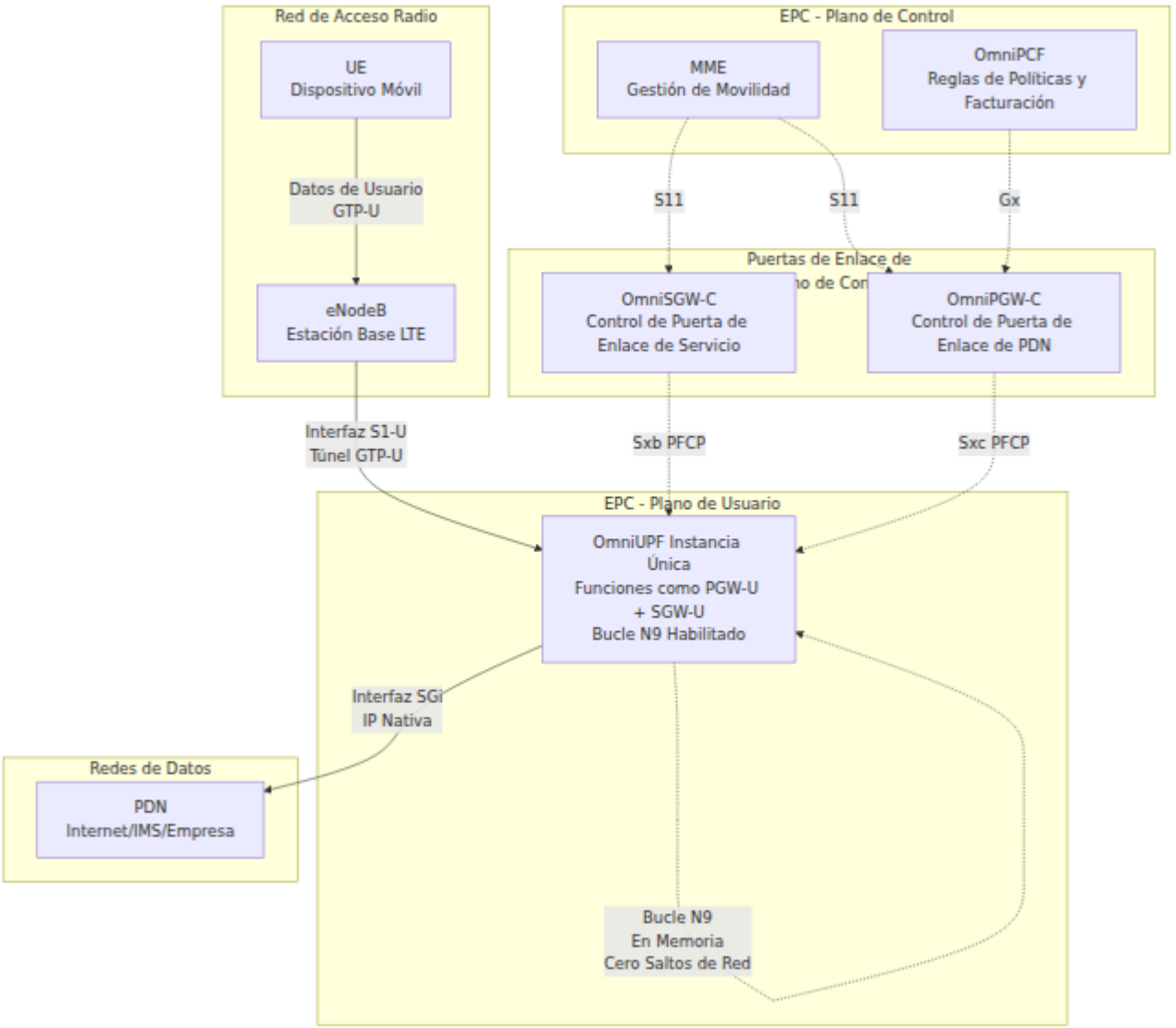
OmniUPF detecta automáticamente cuando un SMF se reinicia y limpia las sesiones huérfanas según las especificaciones de 3GPP TS 29.244.

Cómo Funciona:

Cuando un SMF establece una asociación PFCP, proporciona una **Marca de Tiempo de Recuperación** que indica cuándo se inició. OmniUPF almacena esta marca de tiempo para cada asociación. Si el SMF se reinicia:

1. El SMF pierde todo el estado de sesión en memoria
2. El SMF restablece la asociación PFCP con el UPF
3. El SMF envía **nueva Marca de Tiempo de Recuperación** (diferente de antes)
4. UPF detecta el cambio de marca de tiempo = SMF reiniciado
5. UPF elimina automáticamente **todas las sesiones huérfanas** de la antigua instancia de SMF
6. El SMF crea sesiones nuevas para suscriptores activos

Flujo de Detección de Reinicio:



Ejemplo de Registro:

Cuando un SMF se reinicia, verá:

```
WARN: La asociación con NodeID: smf-1 y dirección: 192.168.1.10 ya
existe
WARN: La Marca de Tiempo de Recuperación del SMF cambió (antigua:
2025-01-15T10:00:00Z, nueva: 2025-01-15T10:30:15Z) - SMF
reiniciado, eliminando 245 sesiones huérfanas
INFO: Eliminando sesión huérfana 2 (LocalSEID) debido al reinicio
del SMF
INFO: Eliminando sesión huérfana 3 (LocalSEID) debido al reinicio
del SMF
...
INFO: Eliminando sesión huérfana 246 (LocalSEID) debido al
reinicio del SMF
```

Notas Importantes:

1. **Aislamiento**: Solo se eliminan las sesiones del SMF reiniciado. Otras asociaciones de SMF y sus sesiones **no se ven afectadas**.

2. **Comparación de Marcas de Tiempo**: Si la Marca de Tiempo de Recuperación es **idéntica**, las sesiones se **mantienen** (el SMF se reconectó sin reiniciarse).

3. **Cumplimiento de 3GPP**: Este comportamiento está mandado por la Sección 5.22.2 de 3GPP TS 29.244:

   > "Si la Marca de Tiempo de Recuperación de la función CP ha cambiado desde el último Establecimiento de Asociación, la función UP deberá considerar que la función CP se ha reiniciado y deberá eliminar todas las sesiones PFCP asociadas con esa función CP."

Para la resolución de problemas de sesiones huérfanas, consulte Detección de Sesiones Huérfanas.

Manejo de Indicación de Error GTP-U

OmniUPF maneja mensajes de Indicación de Error GTP-U de pares descendentes (PGW-U, SGW-U, eNodeB, gNodeB) según las especificaciones de 3GPP TS 29.281.

¿Qué son las Indicaciones de Error?:

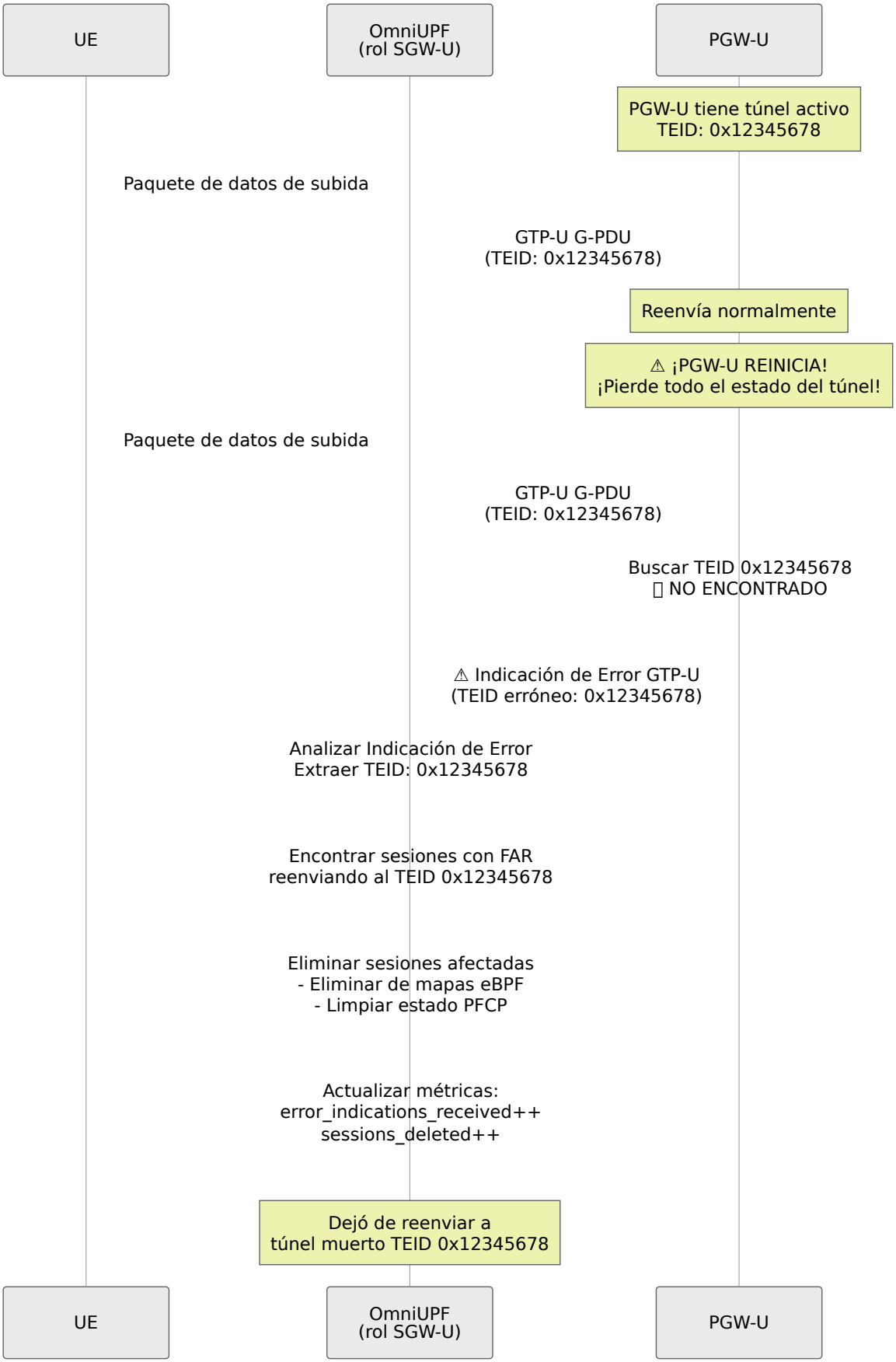
Cuando OmniUPF reenvía un paquete GTP-U a un par remoto (por ejemplo, PGW-U en una implementación SGW-U), el par puede enviar de vuelta una **Indicación de Error** si no reconoce el TEID (Identificador de Punto de Túnel). Esto indica:

- El par remoto se ha reiniciado y ha perdido el estado del túnel
- El túnel nunca se creó en el lado remoto (desajuste de configuración)
- El túnel ya se eliminó en el lado remoto

Cómo Funciona:

1. **UPF reenvía el paquete** → Envía un paquete GTP-U con TEID X al par remoto (puerto 2152)
2. **El par remoto no reconoce el TEID X** → Busca el TEID en su tabla de túneles, no encontrado
3. **El par remoto envía Indicación de Error** → Mensaje GTP-U tipo 26 con IE que contiene el TEID erróneo
4. **UPF recibe la Indicación de Error** → Analiza el mensaje para extraer el TEID X
5. **UPF encuentra sesiones afectadas** → Busca todas las sesiones para FARs que reenvían al TEID X
6. **UPF elimina sesiones** → Elimina sesiones de los mapas eBPF y del estado PFCP
7. **UPF actualiza métricas** → Incrementa contadores de Prometheus para monitoreo

Flujo de Indicación de Error:



Formato de Paquete (Sección 7.3.1 de 3GPP TS 29.281):

```
Indicación de Error GTP-U:
┌──────────────────────────────────────────────┐
│ Encabezado GTP-U (12 bytes)                  │
├──────────────────────────────────────────────┤
│ Versión, PT, Banderas       │ 0x32           │
│ Tipo de Mensaje             │ 26 (0x1A)      │
│ Longitud                    │ 9 bytes        │
│ TEID                        │ 0 (siempre)    │
│ Número de Secuencia         │ varía          │
│ Número de N-PDU             │ 0              │
│ Siguiente Encabezado de Extensión │ 0          │
├──────────────────────────────────────────────┤
│ IE: Datos de TEID I (5 bytes)                │
├──────────────────────────────────────────────┤
│ Tipo                        │ 16 (0x10)      │
│ TEID erróneo                │ 4 bytes        │
└──────────────────────────────────────────────┘
```

Cuándo Esto Importa:

Escenario 1: Reinicio de PGW-U en Arquitectura GTP S5/S8

- SGW-U (OmniUPF) reenvía tráfico S5/S8 a PGW-U
- PGW-U se reinicia y pierde todo el estado del túnel S5/S8
- SGW-U continúa reenviando a antiguos TEIDs
- PGW-U envía Indicaciones de Error
- SGW-U **deja de usar túneles muertos automáticamente**

Escenario 2: Reinicio de UPF Par en Arquitectura N9

- UPF-1 (OmniUPF) reenvía tráfico N9 a UPF-2
- UPF-2 se reinicia
- UPF-1 recibe Indicaciones de Error
- UPF-1 limpia sesiones

Ejemplo de Registro:

Al recibir una Indicación de Error:

```
WARN: Recibida Indicación de Error GTP-U de 192.168.50.10:2152
para TEID 0x12345678 - el par remoto no reconoce este TEID
WARN: Se encontró la sesión LocalSEID=42 con FAR GlobalId=1
reenviando al TEID erróneo 0x12345678 del par 192.168.50.10
INFO: Eliminando la sesión LocalSEID=42 debido a la Indicación de
Error GTP-U para TEID 0x12345678 de 192.168.50.10
WARN: Eliminadas 1 sesión(es) debido a la Indicación de Error GTP-
U para TEID 0x12345678 del par 192.168.50.10
```

Métricas de Prometheus:

Monitorear la actividad de Indicación de Error con granularidad por par y por nodo:

```
# Total de Indicaciones de Error recibidas de pares
upf_buffer_listener_error_indications_received_total{node_id="pgw-u-
1",peer_address="192.168.50.10"}

# Sesiones eliminadas debido a Indicaciones de Error
upf_buffer_listener_error_indication_sessions_deleted_total{node_id="
u-1",peer_address="192.168.50.10"}

# Indicaciones de Error enviadas (para TEIDs entrantes desconocidos)
upf_buffer_listener_error_indications_sent_total{node_id="enodeb-
1",peer_address="10.60.0.1"}
```

Etiquetas de Métrica:

- `node_id`: ID de Nodo PFCP de la asociación (o "desconocido" si no existe asociación)
- `peer_address`: Dirección IP del par remoto

Estas métricas ayudan a identificar pares problemáticos y rastrear patrones de Indicación de Error por nodo de plano de control.

Notas Importantes:

1. **Limpieza Automática**: No se necesita intervención del operador: las sesiones se eliminan automáticamente

2. **Coincidencia de TEID**: Solo se eliminan las sesiones con FARs que reenvían al TEID erróneo exacto

3. **Aislamiento por Par**: Las Indicaciones de Error de un par solo afectan a las sesiones que reenvían a ese par

4. **Múltiples Sesiones**: Si múltiples sesiones reenvían al mismo TEID muerto, **todas se eliminan**

5. **Complementario a Marca de Tiempo de Recuperación**:

    - La detección de Marca de Tiempo de Recuperación = proactiva (detecta reinicio durante el establecimiento de asociación)
    - El manejo de Indicación de Error = reactivo (detecta túneles muertos cuando fluyen datos)

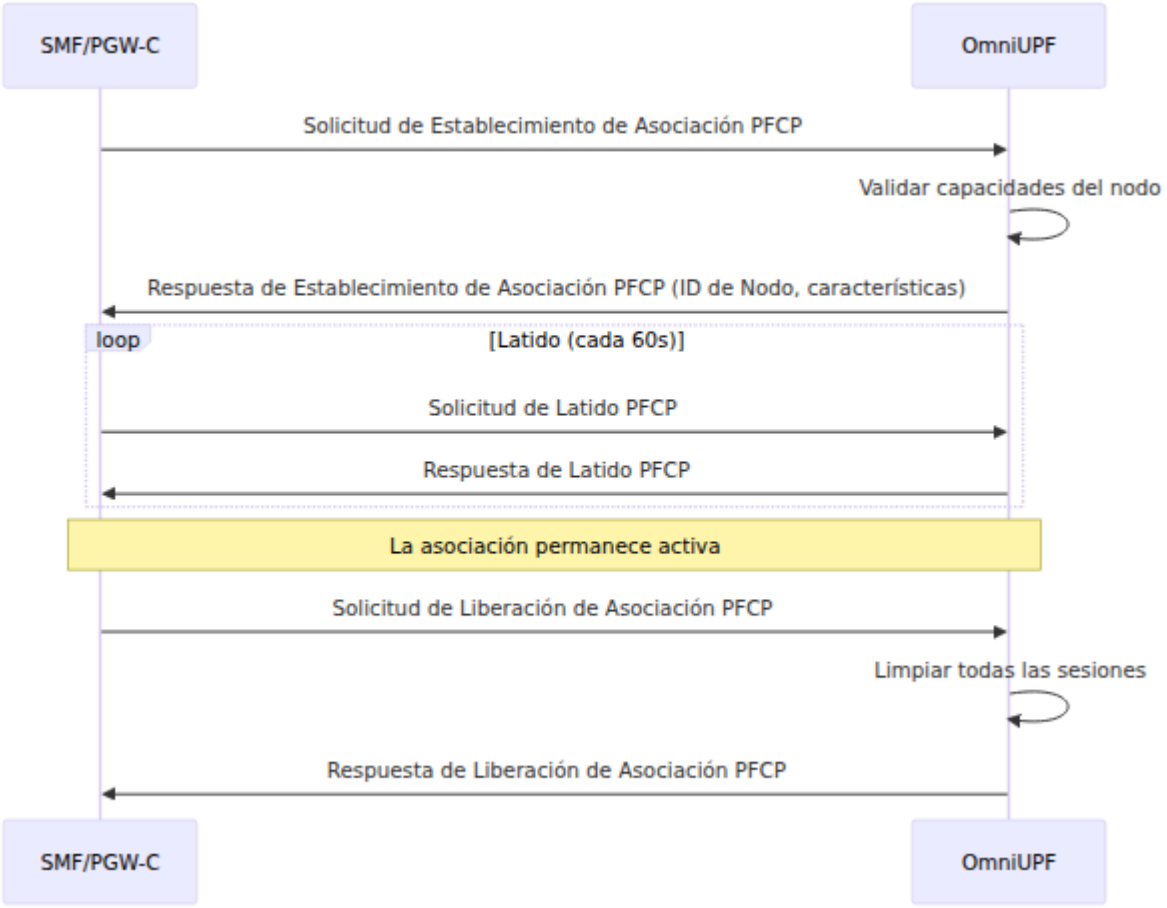
6. **Manejo de Paquete Malformado**: Las Indicaciones de Error inválidas se registran e ignoran (no se eliminan sesiones)

Para la resolución de problemas de Indicaciones de Error, consulte Depuración de Indicaciones de Error GTP-U.

Creación de Sesión PFCP

Cuando un UE establece una sesión PDU (5G) o contexto PDP (LTE), el SMF crea una sesión PFCP en el UPF.

Flujo de Establecimiento de Sesión:



Contenido Típico de la Sesión:

- **PDR de Subida**: Coincide con TEID N3, reenvía a través de FAR a N6
- **PDR de Bajada**: Coincide con dirección IP de UE, reenvía a través de FAR a N3 con encapsulación GTP-U
- **FAR**: Parámetros de reenvío (creación de encabezado externo, instancia de red)
- **QER**: Límites de QoS (MBR, GBR) y marcado de paquetes (QFI)
- **URR**: Informe de volumen para facturación (opcional)

Modificación de Sesión PFCP

El SMF puede modificar sesiones para eventos de movilidad (transferencia), cambios de QoS o actualizaciones de servicio.

Escenarios Comunes de Modificación:

1. **Transferencia (basada en N2)**

- Actualizar FAR de subida con nuevo punto final de túnel gNB (F-TEID)
- Opcionalmente almacenar paquetes durante el cambio de ruta
- Vaciar el búfer a la nueva ruta cuando esté listo

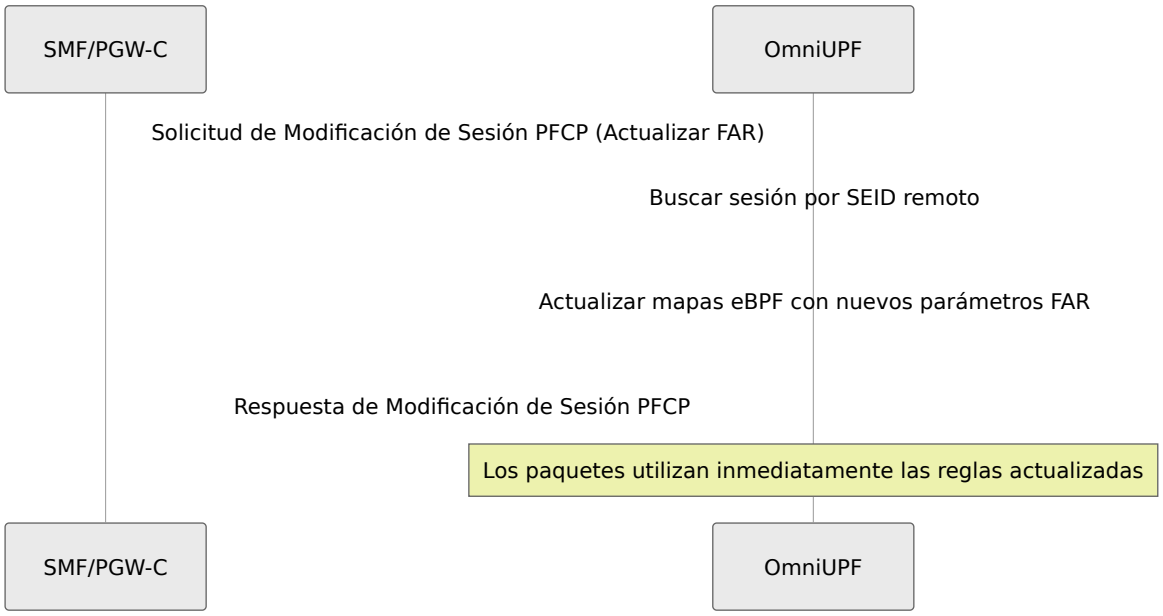
2. **Cambio de QoS**

   - Actualizar QER con nuevos valores MBR/GBR
   - Puede agregar/eliminar filtros SDF en PDR para QoS específico de aplicación

3. **Actualización de Servicio**

   - Agregar nuevos PDRs para flujos de tráfico adicionales
   - Modificar FARs para cambios de enrutamiento

Flujo de Modificación de Sesión:

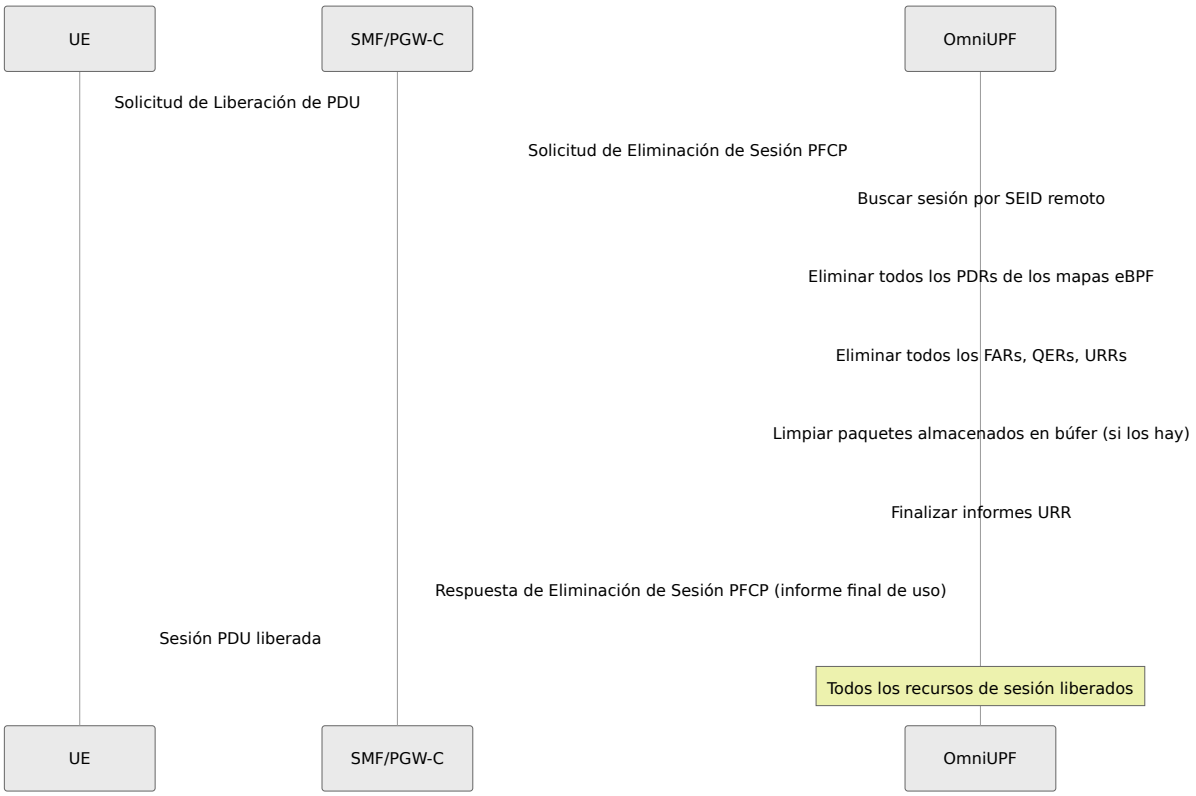


Para la gestión de reglas, consulte Guía de Gestión de Reglas.

Eliminación de Sesión PFCP

Cuando se libera una sesión PDU, el SMF elimina la sesión PFCP en el UPF.

Flujo de Eliminación de Sesión:



Limpieza Realizada:

- Todos los PDRs eliminados (subida y bajada)
- Todos los FARs, QERs, URRs eliminados
- Buffers de paquetes limpiados
- Informe final de uso enviado al SMF para facturación

Operaciones Comunes

OmniUPF proporciona capacidades operativas integrales a través de su panel de control basado en web y API REST. Esta sección cubre tareas operativas comunes y su importancia.

Monitoreo de Sesiones

Entendiendo las Sesiones PFCP:

Las sesiones PFCP representan sesiones PDU activas de UE (5G) o contextos PDP (LTE). Cada sesión contiene:

- SEIDs local y remoto (Identificadores de Punto de Sesión)

- PDRs para clasificación de paquetes
- FARs para decisiones de reenvío
- QERs para aplicación de QoS (opcional)
- URRs para seguimiento de uso (opcional)

Operaciones Clave de Sesión:

- **Ver todas las sesiones** con direcciones IP de UE, TEIDs y conteos de reglas
- **Filtrar sesiones** por dirección IP o TEID
- **Inspeccionar detalles de sesión** incluyendo configuraciones completas de PDR/FAR/QER/URR
- **Monitorear conteos de sesiones** por asociación PFCP

Para procedimientos detallados de sesiones, consulte Vista de Sesiones.

Gestión de Reglas

Reglas de Detección de Paquetes (PDR):

Las PDRs determinan qué paquetes coinciden con flujos de tráfico específicos. Los operadores pueden:

- **Ver PDRs de subida** indexados por TEID desde la interfaz N3
- **Ver PDRs de bajada** indexados por dirección IP de UE (IPv4 e IPv6)
- **Inspeccionar filtros SDF** para clasificación específica de aplicaciones
- **Monitorear conteos de PDR** y uso de capacidad

Reglas de Acción de Reenvío (FAR):

Las FARs definen qué hacer con los paquetes coincidentes. Los operadores pueden:

- **Ver acciones FAR** (REENVIAR, DESCARTAR, ALMACENAR EN BÚFER, DUPLICAR, NOTIFICAR)
- **Inspeccionar parámetros de reenvío** (creación de encabezado externo, destino)

- **Monitorear estado de almacenamiento en búfer** por FAR
- **Alternar almacenamiento en búfer** para FARs específicas durante la resolución de problemas

Reglas de Aplicación de QoS (QER):

Las QERs aplican límites de ancho de banda y marcado de paquetes. Los operadores pueden:

- **Ver parámetros de QoS** (MBR, GBR, marcado de paquetes)
- **Monitorear QERs activas** por sesión
- **Inspeccionar marcas QFI** para flujos de QoS 5G

Reglas de Informe de Uso (URR):

Las URRs rastrean volúmenes de datos para facturación. Los operadores pueden:

- **Ver contadores de volumen** (subida, bajada, total de bytes)
- **Monitorear umbrales de uso** y disparadores de informes
- **Inspeccionar URRs activas** a través de todas las sesiones

Para operaciones de reglas, consulte Guía de Gestión de Reglas.

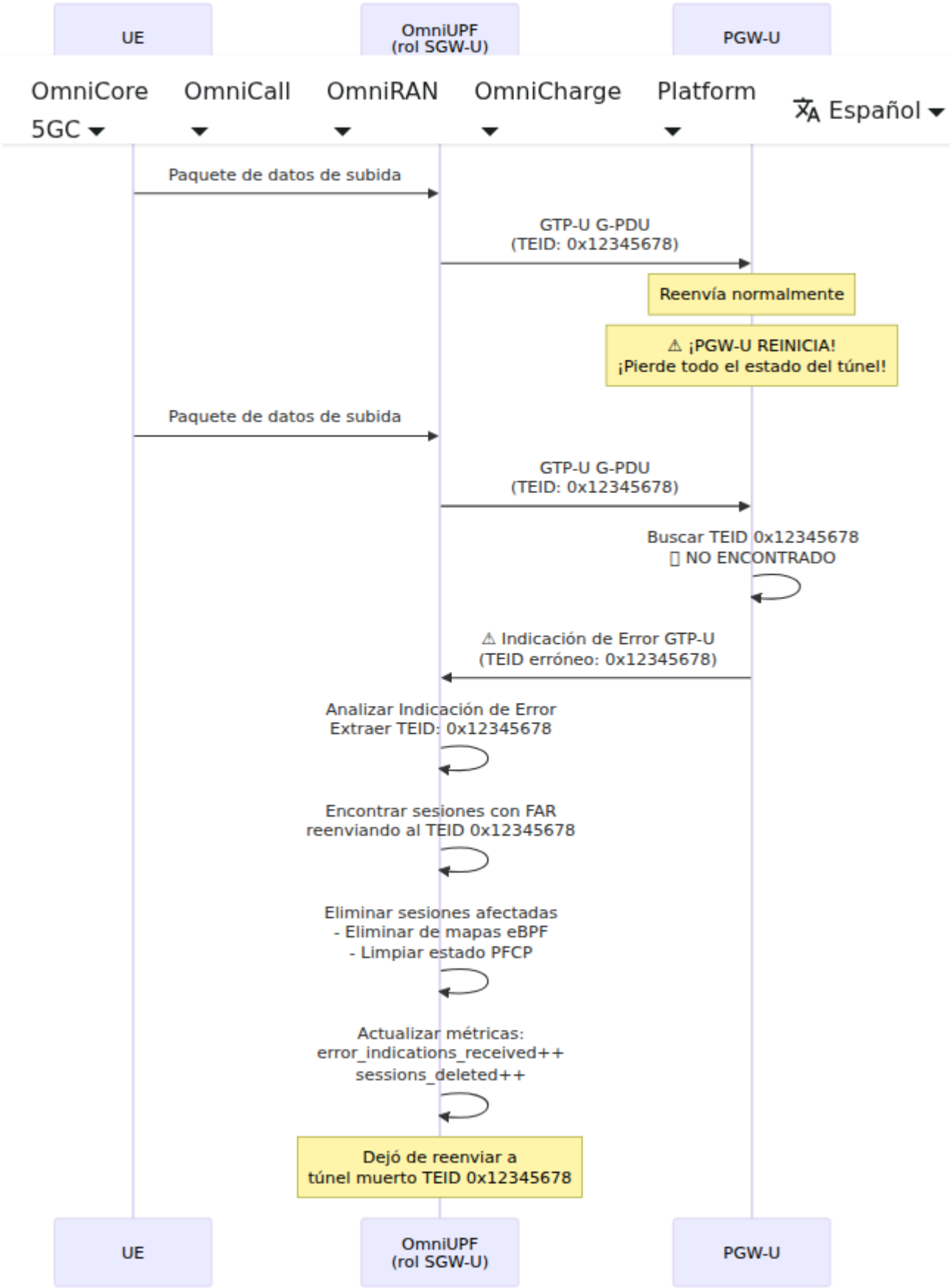
Almacenamiento en Búfer de Paquetes

Por qué el Almacenamiento en Búfer es Crítico para UPF

El almacenamiento en búfer de paquetes es una de las funciones más importantes de un UPF porque previene la pérdida de paquetes durante eventos de movilidad y reconfiguraciones de sesión. Sin almacenamiento en búfer, los usuarios móviles experimentarían desconexiones, descargas interrumpidas y fallos en las comunicaciones en tiempo real cada vez que se mueven entre torres celulares o cuando cambian las condiciones de la red.

El Problema: Pérdida de Paquetes Durante la Movilidad

En redes móviles, los usuarios están en constante movimiento. Cuando un dispositivo se mueve de una torre celular a otra (transferencia), o cuando la red necesita reconfigurar la ruta de datos, hay una ventana crítica donde los paquetes están en vuelo pero la nueva ruta aún no está lista:



Sin almacenamiento en búfer: Los paquetes que llegan durante esta ventana crítica serían **descartados**, causando:

- **Conexiones TCP que se detienen** o se reinician (navegación web, descargas interrumpidas)
- **Videollamadas que se congelan** o se caen (Zoom, Teams, llamadas de WhatsApp fallan)
- **Sesiones de juego que se desconectan** (juegos en línea, aplicaciones en tiempo real fallan)
- **Llamadas VoIP que tienen interrupciones** o se caen completamente (llamadas telefónicas interrumpidas)
- **Descargas que fallan** y necesitan reiniciarse

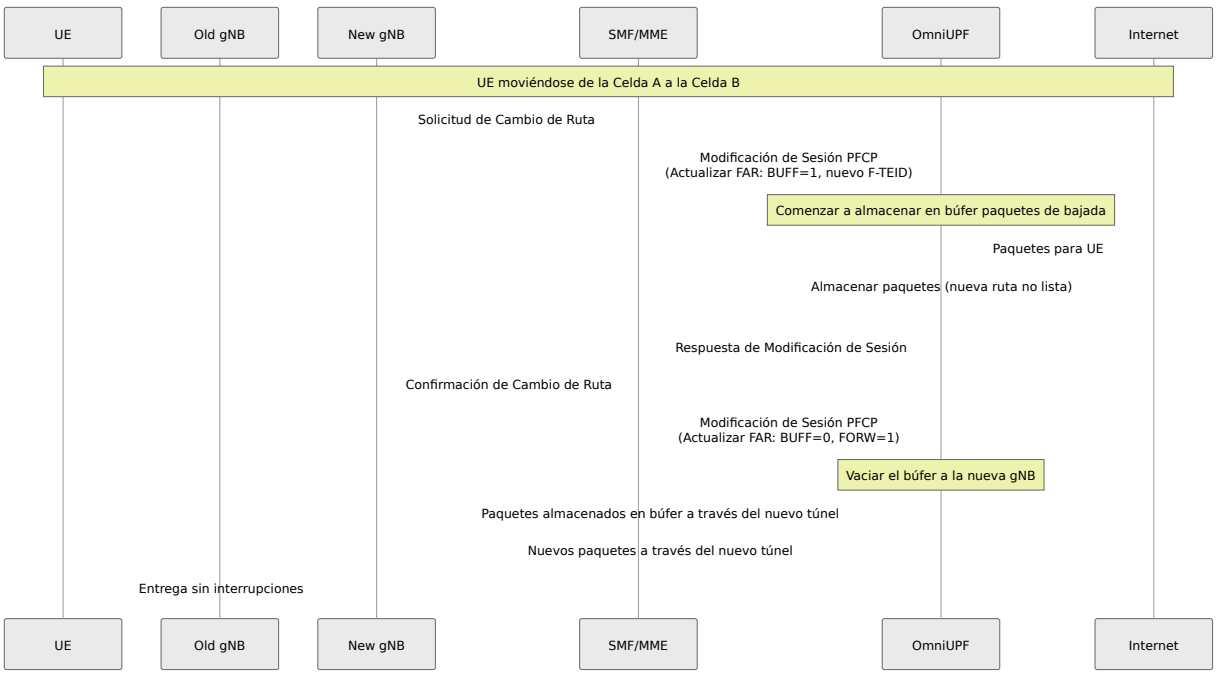
Con almacenamiento en búfer: OmniUPF retiene temporalmente los paquetes hasta que la nueva ruta se establece, luego los reenvía sin problemas. El usuario experimenta **cero interrupciones**.

Cuándo Ocurre el Almacenamiento en Búfer

OmniUPF almacena paquetes en búfer en estos escenarios críticos:

1. Transferencia Basada en N2 (5G) / Transferencia Basada en X2 (4G)

Cuando un UE se mueve entre torres celulares:



Línea de Tiempo:

- **T+0ms**: Ruta antigua aún activa
- **T+10ms**: SMF le dice a UPF que almacene en búfer (ruta antigua cerrando, nueva ruta no lista)
- **T+10-50ms**: **Ventana crítica de almacenamiento en búfer** - los paquetes llegan pero no se pueden reenviar
- **T+50ms**: Nueva ruta lista, SMF le dice a UPF que reenvíe
- **T+50ms+**: UPF vacía los paquetes almacenados en búfer a la nueva ruta, luego reenvía nuevos paquetes normalmente

Sin almacenamiento en búfer: ~40ms de paquetes (potencialmente miles) serían **perdidos**. **Con almacenamiento en búfer**: Cero pérdida de paquetes, transferencia sin interrupciones.

---

2. Modificación de Sesión (Cambio de QoS, Actualización de Ruta)

Cuando la red necesita cambiar parámetros de sesión:

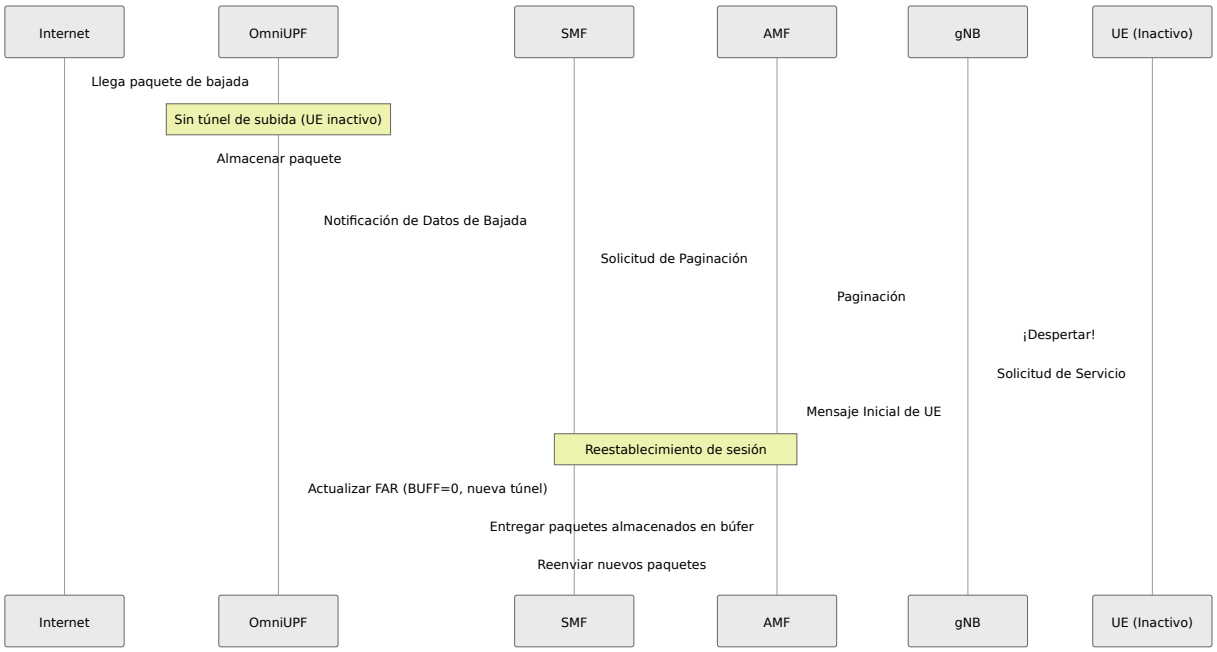
- **Actualización/disminución de QoS**: El usuario se mueve de cobertura 4G a 5G (modo NSA)
- **Cambio de política**: El usuario de empresa entra en el campus corporativo (cambios de direccionamiento de tráfico)
- **Optimización de red**: La red central redirige el tráfico a un UPF más cercano (actualización ULCL)

Durante la modificación, el plano de control puede necesitar actualizar múltiples reglas de manera atómica. El almacenamiento en búfer asegura que los paquetes no se reenvíen con conjuntos de reglas parciales/inconsistentes.

---

3. Notificación de Datos de Bajada (Recuperación en Modo Inactivo)

Cuando un UE está en modo inactivo (pantalla apagada, ahorro de batería) y llegan datos de bajada:



Sin almacenamiento en búfer: El paquete inicial que activó la notificación sería **perdido**, requiriendo que el remitente retransmita (agrega latencia). **Con almacenamiento en búfer**: El paquete que despertó al UE se entrega inmediatamente cuando el UE se reconecta.

---

4. Transferencia Inter-RAT (4G ↔ 5G)

Cuando un UE se mueve entre cobertura 4G y 5G:

- Cambios de arquitectura (eNodeB ↔ gNB)
- Cambian los puntos finales de túnel (nueva asignación de TEID)
- El almacenamiento en búfer asegura una transición suave entre tipos de RAT

Cómo Funciona el Almacenamiento en Búfer en OmniUPF

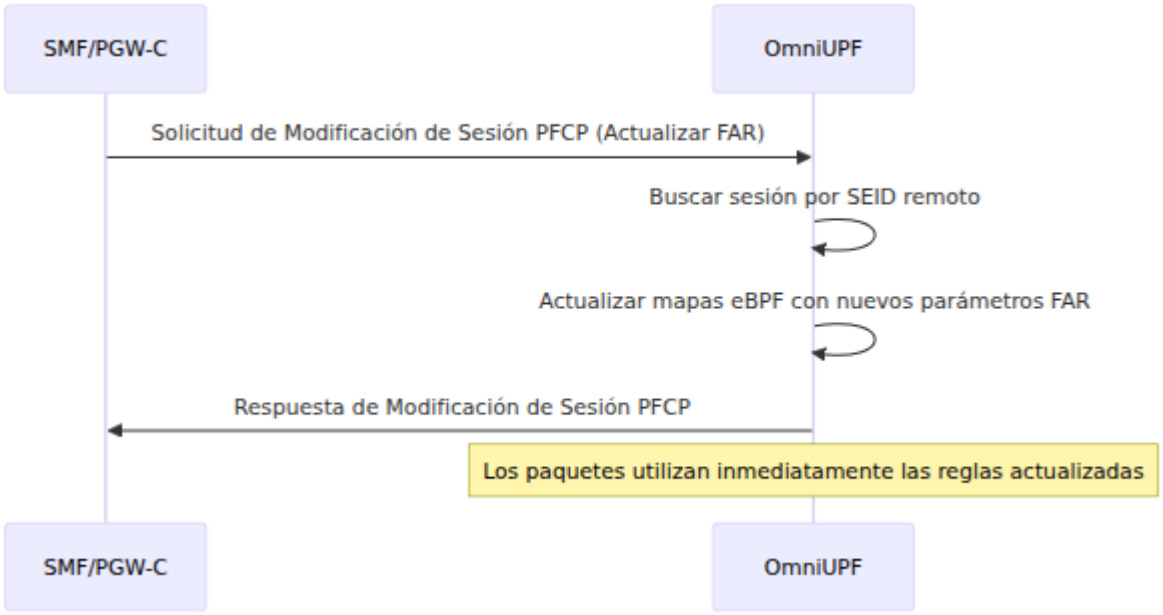
Mecanismo Técnico:

OmniUPF utiliza una **arquitectura de almacenamiento en búfer de dos etapas**:

1. **Etapa eBPF (Núcleo)**: Detecta paquetes que requieren almacenamiento en búfer basándose en las banderas de acción FAR

2. **Etapa de Espacio de Usuario**: Almacena y gestiona paquetes almacenados en búfer en memoria

Proceso de Almacenamiento en Búfer:



Detalles Clave:

- **Puerto de Búfer**: Puerto UDP 22152 (paquetes enviados desde eBPF a espacio de usuario)
- **Encapsulación**: Paquetes envueltos en GTP-U con ID FAR como TEID
- **Almacenamiento**: Buffers en memoria por FAR con metadatos (marca de tiempo, dirección, tamaño del paquete)
- **Límites**:
  - Límite por FAR: 10,000 paquetes (por defecto)
  - Límite global: 100,000 paquetes en todos los FARs
  - TTL: 30 segundos (por defecto) - paquetes más antiguos que TTL se descartan
- **Limpieza**: Proceso en segundo plano elimina paquetes expirados cada 60 segundos

Ciclo de Vida del Búfer:

1. **Almacenamiento en Búfer Habilitado**: SMF establece acción FAR BUFF=1 (bit 2) a través de Modificación de Sesión PFCP

2. **Paquetes Almacenados en Búfer**: eBPF detecta la bandera BUFF, encapsula paquetes, envía al puerto 22152
3. **Almacenamiento en Espacio de Usuario**: El gestor de búfer almacena paquetes con ID FAR, marca de tiempo, dirección
4. **Almacenamiento en Búfer Deshabilitado**: SMF establece acción FAR FORW=1, BUFF=0 con nuevos parámetros de reenvío
5. **Vaciar Búfer**: El espacio de usuario reproduce paquetes almacenados en búfer utilizando nuevas reglas FAR (nuevo punto final de túnel)
6. **Reanudar Normalidad**: Nuevos paquetes reenviados inmediatamente a través de la nueva ruta

---

Por qué Esto Importa para la Experiencia del Usuario

Impacto en el Mundo Real:

Escenario	Sin Almacenamiento en Búfer	Con Almacenamiento en Búfer
Videollamada Durante la Transferencia	La llamada se congela durante 1-2 segundos, puede caerse	Sin interrupciones, sin problemas
Descarga de Archivo en el Límite de la Celda	La descarga falla, debe reiniciarse	La descarga continúa sin interrupciones
Juego en Línea Mientras se Mueve	La conexión se cae, se expulsa del juego	Juego fluido, sin desconexiones
Llamada VoIP en el Coche	La llamada se cae en cada transferencia	Clara como el cristal, sin caídas
Streaming de Video en el Tren	El video se almacena en búfer, la calidad disminuye	Reproducción fluida
Hotspot Móvil para Portátil	La sesión SSH se cae, la videollamada falla	Todas las conexiones mantenidas

Beneficios para el Operador de Red:

- **Reducción de la Tasa de Caídas de Llamadas (CDR)**: KPI crítico para la calidad de la red
- **Mayor Satisfacción del Cliente**: Los usuarios no notan las transferencias
- **Menores Costos de Soporte**: Menos quejas sobre conexiones caídas
- **Ventaja Competitiva**: Marketing de "mejor red para cobertura"

Operaciones de Gestión de Búfer

Los operadores pueden monitorear y controlar el almacenamiento en búfer a través de la Interfaz Web y la API:

Monitoreo:

- **Ver paquetes almacenados en búfer** por ID FAR (conteo, bytes, edad)
- **Rastrear uso del búfer** contra límites (por FAR, global)
- **Alertar sobre desbordamiento de búfer** o duración excesiva de almacenamiento en búfer
- **Identificar búferes atascados** (paquetes almacenados en búfer > umbral TTL)

Operaciones de Control:

- **Vaciar búferes**: Activar manualmente la reproducción del búfer (resolución de problemas)
- **Limpiar búferes**: Descartar paquetes almacenados en búfer (limpiar búferes atascados)
- **Ajustar TTL**: Cambiar el tiempo de expiración de paquetes
- **Modificar límites**: Aumentar la capacidad del búfer por FAR o global

Resolución de Problemas:

- **Búfer no vaciándose**: Verifique si el SMF envió una actualización de FAR para deshabilitar el almacenamiento en búfer
- **Desbordamiento de búfer**: Aumente los límites o investigue por qué la duración del almacenamiento en búfer es excesiva
- **Paquetes antiguos en el búfer**: TTL puede ser demasiado alto, o la actualización de FAR retrasada
- **Almacenamiento en búfer excesivo**: Puede indicar problemas de movilidad o problemas con el SMF

Para operaciones detalladas de búfer, consulte Guía de Gestión de Búfer.

Configuración del Búfer

Configure el comportamiento del almacenamiento en búfer en `/etc/omniupf/runtime.exs`:

```
# Configuración del búfer
buffer_port = 22152                # Puerto UDP para paquetes
almacenados en búfer (por defecto)
```

Recomendaciones:

- **Redes de alta movilidad** (autopistas, trenes): Aumente `buffer_max_packets` a 20,000+
- **Áreas urbanas densas** (transferencias frecuentes): Disminuya `buffer_packet_ttl` a 15s
- **Aplicaciones de baja latencia**: Establezca `buffer_packet_ttl` a 10s para prevenir datos obsoletos
- **Redes IoT**: Disminuya límites (los dispositivos IoT generan menos tráfico durante la transferencia)

Para opciones completas de configuración, consulte Guía de Configuración.

Estadísticas y Monitoreo

Estadísticas de Paquetes:

Métricas de procesamiento de paquetes en tiempo real que incluyen:

- **Paquetes RX**: Total recibido de todas las interfaces
- **Paquetes TX**: Total transmitido a todas las interfaces
- **Paquetes descartados**: Paquetes descartados debido a errores o políticas
- **Paquetes GTP-U**: Conteos de paquetes tunelados

Estadísticas de Rutas:

Métricas de reenvío por ruta:

- **Acercamientos de ruta**: Paquetes coincidentes por cada ruta

- **Conteos de reenvío**: Éxito/fallo por destino
- **Contadores de error**: TEIDs inválidos, IPs de UE desconocidas

Estadísticas de XDP:

Métricas de rendimiento de eXpress Data Path:

- **XDP procesados**: Paquetes manejados en la capa XDP
- **XDP pasados**: Paquetes enviados a la pila de red
- **XDP descartados**: Paquetes descartados en la capa XDP
- **XDP abortados**: Errores de procesamiento

Estadísticas de Interfaces N3/N6:

Contadores de tráfico por interfaz:

- **N3 RX/TX**: Tráfico hacia/desde RAN (gNB/eNodeB)
- **N6 RX/TX**: Tráfico hacia/desde la red de datos
- **Conteos totales de paquetes**: Estadísticas agregadas de interfaz

Para detalles de monitoreo, consulte Guía de Monitoreo.

Gestión de Capacidad

Monitoreo de Capacidad de Mapas eBPF:

El rendimiento de UPF depende de la capacidad de los mapas eBPF. Los operadores pueden:

- **Monitorear uso de mapas** con indicadores de porcentaje en tiempo real
- **Ver límites de capacidad** para cada mapa eBPF
- **Alertas codificadas por colores**:
  - Verde (<50%): Normal
  - Amarillo (50-70%): Precaución
  - Ámbar (70-90%): Advertencia
  - Rojo (>90%): Crítico

Mapas Críticos a Monitorear:

- `uplink_pdr_map`: Clasificación de tráfico de subida
- `downlink_pdr_map`: Clasificación de tráfico de bajada IPv4
- `far_map`: Reglas de reenvío
- `qer_map`: Reglas de QoS
- `urr_map`: Seguimiento de uso

Planificación de Capacidad:

- Cada PDR consume una entrada de mapa (tamaño de clave + tamaño de valor)
- La capacidad del mapa se configura al inicio del UPF (límite de memoria del núcleo)
- Superar la capacidad causa fallos en el establecimiento de sesiones

Para el monitoreo de capacidad, consulte Gestión de Capacidad.

Gestión de Configuración

Configuración del UPF:

Ver y verificar parámetros operacionales del UPF:

- **Interfaz N3**: Dirección IP para conectividad RAN (GTP-U)
- **Interfaz N6**: Dirección IP para conectividad de red de datos
- **Interfaz N9**: Dirección IP para comunicación inter-UPF (opcional)
- **Interfaz PFCP**: Dirección IP para conectividad SMF
- **Puerto API**: Puerto de escucha de API REST
- **Punto de Métricas**: Puerto de métricas de Prometheus

Configuración del Dataplane:

Parámetros activos de la ruta de datos eBPF:

- **Dirección N3 activa**: Vinculación de interfaz N3 en tiempo de ejecución

- **Dirección N9 activa**: Vinculación de interfaz N9 en tiempo de ejecución (si está habilitada)

Para la visualización de la configuración, consulte Vista de Configuración.

Resolución de Problemas

Esta sección cubre problemas operativos comunes y sus estrategias de resolución.

Fallos en el Establecimiento de Sesiones

Síntomas: Las sesiones PFCP no se crean, el UE no puede establecer conectividad de datos

Causas Raíz Comunes:

1. **Asociación PFCP No Establecida**
    - Verifique que el SMF pueda alcanzar la interfaz PFCP del UPF (puerto 8805)
    - Verifique el estado de la asociación PFCP en la vista de Sesiones
    - Verifique que la configuración del ID de Nodo coincida entre SMF y UPF
2. **Exhaustión de Capacidad del Mapa eBPF**
    - Verifique la vista de Capacidad para el uso del mapa rojo (>90%)
    - Aumente los tamaños de los mapas eBPF en la configuración del UPF
    - Elimine sesiones obsoletas si el mapa está lleno
3. **Configuración Inválida de PDR/FAR**
    - Verifique que la dirección IP de UE sea única y válida
    - Verifique que la asignación de TEID no entre en conflicto
    - Asegúrese de que FAR haga referencia a instancias de red válidas
4. **Problemas de Configuración de Interfaz**

- Verifique que la IP de la interfaz N3 sea alcanzable desde gNB
- Verifique tablas de enrutamiento para conectividad N6 a la red de datos
- Confirme que el tráfico GTP-U no esté

Guía de Arquitectura de OmniUPF

Tabla de Contenidos

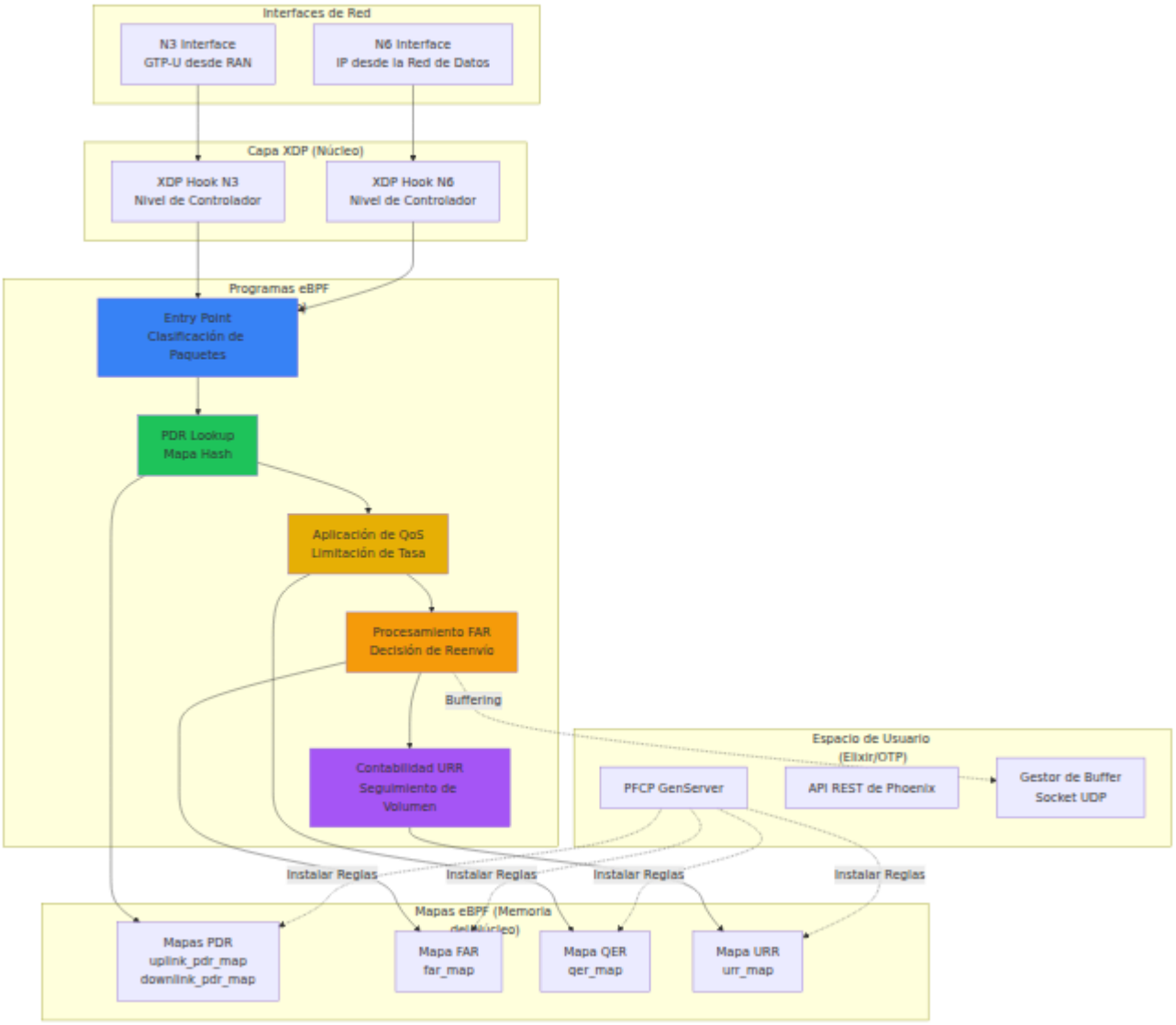
1. Descripción General
2. Fundación de Tecnología eBPF
3. Ruta de Datos XDP
4. Pipeline de Procesamiento de Paquetes
5. Arquitectura de Mapas eBPF
6. Mecanismo de Buffering
7. Aplicación de QoS
8. Características de Rendimiento
9. Escalabilidad y Ajuste

Descripción General

OmniUPF aprovecha eBPF (filtro de paquetes de Berkeley extendido) y XDP (ruta de datos eXpress) para lograr un rendimiento de grado de operador para el procesamiento de paquetes 5G/LTE. El plano de control se implementa en Elixir/OTP utilizando gestión de sesiones PFCP basada en GenServer, mientras que el plano de datos ejecuta programas eBPF directamente en el núcleo de Linux. Esta separación elimina la sobrecarga del procesamiento de paquetes en espacio de usuario y logra un rendimiento de múltiples gigabits con latencia de microsegundos.

El archivo objeto eBPF (`ipentrypoint_bpf.o`) se compila a partir de código fuente C (en el directorio `ebpf/xdp/`) y se carga en tiempo de ejecución por el plano de control de Elixir a través de enlaces NIF a libbpf.

Capas de Arquitectura



Principios de Diseño Clave

Procesamiento Sin Copia:

- Paquetes procesados completamente en espacio de núcleo
- Sin copia de datos entre núcleo y espacio de usuario
- Manipulación directa de paquetes usando XDP

Estructuras de Datos Sin Bloqueo:

- Los mapas eBPF utilizan tablas hash por CPU
- Operaciones atómicas para acceso concurrente

- Sin sobrecarga de mutex/bloqueo

Listo para Descarga de Hardware:

- El modo de descarga XDP admite la ejecución en SmartNIC
- Compatible con tarjetas de red que soportan XDP
- Reversa a modos nativos de controlador o genéricos

Fundación de Tecnología eBPF

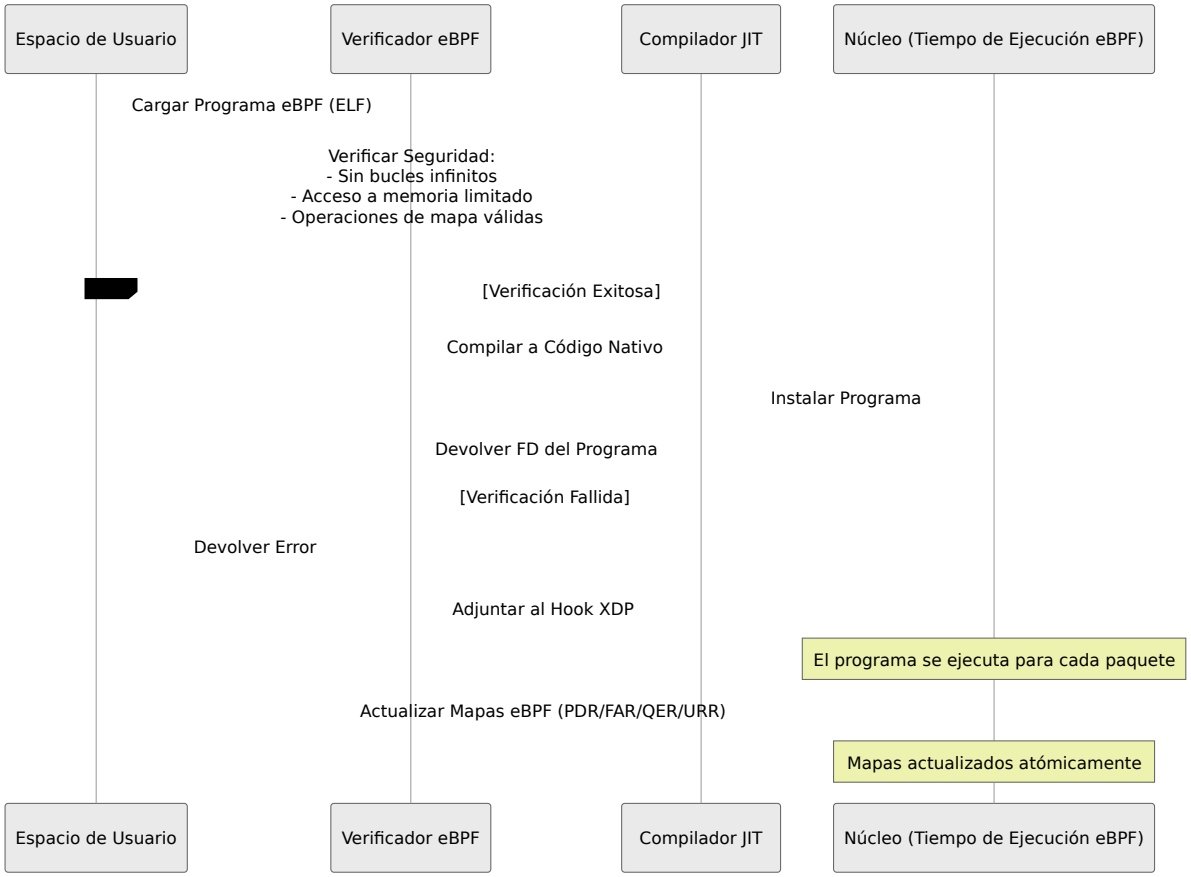
¿Qué es eBPF?

eBPF (filtro de paquetes de Berkeley extendido) es una tecnología revolucionaria del núcleo de Linux que permite que programas seguros y en sandbox se ejecuten en espacio de núcleo sin cambiar el código fuente del núcleo o cargar módulos del núcleo.

Características Clave:

- **Seguridad**: El verificador eBPF asegura que los programas no pueden hacer que el núcleo se bloquee
- **Rendimiento**: Se ejecuta a la velocidad nativa del núcleo (sin sobrecarga de interpretación)
- **Flexibilidad**: Puede ser actualizado en tiempo de ejecución sin reiniciar el núcleo
- **Observabilidad**: Trazado y estadísticas integradas

Ciclo de Vida del Programa eBPF



Mapas eBPF

Los mapas eBPF son estructuras de datos del núcleo compartidas entre programas eBPF y espacio de usuario.

Tipos de Mapa Usados en OmniUPF:

Tipo de Mapa	Descripción	Caso de Uso
`BPF_MAP_TYPE_HASH`	Tabla hash con pares clave-valor	Búsqueda PDR por TEID o IP de UE
`BPF_MAP_TYPE_ARRAY`	Array indexado por entero	Búsqueda QER, FAR, URR por ID
`BPF_MAP_TYPE_PERCPU_HASH`	Tabla hash por CPU (sin bloqueo)	Búsquedas PDR de alto rendimiento
`BPF_MAP_TYPE_LRU_HASH`	Hash LRU (Menos Usado Recientemente)	Evicción automática de entradas antiguas

Operaciones de Mapa:

- **Búsqueda**: O(1) búsqueda hash (sub-microsegundo)
- **Actualización**: Actualizaciones atómicas desde espacio de usuario
- **Eliminación**: Eliminación inmediata de entradas
- **Iterar**: Operaciones por lotes para volcar mapas

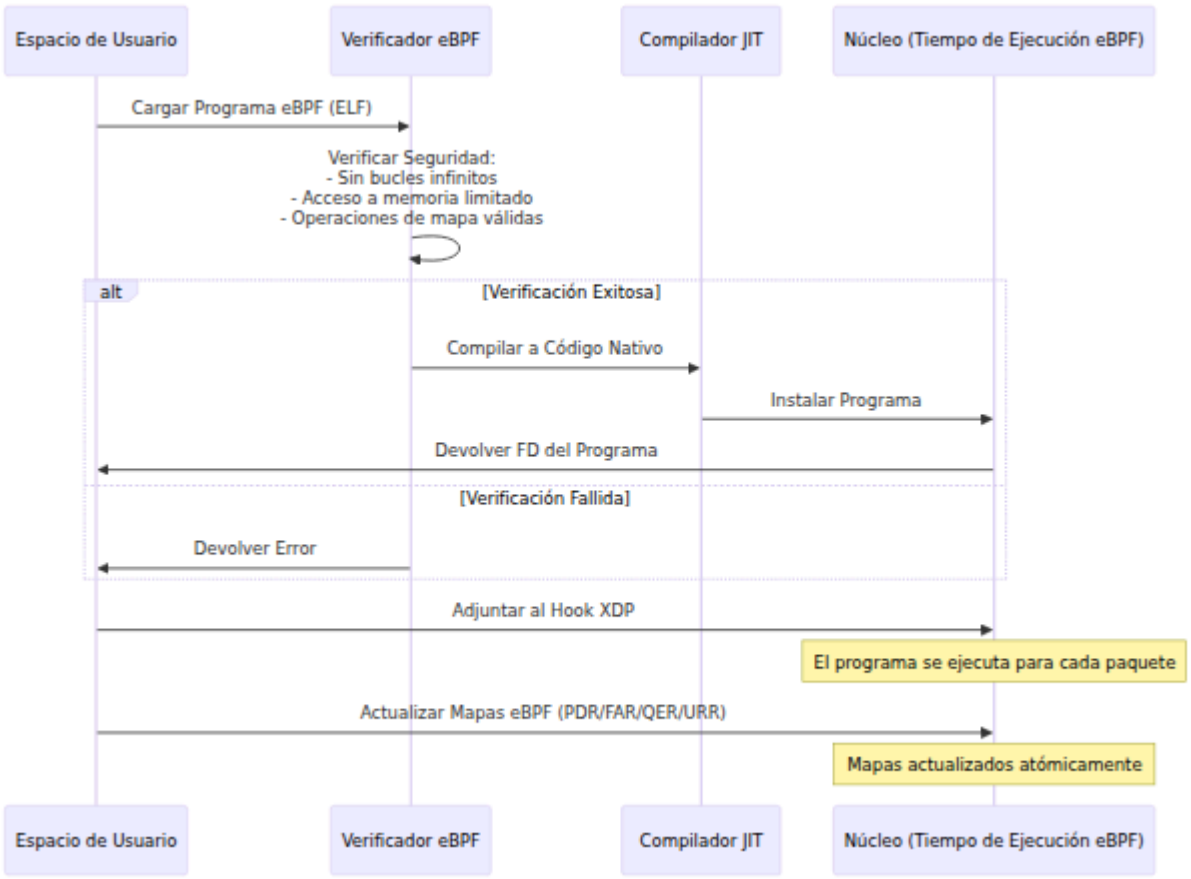
Ruta de Datos XDP

Descripción General de XDP

XDP (ruta de datos eXpress) es un hook del núcleo de Linux que permite que programas eBPF procesen paquetes en el punto más temprano posible—justo después de que el controlador de red los recibe, antes de la pila de red del núcleo.

Modos de Adjuntar XDP

OmniUPF admite tres modos de adjuntar XDP, cada uno con diferentes características de rendimiento y compatibilidad.



1. Modo de Descarga XDP

Ejecución en Hardware (Mejor Rendimiento):

- El programa eBPF se ejecuta directamente en el hardware de SmartNIC
- Procesamiento de paquetes en NIC sin tocar la CPU
- Logra un rendimiento de 100 Gbps+
- Requiere SmartNIC compatible (Netronome, Mellanox ConnectX-6)

Configuración (en `runtime.exs`):

```
xdp_attach_mode = "offload"
```

Limitaciones:

- Requiere hardware SmartNIC costoso
- Complejidad limitada del programa eBPF
- No todas las características eBPF son compatibles en hardware

---

2. Modo Nativo XDP (Predeterminado para Producción)

Ejecución a Nivel de Controlador (Alto Rendimiento):

- El programa eBPF se ejecuta en el contexto del controlador de red
- Los paquetes se procesan antes de la asignación de SKB (buffer de socket)
- Logra de 10 a 40 Gbps por núcleo
- Requiere un controlador con soporte XDP (la mayoría de los controladores modernos)

Configuración (en `runtime.exs`):

```
xdp_attach_mode = "native"
```

Ventajas:

- Rendimiento muy alto (multimillones de pps)
- Amplia compatibilidad de hardware
- Conjunto completo de características eBPF

Controladores Soportados:

- Intel: i40e, ice, ixgbe, igb
- Mellanox: mlx4, mlx5
- Broadcom: bnxt
- Amazon: ena
- La mayoría de las tarjetas de red de 10G+

---

3. Modo Genérico XDP

Emulación de Software (Compatibilidad):

- El programa eBPF se ejecuta después de que el núcleo asigna SKB
- Emulación de software del comportamiento XDP
- Funciona en cualquier interfaz de red
- Útil para pruebas y desarrollo

Configuración (en `runtime.exs`):

```
xdp_attach_mode = "generic"
```

Casos de Uso:

- Desarrollo y pruebas
- Entornos virtualizados (VMs sin SR-IOV)
- Hardware de red más antiguo
- Pruebas de interfaz de loopback

Rendimiento: 1-5 Gbps (significativamente más lento que nativo/descarga)

Códigos de Retorno XDP

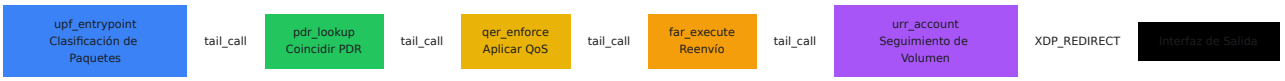
Los programas eBPF devuelven códigos de acción XDP para indicar al núcleo qué hacer con los paquetes:

Código de Retorno	Significado	Uso en OmniUPF
`XDP_PASS`	Enviar paquete a la pila de red del núcleo	Buffering (entrega local), ICMP, tráfico desconocido
`XDP_DROP`	Eliminar paquete inmediatamente	Paquetes inválidos, limitación de tasa, eliminaciones de políticas
`XDP_TX`	Transmitir paquete de vuelta por la misma interfaz	No se utiliza actualmente
`XDP_REDIRECT`	Enviar paquete a una interfaz diferente	Ruta principal de reenvío (N3 ↔ N6)
`XDP_ABORTED`	Error de procesamiento, eliminar paquete y registrar	Errores del programa eBPF

Pipeline de Procesamiento de Paquetes

Estructura del Programa

OmniUPF utiliza llamadas de cola eBPF para crear un pipeline modular de procesamiento de paquetes.



Llamadas de Cola:

- Permiten que programas eBPF llamen a otros programas eBPF
- Reutiliza el mismo marco de pila (profundidad de pila limitada)

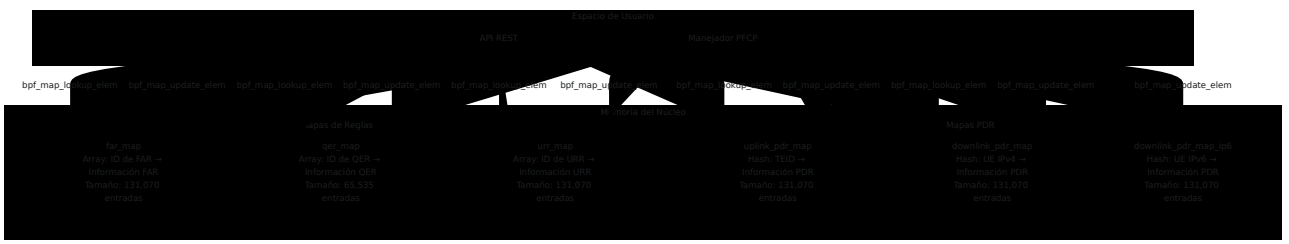
- Habilita el diseño modular del pipeline
- Profundidad máxima de 33 llamadas de cola

Procesamiento de Paquetes de Uplink

Procesamiento de Paquetes de Downlink

Arquitectura de Mapas eBPF

Diseño de Memoria del Mapa



Dimensionamiento del Mapa

OmniUPF calcula automáticamente los tamaños de los mapas en función de la configuración `max_sessions`:

```
Mapas PDR = 2 × max_sessions  (uplink + downlink)
Mapas FAR = 2 × max_sessions  (uplink + downlink)
Mapas QER = 1 × max_sessions  (compartido por sesión)
Mapas URR = 3 × max_sessions  (múltiples URRs por sesión)
```

Ejemplo (max_sessions = 65,535):

- Mapas PDR: 131,070 entradas cada uno
- Mapa FAR: 131,070 entradas
- Mapa QER: 65,535 entradas
- Mapa URR: 131,070 entradas

Memoria Total:

```
Mapas PDR: 3 × 131,070 × 212 B = ~83 MB
Mapa FAR:  131,070 × 20 B = ~2.6 MB
Mapa QER:  65,535 × 36 B = ~2.3 MB
Mapa URR:  131,070 × 20 B = ~2.6 MB
Total: ~91 MB de memoria del núcleo
```

Mecanismo de Buffering

Descripción General del Buffering

OmniUPF implementa buffering de paquetes para escenarios de transferencia encapsulando paquetes en GTP-U y enviándolos a un proceso de espacio de usuario a través de un socket UDP.

Arquitectura de Buffering

Parse error on line 4: ...egar Encabezado UDP (puerto 22152)
3 ----------
----------^ Expecting 'SQE', 'DOUBLECIRCLEEND', 'PE', '-)', 'STADIUMEND',
'SUBROUTINEEND', 'PIPE', 'CYLINDEREND', 'DIAMOND_STOP', 'TAGEND',
'TRAPEND', 'INVTRAPEND', 'UNICODE_TEXT', 'TEXT', 'TAGSTART', got 'PS'

Intente de nuevo

Detalles de Encapsulación del Buffer

Cuando el buffering está habilitado (bit de acción FAR 2 establecido), el programa eBPF:

1. **Calcula el Tamaño del Paquete Original**:

```
orig_packet_len = ntohs(ip->tot_len);  // Desde el encabezado
IP
```

2. **Expande el Encabezado del Paquete**:

```
// Agregar espacio para: IP Externa + UDP + GTP-U
gtp_encap_size = sizeof(struct iphdr) + sizeof(struct udphdr) +
sizeof(struct gtpuhdr);
bpf_xdp_adjust_head(ctx, -gtp_encap_size);
```

3. **Construye el Encabezado IP Externo**:

```
ip->saddr = original_sender_ip;  // Preservar fuente para
evitar filtrado martiano
ip->daddr = local_upf_ip;        // IP local donde se vincula
el listener de espacio de usuario
ip->protocol = IPPROTO_UDP;
ip->ttl = 64;
```

4. **Construye el Encabezado UDP**:

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

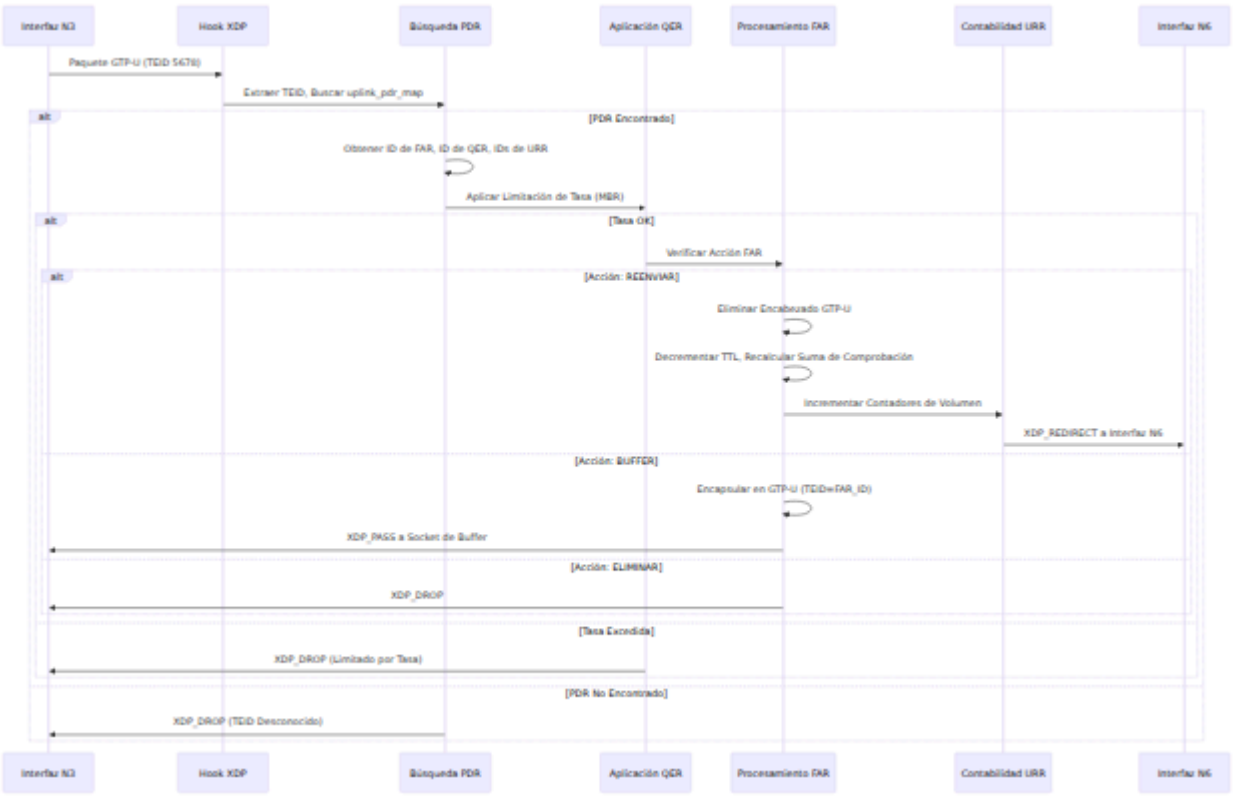
5. **Construye el Encabezado GTP-U**:

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

6. **Devuelve XDP_PASS**:
    - El núcleo entrega el paquete al socket UDP local en el puerto 22152
    - El gestor de buffer de espacio de usuario recibe y almacena el paquete

Operación de Vaciamiento del Buffer

Cuando la transferencia se completa, el SMF actualiza el FAR para limpiar la bandera BUFFER. Los paquetes almacenados se reproducen:



Parámetros de Gestión del Buffer

Parámetro	Predeterminado	Descripción
Máx. por FAR	10,000 paquetes	Máximo de paquetes almacenados por FAR
Máx. Total	100,000 paquetes	Máximo total de paquetes almacenados
TTL del Paquete	30 segundos	Tiempo antes de que los paquetes almacenados expiren
Puerto de Buffer	22152	Puerto UDP para entrega de buffer
Intervalo de Limpieza del Buffer	60 segundos	Frecuencia para verificar paquetes expirados

Aplicación de QoS

Algoritmo de Limitación de Tasa

OmniUPF implementa un **limitador de tasa de ventana deslizante** para la aplicación de QoS.

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

Intente de nuevo

Implementación de Ventana Deslizante

Algoritmo (desde `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;  // ventana de 5ms

    // Tasa = 0 significa ilimitado
    if (rate == 0)
        return XDP_PASS;

    // Calcular tiempo de transmisión para este paquete
    __u64 tx_time = packet_size * 8 * (NSEC_PER_SEC / rate);
    __u64 now = bpf_ktime_get_ns();

    // Verificar si estamos adelantados a la ventana (el paquete se transmitiría en el futuro)
    __u64 start = *window_start;
    if (start + tx_time > now)
        return XDP_DROP;  // Límite de tasa excedido

    // Si la ventana ha pasado, reiniciarla
    if (start + window_size < now) {
        *window_start = now - window_size + tx_time;
        return XDP_PASS;
    }

    // Actualizar ventana para tener en cuenta este paquete
    *window_start = start + tx_time;
    return XDP_PASS;
}
```

Parámetros Clave:

- **Tamaño de Ventana**: 5ms (5,000,000 nanosegundos)
- **Por Dirección**: Ventanas separadas para uplink y downlink
- **Actualizaciones Atómicas**: Utiliza punteros volátiles para acceso concurrente
- **MBR = 0**: Tratado como ancho de banda ilimitado

Ejemplo de Cálculo de QoS

Escenario: MBR = 100 Mbps, Tamaño del Paquete = 1500 bytes

1. **Tiempo de Transmisión**:

```
tx_time = 1500 bytes × 8 bits/byte × (1,000,000,000 ns/sec ÷
100,000,000 bps)
tx_time = 1500 × 8 × 10 = 120,000 ns = 120 µs
```

2. **Verificación de Tasa**:

   - Si el último paquete se transmitió en `t=0`, el siguiente paquete puede transmitirse en `t=120µs`
   - Si el paquete llega en `t=100µs`, se elimina (demasiado temprano)
   - Si el paquete llega en `t=150µs`, se reenvía (ventana avanzada)

3. **Tasa Máxima de Paquetes**:

```
Max PPS = (100 Mbps ÷ 8) ÷ 1500 bytes = 8,333 paquetes/segundo
Tiempo entre paquetes = 120 µs
```

Características de Rendimiento

Rendimiento

Configuración	Rendimiento	Paquetes/Segundo	Latencia
XDP Offload (SmartNIC)	100 Gbps	148 Mpps	< 1 µs
XDP Nativo (NIC de 10G, núcleo único)	10 Gbps	8 Mpps	2-5 µs
XDP Nativo (NIC de 10G, 4 núcleos)	40 Gbps	32 Mpps	2-5 µs
XDP Genérico	1-5 Gbps	0.8-4 Mpps	50-100 µs

Desglose de Latencia

Latencia Total de Procesamiento de Paquetes (XDP Nativo):

Etapa	Latencia	Acumulativa
RX de NIC	0.5 µs	0.5 µs
Invocación de Hook XDP	0.1 µs	0.6 µs
Búsqueda PDR (Hash)	0.3 µs	0.9 µs
Verificación de Tasa QER	0.1 µs	1.0 µs
Procesamiento FAR	0.5 µs	1.5 µs
Actualización URR	0.2 µs	1.7 µs
Encapsulación/Decapsulación GTP-U	0.8 µs	2.5 µs
XDP_REDIRECT	0.5 µs	3.0 µs
TX de NIC	0.5 µs	3.5 µs

Total: ~3.5 µs por paquete (XDP Nativo, NIC de 10G)

Utilización de CPU

Capacidad de Procesamiento por Núcleo:

- Núcleo único: 8-10 Mpps (XDP Nativo)
- Con hyper-threading: 12-15 Mpps
- Escalado multi-núcleo: Casi lineal hasta 8 núcleos

Uso de CPU por Tasa de Paquetes:

```
CPU % ≈ (Tasa de Paquetes / 10,000,000) × 100% por núcleo
```

Ejemplo: Tráfico de 2 Mpps utiliza ~20% de un núcleo

Ancho de Banda de Memoria

Acceso a Mapas eBPF:

- Búsqueda hash: ~100 ns (acierto de caché)
- Búsqueda hash: ~300 ns (fallo de caché)
- Búsqueda de array: ~50 ns (siempre acierto de caché)

Ancho de Banda de Memoria Requerido:

```
Ancho de Banda = Tasa de Paquetes × (Tamaño Promedio del Paquete + Búsquedas de Mapa × 64 bytes)
```

Ejemplo: 10 Mpps × (1500 B + 3 búsquedas × 64 B) ≈ 160 Gbps de ancho de banda de memoria

Escalabilidad y Ajuste

Escalado Horizontal

Múltiples Instancias de UPF:

Setting SMF as parent of SMF would create a cycle

Intente de nuevo

Distribución de Sesiones:

- SMF distribuye sesiones entre instancias de UPF
- Cada UPF maneja un subconjunto de sesiones de UE
- No se necesita comunicación inter-UPF (sin estado)

Escalado Vertical

Ajuste de CPU:

1. Habilitar afinidad de CPU para procesamiento XDP

2. Usar RSS (Escalado Lateral de Recepción) para distribuir colas RX
3. Fijar programas eBPF a núcleos específicos

Ajuste de NIC:

1. Aumentar el tamaño del búfer de recepción
2. Habilitar NICs de múltiples colas (RSS)
3. Usar director de flujo para direccionamiento de tráfico

Ajuste del Núcleo:

```
# Aumentar el límite de memoria bloqueada para mapas eBPF
ulimit -l unlimited

# Deshabilitar el equilibrio de IRQ para núcleos XDP
systemctl stop irqbalance

# Establecer el gobernador de CPU en rendimiento
cpupower frequency-set -g performance

# Aumentar los tamaños de búfer de red
sysctl -w net.core.rmem_max=134217728
sysctl -w net.core.wmem_max=134217728
```

Planificación de Capacidad

Fórmula:

```
Núcleos de CPU Requeridos = (PPS Esperados ÷ 10,000,000) × 1.5
(50% de margen)
Memoria Requerida = (Sesiones Máx. × 212 B × 3) + 100 MB (mapas
eBPF + sobrecarga)
Red Requerida = (Rendimiento Pico × 2) + 10 Gbps (margen)
```

Ejemplo (1 millón de sesiones, 20 Gbps pico):

- CPU: (20 Gbps ÷ 10 Gbps por núcleo) × 1.5 = 3-4 núcleos
- Memoria: (1M × 212 B × 3) + 100 MB ≈ 750 MB

- Red: (20 Gbps × 2) + 10 Gbps = 50 Gbps de interfaces

Documentación Relacionada

- **Guía de Operaciones UPF** - Operaciones generales de UPF y despliegue
- **Guía de Gestión de Reglas** - Detalles de PDR, FAR, QER, URR
- **Guía de Monitoreo** - Monitoreo de rendimiento y métricas
- **Guía de Operaciones de Interfaz Web** - Uso del panel de control
- **Guía de Solución de Problemas** - Problemas comunes y diagnósticos
- **Guía de Jardín Amurallado** - Redirección fuera de crédito utilizando el mecanismo de BUFFER para la interceptación de paquetes en espacio de usuario

Guía de Configuración de OmniUPF

Tabla de Contenidos

1. Descripción General
2. Modos de Operación
3. Modos de Adjunto XDP
4. Parámetros de Configuración
5. Archivo de Configuración
6. Compatibilidad con Hypervisores
7. Compatibilidad con NIC
8. Ejemplos de Configuración
9. Dimensionamiento de Mapas y Planificación de Capacidad

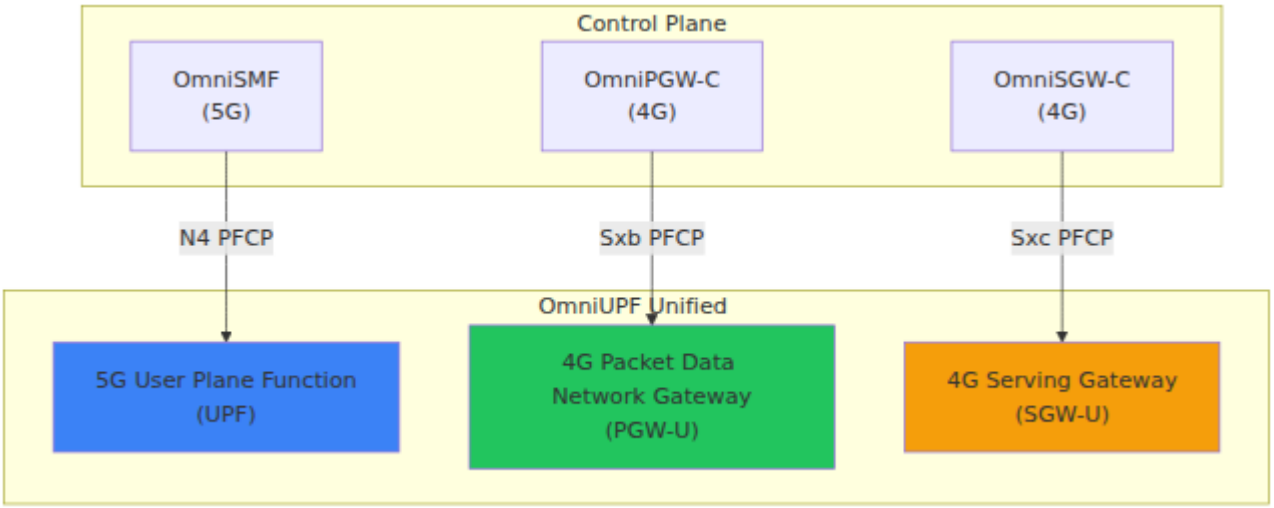
Descripción General

OmniUPF es una función de plano de usuario versátil que puede operar en múltiples modos para soportar redes centrales 4G (EPC) y 5G. El plano de control está implementado en Elixir/OTP, y la configuración se gestiona a través de un archivo de configuración de Elixir (`runtime.exs`).

Al instalar el paquete, el archivo de configuración se coloca en `/etc/omniupf/runtime.exs` y se preserva a través de las actualizaciones. El UPF debe reiniciarse después de los cambios de configuración.

Modos de Operación

OmniUPF es una **plataforma unificada** que puede operar simultáneamente como:



Configuración del Modo

El modo de operación es **determinado por el plano de control** (SMF, PGW-C o SGW-C) que establece asociaciones PFCP con OmniUPF. No se requiere una configuración específica de OmniUPF para cambiar entre modos.

Operación Simultánea:

- OmniUPF puede aceptar asociaciones PFCP de múltiples planos de control simultáneamente
- Una sola instancia de OmniUPF puede actuar como UPF, PGW-U y SGW-U **al mismo tiempo**
- Las sesiones de diferentes planos de control están aisladas y se gestionan de manera independiente

---

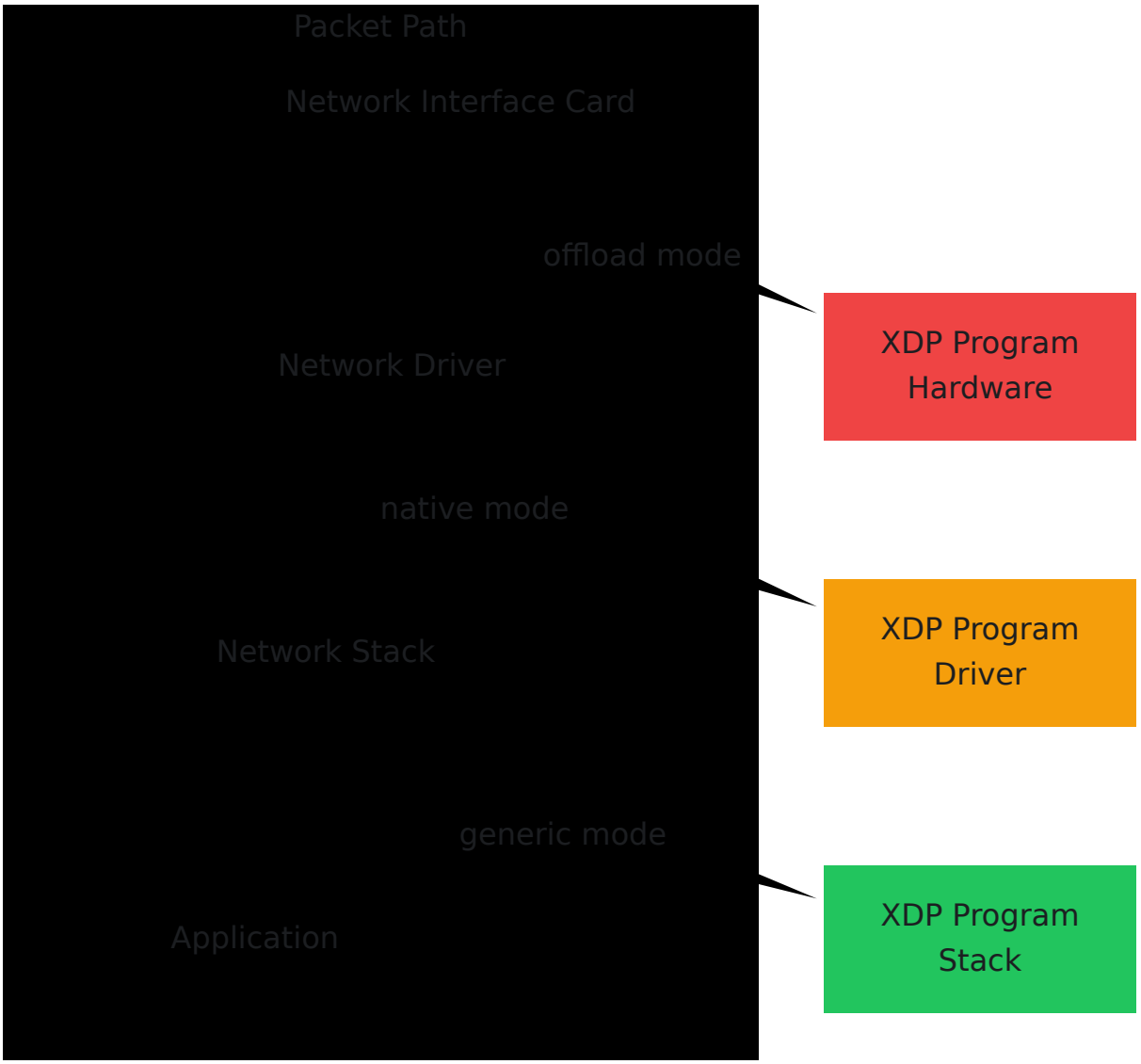
Modos de Adjunto XDP

OmniUPF utiliza XDP (eXpress Data Path) para el procesamiento de paquetes de alto rendimiento. Se admiten tres modos de adjunto.

Para instrucciones detalladas sobre la configuraci��n de XDP, especialmente para Proxmox y otros hipervisores, consulte la Guía de Modos XDP.

Comparación de Modos

Modo	Punto de Adjunto	Rendimiento	Caso de Uso	Requisitos de NIC
Genérico	Pila de red (núcleo)	~1-2 Mpps	Pruebas, desarrollo, compatibilidad	Cualquier NIC
Nativo	Controlador de red (núcleo)	~5-10 Mpps	Producción (bare metal, VM con SR-IOV)	Controlador compatible con XDP
Offload	Hardware de NIC (SmartNIC)	~10-40 Mpps	Producción de alto rendimiento	SmartNIC con descarga de XDP



Modo Genérico (Por Defecto)

Descripción: El programa XDP se ejecuta en la pila de red del núcleo

Ventajas:

- Funciona con **cualquier** interfaz de red
- Sin requisitos especiales de controlador o hardware
- Ideal para pruebas y desarrollo
- Compatible con todos los hipervisores y plataformas de virtualización

Desventajas:

- Rendimiento más bajo (~1-2 Mpps por núcleo)

- Los paquetes ya han pasado por el controlador antes del procesamiento de XDP

Configuración (en `runtime.exs`):

```
xdp_attach_mode = "generic"
```

Mejor para:

- Máquinas virtuales sin SR-IOV
- Entornos de pruebas y validación
- NICs sin soporte de controlador XDP
- Hipervisores como Proxmox, VMware, VirtualBox

Modo Nativo (Recomendado)

Descripción: El programa XDP se ejecuta a nivel del controlador de red

Ventajas:

- Alto rendimiento (~5-10 Mpps por núcleo)
- Paquetes procesados antes de entrar en la pila de red
- Latencia significativamente más baja que el modo genérico
- Funciona en bare metal y VMs con SR-IOV

Desventajas:

- Requiere un controlador de red con soporte para XDP
- No todos los NICs/controladores soportan XDP nativo

Configuración (en `runtime.exs`):

```
xdp_attach_mode = "native"
```

Mejor para:

- Despliegues de producción en bare metal
- VMs con paso de SR-IOV
- NICs con controladores compatibles con XDP (Intel, Mellanox, etc.)

Requisitos:

- Controlador de red compatible con XDP (ver Compatibilidad de NIC)
- Núcleo de Linux 5.15+ con soporte para XDP habilitado

Modo Offload (Rendimiento Máximo)

Descripción: El programa XDP se ejecuta directamente en el hardware de SmartNIC

Ventajas:

- Rendimiento máximo (~10-40 Mpps)
- Cero sobrecarga de CPU para el procesamiento de paquetes
- Latencia sub-microsegundo
- Libera CPU para el procesamiento del plano de control

Desventajas:

- Requiere hardware de SmartNIC costoso
- Disponibilidad limitada de SmartNIC
- Configuración y ajuste complejos

Configuración (en `runtime.exs`):

```
xdp_attach_mode = "offload"
```

Mejor para:

- Despliegues de producción de ultra-alto rendimiento
- Computación en el borde con requisitos de latencia estrictos
- Entornos donde los recursos de CPU son limitados

Requisitos:

- SmartNIC con soporte para descarga de XDP (Netronome Agilio CX, Mellanox BlueField)
- Firmware y controladores especializados

Parámetros de Configuración

Interfaces de Red

Parámetro	Descripción	Tipo	Predeterminado
`xdp_interfaces`	Interfaces de red para tráfico N3/N6/N9 (separadas por comas, o `"auto"` para todas las no-loopback)	Cadena	`"auto"`
`n3_address`	Dirección IPv4 para la interfaz N3 (GTP-U desde RAN)	IP	`"127.0.0.1"`
`n9_address`	Dirección IPv4 para la interfaz N9 (UPF-to-UPF para ULCL)	IP	Igual que `n3_address`

Ejemplo (en `runtime.exs`):

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

Configuración PFCP

Parámetro	Descripción	Tipo	Predeterminado
`pfcp_address`	Dirección local para el servidor PFCP (interfaz N4/Sxb/Sxc)	Cadena IP	`"127.0.0.1"`
`pfcp_port`	Puerto PFCP	Entero	`8805`
`node_id`	ID de Nodo Local para el protocolo PFCP	Cadena IP	`"127.0.0.1"`
`heartbeat_interval_ms`	Intervalo de latido PFCP (milisegundos)	Entero	`5000`
`heartbeat_timeout_ms`	Tiempo de espera del latido PFCP (milisegundos)	Entero	`5000`
`heartbeat_retries`	Número de reintentos de latido antes de declarar al par como muerto	Entero	`3`

Ejemplo (en `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 y Monitoreo

Parámetro	Descripción	Tipo	Predeterminado
`api_port`	Puerto local para el servidor REST API (Phoenix)	Entero	`8080`
`log_level`	Nivel de registro (`:debug`, `:info`, `:warning`, `:error`)	Átomo	`:info`

Ejemplo (en `runtime.exs`):

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

Gestión de Rutas GTP

Parámetro	Descripción	Tipo	Predeterminado
`gtpu_port`	Puerto GTP-U	Entero	`2152`
`gtp_echo_interval`	Intervalo de eco predeterminado en segundos	Entero	`10`
`gtp_echo_retries`	Máximo de ecos perdidos antes de fallar la ruta	Entero	`3`
`gtp_peers`	Lista de pares GTP para mantener vivos los ecos de solicitud	Lista	`[]`

Ejemplo (en `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},
]
```

Capacidad del Mapa eBPF

Parámetro	Descripción	Tipo	Predeterminado
`max_sessions`	Número máximo de sesiones concurrentes	Entero	`1_000_000`
`far_map_size`	Tamaño del mapa eBPF FAR	Entero	`131_070`
`qer_map_size`	Tamaño del mapa eBPF QER	Entero	`65_535`
`urr_map_size`	Tamaño del mapa eBPF URR	Entero	`65_535`

Ejemplo (en `runtime.exs`):

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

Configuración de Buffers

Parámetro	Descripción	Tipo	Predeterminado
`buffer_port`	Puerto UDP para paquetes en buffer desde eBPF	Entero	`22152`

Ejemplo (en `runtime.exs`):

```
buffer_port = 22152
```

Flags de Características

Parámetro	Descripción	Tipo	Predeterminado
`feature_ueip`	Habilitar la asignación de IP de UE por OmniUPF	Booleano	`false`
`ueip_pool`	Pool de IP para la asignación de IP de UE (requiere `feature_ueip`)	Cadena CIDR	`"10.60.0.0/24"`
`feature_ftup`	Habilitar la asignación de F-TEID por OmniUPF	Booleano	`true`
`teid_pool_start`	Inicio del rango de asignación de TEID	Entero	`1`
`teid_pool_end`	Fin del rango de asignación de TEID	Entero	`10_000_000`

Ejemplo (asignación de IP de UE) (en `runtime.exs`):

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

Ejemplo (asignación de F-TEID) (en `runtime.exs`):

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

Configuración del Gestor de Rutas

Para la sincronización de rutas de UE con el demonio FRR (Free Range Routing). Consulte la Guía de Gestión de Rutas para obtener detalles.

Parámetro	Descripción	Tipo	Predeterminado
`route_manager_enabled`	Habilitar la sincronización automática de rutas de UE	Booleano	`true`
`route_manager_type`	Tipo de demonio de enrutamiento (`"frr"` o `"static"`)	Cadena	`"frr"`

Ejemplo (en `runtime.exs`):

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

Cuándo Habilitar:

- Despliegues Multi-UPF que requieren publicidad de rutas
- Integración con protocolos de enrutamiento OSPF o BGP
- Requiere el demonio FRRouting instalado y configurado

Configuración eBPF / XDP

Parámetro	Descripción	Tipo	Predeterminado
`xdp_interfaces`	Lista separada por comas de interfaces para XDP, o `"auto"` para todas las no-loopback	Cadena	`"auto"`
`xdp_attach_mode`	Modo XDP: `"generic"`, `"native"` o `"offload"`	Cadena	`"generic"`
`ebpf_pin_path`	Ruta de pin del sistema de archivos BPF	Cadena	`"/sys/fs/bpf/upf_pipeline"`
`xdp_obj_path`	Ruta al objeto eBPF compilado	Cadena	`"/etc/omniupf/ipentrypoint_`

Ejemplo (en `runtime.exs`):

```
xdp_interfaces = "auto"              # o "eth0,eth1" para
interfaces específicas
xdp_attach_mode = "native"
ebpf_pin_path = "/sys/fs/bpf/upf_pipeline"
xdp_obj_path = "/etc/omniupf/ipentrypoint_bpf.o"
```

Archivo de Configuración

Archivo de Configuración de Elixir (runtime.exs)

Archivo: `/etc/omniupf/runtime.exs`

El archivo de configuración utiliza la sintaxis de Elixir con asignaciones de variables simples para facilitar la lectura. Las variables se aplican a la configuración de la aplicación en la parte inferior del archivo a través de `config :upf_ex, ...`.

```
import Config

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

#
============================================================================
# PFCP (N4/Sx) Configuración
#
============================================================================
pfcp_address = "10.100.50.241"       # Dirección IP para escuchar mer
PFCP
pfcp_port = 8805                      # Puerto PFCP (estándar: 8805)
node_id = "10.100.50.241"            # ID de Nodo publicitado en la
Configuración de Asociación

#
============================================================================
# Configuración del Plano de Datos GTP-U
#
============================================================================
n3_address = "10.100.50.233"         # IP de la interfaz N3 (GTP-U ha
n9_address = n3_address               # IP de la interfaz N9 (por def
N3)
gtpu_port = 2152                      # Puerto GTP-U (estándar: 2152)
buffer_port = 22152                   # Puerto del oyente de buffer

#
============================================================================
# Configuración eBPF / XDP
#
============================================================================
xdp_interfaces = "auto"              # "auto" para todas las no-loopb
lista separada por comas
xdp_attach_mode = "native"           # Modo XDP: "generic", "native"
"offload"
ebpf_pin_path = "/sys/fs/bpf/upf_pipeline"
xdp_obj_path = "/etc/omniupf/ipentrypoint_bpf.o"

#
============================================================================
```

```
# Configuración de Sesiones y Pool de Recursos
#
============================================================================
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

#
============================================================================
# Flags de Características
#
============================================================================
feature_ueip = true
feature_ftup = true
ueip_pool = "10.45.0.0/16"

#
============================================================================
# Gestión de Rutas
#
============================================================================
route_manager_enabled = true
route_manager_type = "frr"

#
============================================================================
# Configuración de Latidos
#
============================================================================
heartbeat_interval_ms = 10_000
heartbeat_timeout_ms = 5_000
heartbeat_retries = 5

#
============================================================================
# Monitoreo de Eco de Pares GTP-U
#
============================================================================
gtp_echo_interval = 10
gtp_echo_retries = 3
gtp_peers = [
```

```
  %{address: parse_ip.("10.100.50.50"), echo: true, echo_interval: 10
]

#
# ============================================================================
# API HTTP y Registro
#
# ============================================================================
api_port = 8080
log_level = :info

#
# ============================================================================
# Aplicar Configuración (no editar debajo de esta línea)
#
# ============================================================================
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
```

Gestión del Servicio

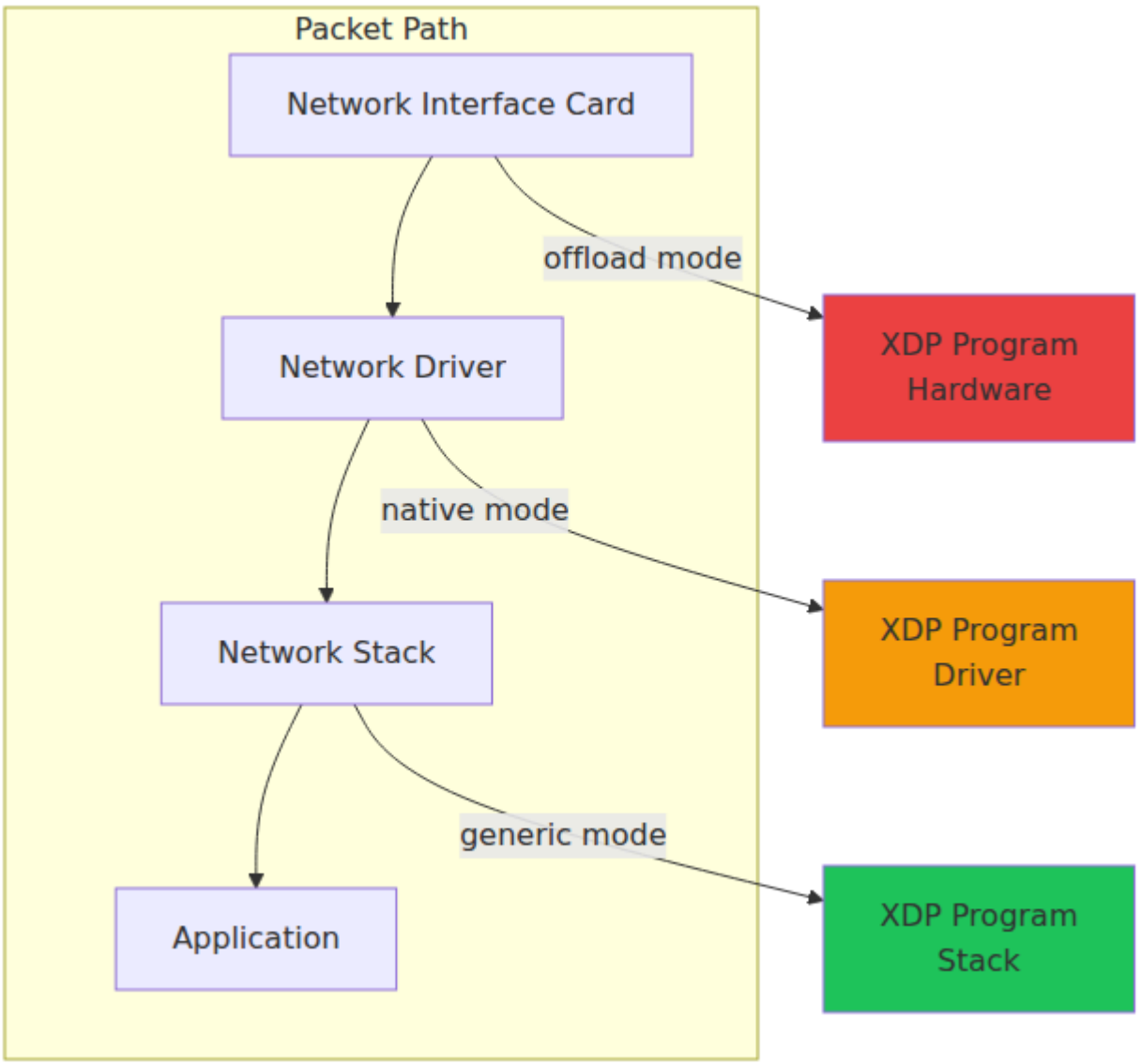
```
# Iniciar el UPF
sudo systemctl start omniupf

# Detener el UPF
sudo systemctl stop omniupf

# Reiniciar después de cambios de configuración
sudo systemctl restart omniupf

# Verificar estado
sudo systemctl status omniupf

# Ver logs
sudo journalctl -u omniupf -f

# O use el binario de lanzamiento directamente
sudo /opt/omniupf/bin/upf_ex start
sudo /opt/omniupf/bin/upf_ex stop
sudo /opt/omniupf/bin/upf_ex daemon   # iniciar como demonio en
segundo plano
```

Compatibilidad con Hypervisores

Descripción General

OmniUPF es compatible con todos los principales hipervisores y plataformas de virtualización. El modo de adjunto XDP y la configuración de red dependen de las capacidades de red del hipervisor.

Para instrucciones paso a paso sobre cómo habilitar XDP nativo en Proxmox y otros hipervisores, consulte la Guía de Modos XDP.



Proxmox VE

Configuraciones Soportadas:

1. Modo Puente (XDP Genérico)

Caso de uso: Redes estándar de VM

Configuración:

- Dispositivo de Red: VirtIO o E1000

- Modo XDP: `generic`
- Rendimiento: ~1-2 Mpps

Configuraciones de VM de Proxmox:

```
Dispositivo de Red: net0
Modelo: VirtIO (paravirtualizado)
Puente: vmbr0
```

Configuración de OmniUPF (en `runtime.exs`):

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

---

2. Passthrough SR-IOV (XDP Nativo)

Caso de uso: Producción de alto rendimiento

Configuración:

- Dispositivo de Red: Función Virtual SR-IOV
- Modo XDP: `native`
- Rendimiento: ~5-10 Mpps

Requisitos:

- NIC física con soporte para SR-IOV (Intel X710, Mellanox ConnectX-5)
- SR-IOV habilitado en BIOS
- IOMMU habilitado (`intel_iommu=on` o `amd_iommu=on` en GRUB)

Habilitar SR-IOV en Proxmox:

```
# Editar la configuración de GRUB
nano /etc/default/grub

# Agregar a GRUB_CMDLINE_LINUX_DEFAULT:
intel_iommu=on iommu=pt

# Actualizar GRUB y reiniciar
update-grub
reboot

# Habilitar VFs en NIC (ejemplo: 4 funciones virtuales en eth0)
echo 4 > /sys/class/net/eth0/device/sriov_numvfs

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

Configuraciones de VM de Proxmox:

```
Hardware -> Agregar -> Dispositivo PCI
Seleccionar: Función Virtual SR-IOV
Todas las Funciones: No
GPU Principal: No
PCI-Express: Sí (opcional)
```

Configuración de OmniUPF (en `runtime.exs`):

```
xdp_interfaces = "ens1f0"    # Nombre de VF SR-IOV
xdp_attach_mode = "native"
```

---

3. Passthrough PCI (XDP Nativo)

Caso de uso: NIC dedicada para una sola VM

Configuración:

- Todo el NIC físico pasado a la VM

- Modo XDP: `native` u `offload` (si es SmartNIC)
- Rendimiento: ~5-40 Mpps (depende del NIC)

Configuraciones de VM de Proxmox:

```
Hardware -> Agregar -> Dispositivo PCI
Seleccionar: NIC Física (ejemplo, 0000:01:00.0)
Todas las Funciones: Sí
GPU Principal: No
PCI-Express: Sí
```

Configuración de OmniUPF (en `runtime.exs`):

```
xdp_interfaces = "ens1f0"
xdp_attach_mode = "native"    # o "offload" para SmartNIC
```

KVM/QEMU

Modo Puente:

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

Passthrough SR-IOV:

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

VMware ESXi

vSwitch Estándar (XDP Genérico):

- Adaptador de Red: VMXNET3
- Modo XDP: `generic`

SR-IOV (XDP Nativo):

- Habilitar SR-IOV en la configuración del host ESXi
- Agregar adaptador de red SR-IOV a la VM
- Modo XDP: `native`

Microsoft Hyper-V

Interruptor Virtual (XDP Genérico):

- Adaptador de Red: Sintético
- Modo XDP: `generic`

SR-IOV (XDP Nativo):

- Habilitar SR-IOV en el Administrador de Hyper-V
- Configurar SR-IOV en el adaptador de red virtual
- Modo XDP: `native`

VirtualBox

Modo NAT/Puente (solo XDP Genérico):

- Adaptador de Red: VirtIO-Net o Intel PRO/1000
- Modo XDP: `generic`
- Nota: VirtualBox **no** soporta SR-IOV

Compatibilidad con NIC

Entendiendo Mpps vs Rendimiento

Los paquetes por segundo (Mpps) y el rendimiento (Gbps) no son equivalentes directamente - la relación depende completamente del tamaño del paquete. El tráfico de red móvil varía drásticamente en tamaño de paquete, desde pequeños paquetes de VoIP hasta grandes tramas de transmisión de video.

Impacto del Tamaño del Paquete en el Rendimiento

En las redes móviles, el UPF procesa paquetes encapsulados GTP-U en la interfaz N3 y paquetes IP nativos en la interfaz N6.

Sobrecarga de Encapsulación GTP-U (Interfaz N3):

- **Encabezado IPv4 externo**: 20 bytes
- **Encabezado UDP externo**: 8 bytes
- **Encabezado GTP-U**: 8 bytes
- **Total de sobrecarga GTP-U**: 36 bytes

Paquete GTP-U Mínimo (N3):

- **Encabezado IP interno**: 20 bytes (IPv4)
- **Encabezado UDP interno**: 8 bytes
- **Carga útil mínima**: 1 byte
- **Total del paquete interno**: 29 bytes
- **Más sobrecarga GTP-U**: 36 bytes
- **Tamaño total del paquete**: 65 bytes

Rendimiento a 1 Mpps con paquetes GTP-U mínimos:

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

Paquete GTP-U Máximo (N3 con MTU de 1500):

- **MTU IP interno**: 1500 bytes (paquete IP interno completo)
- **Más sobrecarga GTP-U**: 36 bytes
- **Tamaño total del paquete**: 1536 bytes

Rendimiento a 1 Mpps con paquetes GTP-U máximos:

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

Paquetes IP Nativos (Interfaz N6):

En N6 (hacia Internet), los paquetes son IP nativos sin GTP-U:

Paquete N6 Mínimo:

- **Encabezado IP**: 20 bytes
- **Encabezado UDP**: 8 bytes
- **Carga útil mínima**: 1 byte
- **Total**: 29 bytes

Rendimiento a 1 Mpps con paquetes N6 mínimos:

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

Paquete N6 Máximo (MTU de 1500):

- **MTU IP**: 1500 bytes
- **Total**: 1500 bytes

Rendimiento a 1 Mpps con paquetes N6 máximos:

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

Ejemplos de Rendimiento en el Mundo Real

NIC Intel X710 (capacidad de 10 Mpps en la interfaz N3 con GTP-U):

Patrón de Tráfico	Tamaño de Paquete Interno	Total GTP-U	Rendimiento a 10 Mpps	Caso de Uso Típico
Llamadas VoIP (N3)	65-150 bytes	101-186 bytes	**0.8-1.5 Gbps**	Voz AMR-WB, G.711
Web ligera (N3)	400-600 bytes	436-636 bytes	**3.5-5.1 Gbps**	HTTP/HTTPS, mensajería
Móvil moderno (N3)	**1200 bytes**	**1236 bytes**	**9.9 Gbps**	**Mezcla de tráfico típica 2024**
Transmisión de video (N3)	1400-1450 bytes	1436-1486 bytes	**11.5-11.9 Gbps**	Fragmentos de video HD/4K
MTU máximo (N3)	1500 bytes	1536 bytes	**12.3 Gbps**	Descargas de TCP grandes

En la interfaz N6 (IP nativa, sin GTP-U):

Patrón de Tráfico	Tamaño de Paquete	Rendimiento a 10 Mpps	Caso de Uso Típico
Paquetes VoIP	65-150 bytes	**0.5-1.2 Gbps**	Flujos RTP de voz
Web ligera	400-600 bytes	**3.2-4.8 Gbps**	Solicitudes HTTP
Móvil moderno	**1200 bytes**	**9.6 Gbps**	**Tráfico típico 2024**
Transmisión de video	1400-1450 bytes	**11.2-11.6 Gbps**	Descargas de video
MTU máximo	1500 bytes	**12.0 Gbps**	Transferencias de archivos grandes

A 10 Mpps con tráfico móvil moderno (promedio de 1200 bytes), se espera un **rendimiento de ~10 Gbps** en ambas interfaces N3 y N6.

Planificación de Capacidad Práctica:

Con un tamaño de paquete promedio de 1200 bytes (típico para redes móviles modernas con transmisión de video):

Capacidad Mpps de NIC	Rendimiento N3 (GTP-U)	Rendimiento N6 (IP Nativa)	Despliegue Realista
1 Mpps	~1.0 Gbps	~1.0 Gbps	Sitio de pequeña celda, puerta de enlace IoT
5 Mpps	~4.9 Gbps	~4.8 Gbps	Sitio de celda mediana, empresa
10 Mpps	~9.9 Gbps	~9.6 Gbps	Sitio de celda grande, pequeña ciudad
20 Mpps	~19.7 Gbps	~19.2 Gbps	Área metropolitana, ciudad mediana
40 Mpps	~39.4 Gbps	~38.4 Gbps	Gran metrópoli, centro regional

---

Controladores de Red Compatibles con XDP

OmniUPF requiere controladores de red con soporte para XDP para los modos **nativo** y **offload**. El modo genérico funciona con **cualquier** NIC.

NICs Intel

Modelo	Controlador	Soporte XDP	Modo	Rendimiento
Intel X710	i40e	Sí	Nativo	~10 Mpps
Intel XL710	i40e	Sí	Nativo	~10 Mpps
Intel E810	ice	Sí	Nativo	~15 Mpps
Intel 82599ES	ixgbe	Sí	Nativo	~8 Mpps
Intel I350	igb	Limitado	Genérico	~1 Mpps
Intel E1000	e1000	No	Solo Genérico	~1 Mpps

NICs Mellanox/NVIDIA

Modelo	Controlador	Soporte XDP	Modo	Rendimiento
Mellanox ConnectX-5	mlx5	Sí	Nativo	~12 Mpps
Mellanox ConnectX-6	mlx5	Sí	Nativo	~20 Mpps
Mellanox BlueField	mlx5	Sí	Nativo + Offload	~40 Mpps
Mellanox ConnectX-4	mlx4	Limitado	Genérico	~2 Mpps

NICs Broadcom

Modelo	Controlador	Soporte XDP	Modo	Rendimiento
Broadcom BCM57xxx	bnxt_en	Sí	Nativo	~8 Mpps
Broadcom NetXtreme II	bnx2x	No	Solo Genérico	~1 Mpps

Otros Proveedores

Modelo	Controlador	Soporte XDP	Modo	Rendimiento
Netronome Agilio CX	nfp	Sí	Offload	~30 Mpps
Amazon ENA	ena	Sí	Nativo	~5 Mpps
Solarflare SFC9xxx	sfc	Sí	Nativo	~8 Mpps
VirtIO	virtio_net	Limitado	Genérico	~2 Mpps

Verificando el Soporte XDP de NIC

Verifique si el controlador soporta XDP:

```
# Encontrar el controlador de NIC
ethtool -i eth0 | grep driver

# Verificar el soporte de XDP en el controlador
modinfo <driver_name> | grep -i xdp

# Ejemplo para Intel i40e
modinfo i40e | grep -i xdp
```

Verificar la adjunción del programa XDP:

```
# Comprobar si el programa XDP está adjunto
ip link show eth0 | grep -i xdp

# Salida de ejemplo (XDP adjunto):
# 2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 xdp qdisc mq
```

NICs Recomendadas por Caso de Uso

Con un tamaño de paquete promedio de 1200 bytes (tráfico móvil moderno):

Caso de Uso	NIC Recomendada	Modo	Capacidad Mpps	Rendi (M
Pruebas/Desarrollo	Cualquier NIC (VirtIO, E1000)	Genérico	1-2 Mpps	1-2 Gb
Sitio de Pequeña Celda	Intel X710, Mellanox CX-5	Nativo	5-10 Mpps	5-10 G
Celda Media/Metropolitana	Intel E810, Mellanox CX-6	Nativo	10-20 Mpps	10-20
Gran Metropolitana	Mellanox CX-6, Intel E810 (dual)	Nativo	20-40 Mpps	20-40
Centro Regional	Mellanox BlueField, Netronome Agilio	Offload	40+ Mpps	40+ G
VM de Proxmox (Puente)	VirtIO	Genérico	1-2 Mpps	1-2 Gb
VM de Proxmox (SR-IOV)	Intel X710/E810 VF, Mellanox CX-5 VF	Nativo	5-10 Mpps	5-10 G

Recursos Adicionales

Documentación Oficial de XDP:

- Proyecto XDP
- Documentación de XDP del Núcleo

Listas de Compatibilidad de NIC:

- Compatibilidad de Hardware de Cilium XDP
- Controladores XDP de IO Visor

Ejemplos de Configuración

Ejemplo 1: Entorno de Desarrollo (Modo Genérico)

Escenario: Pruebas de OmniUPF en laptop o VM sin SR-IOV

```
# Configuración de desarrollo (/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
```

Ejemplo 2: Producción Bare Metal (Modo Nativo)

Escenario: UPF de producción en servidor bare metal con NIC Intel X710

```
# Configuración de producción bare metal
(/etc/omniupf/runtime.exs)
xdp_interfaces = "ens1f0,ens1f1"    # N3 en ens1f0, N6 en 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"
```

Ejemplo 3: VM de Proxmox con SR-IOV (Modo Nativo)

Escenario: UPF de producción en VM de Proxmox con passthrough SR-IOV

```
# Configuración de Proxmox SR-IOV (/etc/omniupf/runtime.exs)
xdp_interfaces = "ens1f0"           # VF de 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},
]
```

Ejemplo 4: Modo PGW-U (EPC 4G)

Escenario: OmniUPF actuando como PGW-U en una red EPC 4G

```
# Configuración 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"          # Interfaz 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
```

Ejemplo 5: Múltiples Modos (UPF + PGW-U Simultáneamente)

Escenario: OmniUPF sirviendo tanto redes 5G como 4G simultáneamente

```
# Configuración de múltiples modos (/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"
```

Ejemplo 6: Modo Offload de SmartNIC

Escenario: Despliegue de ultra-alto rendimiento con SmartNIC Netronome Agilio CX

```
# Configuración de SmartNIC offload (/etc/omniupf/runtime.exs)
xdp_interfaces = "enp1s0np0"        # Interfaz 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
```

Dimensionamiento de Mapas y Planificación de Capacidad

Estableciendo Tamaños de Mapas

Establezca `max_sessions` y configure los tamaños de mapa para su despliegue:

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

Uso de memoria: ~91 MB para 100K sesiones

Estimación de Capacidad

Calcule el número máximo de sesiones:

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

Ejemplo:

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

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

Requisitos de Memoria

Uso de memoria por sesión:

- 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
- **Total**: ~540 B por sesión

Para 100K sesiones: ~52 MB de memoria del núcleo

Recomendación: Asegúrese de que el límite de memoria bloqueada permita 2x el uso estimado:

```
# Comprobar límite actual
ulimit -l

# Establecer ilimitado (requerido para eBPF)
ulimit -l unlimited
```

Documentación Relacionada

- **Guía de Arquitectura** - Detalles técnicos de eBPF/XDP y optimización de rendimiento
- **Guía de Gestión de Reglas** - Configuración de PDR, FAR, QER, URR
- **Guía de Monitoreo** - Estadísticas, monitoreo de capacidad y alertas
- **Referencia de Métricas** - Referencia completa de métricas de Prometheus
- **Guía de Interfaz Web** - Operaciones del panel de control
- **Guía de Operaciones** - Descripción general de la arquitectura y despliegue de UPF
- **Guía de Jardín Amurallado** - Redirección fuera de crédito, portal cautivo y configuración de lista blanca

Referencia de Métricas

Este documento describe todas las métricas de Prometheus expuestas por OmniUPF en el endpoint `/metrics`.

Categorías de Métricas

1. **Métricas de mensajes PFCP** - Contadores de mensajes del protocolo de plano de control y latencia por par
2. **Métricas de acción XDP** - Veredictos de paquetes del plano de datos (descartar, pasar, redirigir, etc.)
3. **Métricas de paquetes** - Contadores de paquetes recibidos por tipo de protocolo
4. **Métricas de sesión y asociación PFCP** - Conteos de sesiones y asociaciones por par
5. **Métricas URR** - Contadores de volumen de tráfico agregados por par PFCP
6. **Métricas de almacenamiento en búfer de paquetes** - Estado del búfer de paquetes, capacidad y rendimiento
7. **Métricas de informe de datos de enlace descendente (notificación)** - Notificaciones de solicitud de informe de sesión PFCP y seguimiento del índice FAR
8. **Métricas de capacidad del mapa eBPF** - Utilización y capacidad del mapa eBPF

Referencia de Métricas

Métricas de mensajes PFCP

Métricas para rastrear mensajes del protocolo PFCP entre el UPF y los nodos del plano de control.

Nombre de la Métrica	Tipo	Etiquetas	Descripción
upf_pfcp_rx	Counter	`message_name`, `peer_address`	Número total de mensajes PFCP recibidos por tipo de mensaje y par
upf_pfcp_tx	Counter	`message_name`, `peer_address`	Número total de mensajes PFCP transmitidos por tipo de mensaje y par
upf_pfcp_rx_errors	Counter	`message_name`, `cause_code`, `peer_address`	Número total de mensajes PFCP rechazados con causa de error por tipo de mensaje y par
upf_pfcp_rx_latency	Summary	`message_type`, `peer_address`	Duración del procesamiento de mensajes PFCP en microsegundos (cuantiles p50, p90, p99) por tipo de mensaje y par

Nota: Todos los contadores rastrean mensajes por par PFCP para una visibilidad granular del comportamiento del nodo del plano de control.

Métricas de acción XDP

Contadores de paquetes por acción/veredicto del programa XDP. Estas métricas rastrean la decisión del plano de datos para cada paquete.

Nombre de la Métrica	Tipo	Etiquetas	Descripción
upf_xdp_aborted	Counter	none	Número total de paquetes abortados (XDP_ABORTED)
upf_xdp_drop	Counter	none	Número total de paquetes descartados (XDP_DROP)
upf_xdp_pass	Counter	none	Número total de paquetes pasados al kernel (XDP_PASS)
upf_xdp_tx	Counter	none	Número total de paquetes transmitidos (XDP_TX)
upf_xdp_redirect	Counter	none	Número total de paquetes redirigidos (XDP_REDIRECT)

Métricas de paquetes

Contadores para paquetes recibidos por tipo de protocolo. Todas las métricas utilizan la etiqueta `packet_type`.

Nombre de la Métrica	Tipo	Etiquetas	Descripción
upf_rx	Counter	`packet_type`	Número total de paquetes recibidos por tipo
upf_route	Counter	`packet_type`	Número total de paquetes enrutados por resultado de búsqueda

Valores de packet_type de upf_rx:

- `arp` - Paquetes ARP
- `icmp` - Paquetes ICMP
- `icmp6` - Paquetes ICMPv6
- `ip4` - Paquetes IPv4
- `ip6` - Paquetes IPv6
- `tcp` - Paquetes TCP
- `udp` - Paquetes UDP
- `other` - Otros tipos de paquetes
- `gtp-echo` - Solicitud/respuesta de eco GTP
- `gtp-pdu` - PDU GTP-U (datos de usuario encapsulados)
- `gtp-other` - Otros tipos de mensajes GTP
- `gtp-unexp` - Paquetes GTP inesperados/malformados

Valores de packet_type de upf_route:

- `ip4-cache` - Acertijos de caché de ruta IPv4
- `ip4-ok` - Éxito en la búsqueda de FIB IPv4
- `ip4-error-drop` - Fallo en la búsqueda de FIB IPv4, paquete descartado
- `ip4-error-pass` - Fallo en la búsqueda de FIB IPv4, paquete pasado al kernel
- `ip6-cache` - Acertijos de caché de ruta IPv6
- `ip6-ok` - Éxito en la búsqueda de FIB IPv6
- `ip6-error-drop` - Fallo en la búsqueda de FIB IPv6, paquete descartado
- `ip6-error-pass` - Fallo en la búsqueda de FIB IPv6, paquete pasado al kernel

Métricas de sesión y asociación PFCP

Métricas para rastrear sesiones y asociaciones PFCP entre el UPF y los nodos del plano de control.

Nombre de la Métrica	Tipo	Etiquetas	Descripción
upf_pfcp_sessions	Gauge	none	Número total de sesiones PFCP actualmente establecidas (todos los pares)
upf_pfcp_associations	Gauge	none	Número total de asociaciones PFCP actualmente establecidas (todos los pares)
upf_pfcp_association_status	Gauge	`node_id`, `address`	Estado de la asociación PFCP por par (1=activo, 0=inactivo)
upf_pfcp_sessions_per_node	Gauge	`node_id`, `address`	Número de sesiones PFCP activas por nodo del plano de control

Métricas URR (Regla de Informe de Uso)

Métricas de volumen de tráfico agregadas por par PFCP. El volumen de cada par representa la suma de todos los contadores URR a través de todas las sesiones de ese nodo del plano de control.

Nombre de la Métrica	Tipo	Etiquetas	Descripción
upf_urr_uplink_volume_bytes	Gauge	`peer_address`	Volumen total de tráfico de enlace ascendente en bytes para todas las sesiones de este par
upf_urr_downlink_volume_bytes	Gauge	`peer_address`	Volumen total de tráfico de enlace descendente en bytes para todas las sesiones de este par
upf_urr_total_volume_bytes	Gauge	`peer_address`	Volumen total de tráfico en bytes (ascendente + descendente) para todas las sesiones de este par

Nota: Los volúmenes se agregan por par PFCP para evitar problemas de alta cardinalidad. Las estadísticas individuales de URR están disponibles a través de la API REST en `/api/v1/urr_map`.

Métricas de almacenamiento en búfer de paquetes

Métricas para rastrear el estado y rendimiento del búfer de paquetes. El UPF puede almacenar en búfer paquetes de enlace descendente cuando un UE está en estado inactivo, manteniéndolos hasta que el UE sea paginado y transicione a estado conectado.

Nombre de la Métrica	Tipo	Etiquetas	
upf_buffer_packets_total	Counter	none	N o a b t
upf_buffer_packets_dropped	Counter	`reason`	N o o o
upf_buffer_packets_flushed	Counter	none	N o v b
upf_buffer_packets_current	Gauge	none	N o e
upf_buffer_bytes_total	Counter	none	T a b t
upf_buffer_bytes_current	Gauge	none	B e
upf_buffer_fars_active	Gauge	none	N o p b
upf_buffer_listener_packets_received_total	Counter	none	T p

Nombre de la Métrica	Tipo	Etiquetas	
			r e b p e
upf_buffer_listener_packets_buffered_total	Counter	none	T p a c e
upf_buffer_listener_errors_total	Counter	`type`	E p c c b
upf_buffer_listener_error_indications_sent_total	Counter	`remote_peer`	T n I E e T c p r
upf_buffer_flush_success_total	Counter	none	T c c b
upf_buffer_flush_errors_total	Counter	`reason`	T c

Nombre de la Métrica	Tipo	Etiquetas	
			c b
upf_buffer_flush_packets_sent_total	Counter	none	T p e c c c

Valores de razón de upf_buffer_packets_dropped:

- `expired` - Paquetes descartados debido a la expiración de TTL
- `global_limit` - Descartados debido a que se alcanzó el límite total del búfer
- `far_limit` - Descartados debido a que se alcanzó el límite del búfer por FAR
- `cleared` - Paquetes eliminados manualmente del búfer

Valores de tipo de upf_buffer_listener_errors_total:

- `read_error` - Error al leer del socket del búfer
- `too_small` - Paquete demasiado pequeño para el encabezado GTP
- `invalid_gtp_type` - Tipo de mensaje GTP no G-PDU
- `unknown_teid` - No se encontró PDR/FAR para TEID
- `not_buffering_far` - FAR no tiene acción BUFF
- `truncated_ext` - Encabezados de extensión GTP truncados
- `no_payload` - El paquete GTP no tiene carga útil
- `buffer_full` - Capacidad del búfer excedida

Valores de razón de upf_buffer_flush_errors_total:

- `far_lookup_failed` - Falló al buscar información de FAR en el mapa eBPF
- `no_forw_action` - FAR no tiene acción FORW establecida
- `connection_failed` - Falló al crear conexión UDP para el vaciado

Métricas de informe de datos de enlace descendente (notificación)

Métricas para las notificaciones de solicitud de informe de sesión PFCP enviadas al plano de control cuando los paquetes están en búfer. Estas notificaciones desencadenan que el plano de control pagine al UE.

Nombre de la Métrica	Tipo	Etiquetas	Des
upf_dldr_sent_total	Counter	none	Núm de noti de I de D Enla Des (DLI env SMF
upf_dldr_send_errors	Counter	none	Núm de e env noti de I de D Enla Des
upf_dldr_active_notifications	Gauge	none	Núm actu FAR noti DLD (no elim
upf_far_index_size	Gauge	none	Núm actu FAR regi en F para

Nombre de la Métrica	Tipo	Etiquetas	Des
			noti DLD
upf_far_index_registrations_total	Counter	none	Núm de r de F FarI
upf_far_index_unregistrations_total	Counter	none	Núm de des de F FarI
upf_buffer_notify_to_flush_duration_seconds	Histogram	`pfcp_peer`	Tier el e noti DLD vaci los en b

upf_buffer_notify_to_flush_duration_seconds:

- Cubos de histograma: 0.01, 0.05, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0, 30.0, 60.0 segundos
- Etiqueta `pfcp_peer`: dirección SMF/PGW-C (por ejemplo, `10.100.50.241`)
- Mide la latencia entre el envío de la notificación por parte del UPF al SMF y la respuesta del SMF con la modificación de sesión para vaciar paquetes
- Útil para monitorear la capacidad de respuesta del plano de control durante las transiciones de inactivo a conectado

Métricas de Indicación de Error GTP-U

Métricas para rastrear los mensajes de Indicación de Error GTP-U enviados y recibidos. Las Indicación de Error se envían cuando un par recibe paquetes para TEIDs desconocidos, indicando desajustes en el estado del túnel (a menudo debido a reinicios de pares).

Nombre de la Métrica	Tipo	Etic
upf_buffer_listener_error_indications_sent_total	Counter	`node_` `peer_`
upf_buffer_listener_error_indications_received_total	Counter	`node_` `peer_`
upf_buffer_listener_error_indication_sessions_deleted_total	Counter	`node_` `peer_`

Definiciones de Etiquetas:

- `node_id`: ID de nodo PFCP de la asociación (por ejemplo, `"pgw-u-1"`, `"smf-1"`). Se establece en `"unknown"` si no existe asociación PFCP para ese par.
- `peer_address`: dirección IP del par remoto (por ejemplo, `"192.168.50.10"`)

Cuándo se Envía la Indicación de Error:

- El UPF recibe un paquete GTP-U para un TEID que no existe (por ejemplo, después de un reinicio del UPF, la sesión ya fue eliminada)
- El remitente (eNodeB, gNodeB, UPF ascendente) está reenviando a un túnel obsoleto/eliminado
- El UPF envía la Indicación de Error para informar al remitente que deje de enviar

Cuándo se Recibe la Indicación de Error:

- El UPF reenvía un paquete GTP-U a un par descendente (PGW-U, SGW-U, UPF) para un TEID desconocido
- El par remoto no reconoce el TEID de destino (por ejemplo, el par se reinició y perdió el estado del túnel)
- El UPF elimina automáticamente las sesiones afectadas para dejar de reenviar a túneles muertos

Casos de Uso:

- Detectar reinicios de pares (una alta tasa de Indicación de Error indica pérdida de estado)
- Identificar desajustes de configuración (problemas de asignación de TEID)
- Monitorear la salud de sincronización de túneles entre elementos de red
- Alertar sobre eliminaciones inesperadas de sesiones

Ejemplos de Consultas PromQL:

```
# Tasa de Indicación de Error recibidas por par (por segundo)
rate(upf_buffer_listener_error_indications_received_total[5m])

# Total de sesiones eliminadas debido a Indicación de Error de un par
upf_buffer_listener_error_indication_sessions_deleted_total{peer_addr

# Pares enviando TEIDs desconocidos a este UPF
sum by (node_id, peer_address) (upf_buffer_listener_error_indications
```

Métricas de capacidad del mapa eBPF

Métricas para rastrear la utilización del mapa eBPF. Estas métricas ayudan a monitorear el uso de recursos y detectar posibles problemas de capacidad.

Nombre de la Métrica	Tipo	Etiquetas	Descripción
upf_ebpf_map_capacity	Gauge	`map_name`	Capacidad máxima del mapa eBPF
upf_ebpf_map_used	Gauge	`map_name`	Número actual de entradas en el mapa eBPF

Valores comunes de map_name:

- `pdr_map` - Mapa de Reglas de Detección de Paquetes
- `far_map` - Mapa de Reglas de Acción de Reenvío
- `qer_map` - Mapa de Reglas de Aplicación de QoS
- `session_map` - Mapa de búsqueda de sesiones
- `teid_map` - Mapeo de TEID a sesión
- `ue_ip_map` - Mapeo de dirección IP de UE a sesión

Uso de Métricas de Prometheus

Accediendo a Métricas

Las métricas están expuestas en el endpoint `/metrics` en la dirección especificada por `metrics_address` en el archivo de configuración (por defecto `:9090`):

```
# Ver métricas en bruto
curl http://localhost:9090/metrics

# Ejemplo de salida
upf_pfcp_sessions 42
upf_pfcp_associations 2
upf_urr_total_volume_bytes{peer_address="10.100.50.241"}
1048576000
```

Configuración de Prometheus

Agrega el objetivo OmniUPF a tu `prometheus.yml`:

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

Dashboards de Grafana

Importa métricas en Grafana para visualización:

- Conteos y tendencias de sesiones
- Volumen de tráfico por par PFCP
- Tasas de procesamiento de paquetes
- Utilización de búfer
- Monitoreo de capacidad del mapa eBPF

Documentación Relacionada

- **Guía de Monitoreo** - Monitoreo de estadísticas, planificación de capacidad y alertas
- **Guía de Configuración** - Configurar `metrics_address` y otras opciones de UPF
- **Guía de Interfaz Web** - Ver métricas en la página de Estadísticas

- **Guía de Arquitectura** - Ruta de datos eBPF y optimización del rendimiento
- **Guía de Gestión de Reglas** - Comprensión de métricas PDR, FAR, QER, URR
- **Guía de Solución de Problemas** - Uso de métricas para diagnósticos

Guía de Monitoreo

Tabla de Contenidos

1. Descripción General
2. Monitoreo de Estadísticas
3. Monitoreo de Capacidad
4. Métricas de Rendimiento
5. Alertas y Umbrales
6. Planificación de Capacidad
7. Resolución de Problemas de Rendimiento

Descripción General

El monitoreo efectivo de OmniUPF es crítico para mantener la calidad del servicio, prevenir el agotamiento de capacidad y resolver problemas de rendimiento. OmniUPF proporciona métricas completas en tiempo real a través de su interfaz web y API REST.

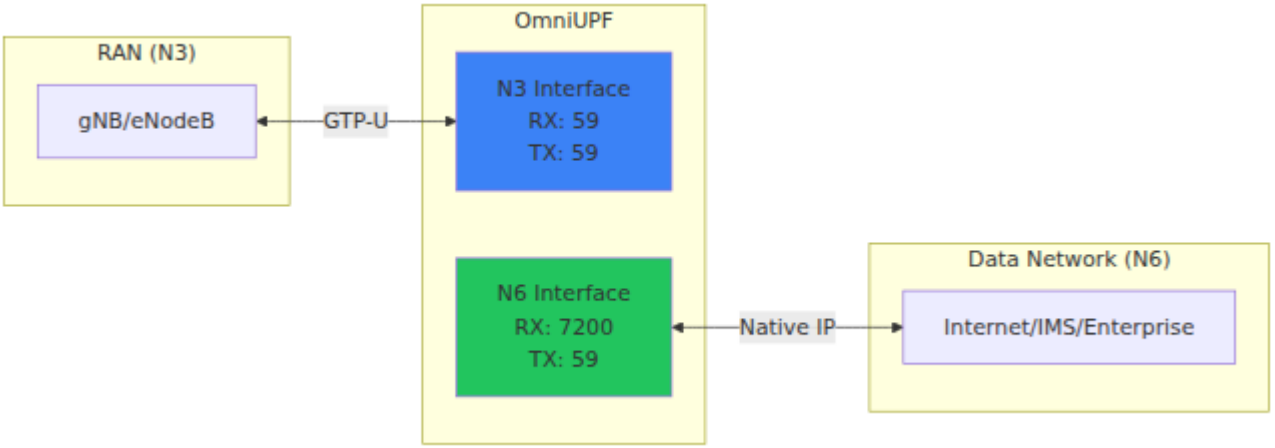
Categorías de Monitoreo

Categoría	Propósito	Frecuencia de Actualización	Métricas Clave
Estadísticas de Paquetes	Rastrear tasas de procesamiento de paquetes y errores	En tiempo real	Paquetes RX/TX, caídas, desglose de protocolos
Estadísticas de Interfaz	Monitorear distribución de tráfico N3/N6	En tiempo real	N3 RX/TX, N6 RX/TX
Estadísticas de XDP	Rastrear rendimiento del datapath del kernel	En tiempo real	XDP procesados, pasados, descartados, abortados
Estadísticas de Rutas	Monitorear decisiones de enrutamiento de paquetes	En tiempo real	Búsquedas FIB, aciertos/fallos de caché
Capacidad del Mapa eBPF	Prevenir agotamiento de recursos	Cada 10s	Porcentajes de uso del mapa, usado vs. capacidad
Estadísticas de Buffers	Rastrear el almacenamiento de paquetes durante la movilidad	Cada 5s	Paquetes en buffer, edad del buffer, conteo de FAR

Monitoreo de Estadísticas

Estadísticas de Interfaz N3/N6

Las estadísticas de la interfaz N3/N6 proporcionan visibilidad sobre la distribución del tráfico entre la RAN (N3) y la Red de Datos (N6).



Métricas:

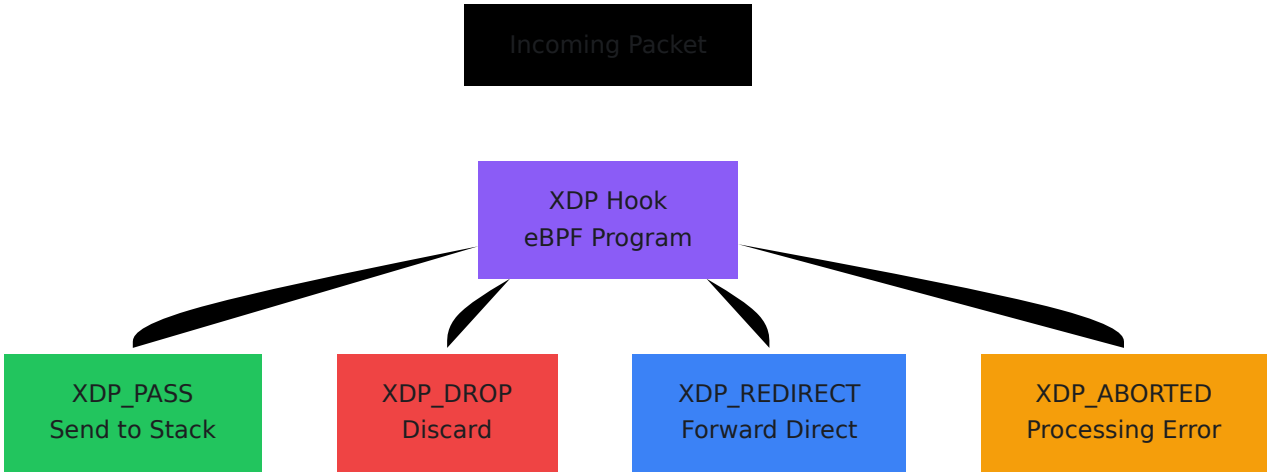
- **RX N3**: Paquetes recibidos de RAN (tráfico GTP-U de enlace ascendente)
- **TX N3**: Paquetes transmitidos a RAN (tráfico GTP-U de enlace descendente)
- **RX N6**: Paquetes recibidos de la Red de Datos (IP nativa de enlace descendente)
- **TX N6**: Paquetes transmitidos a la Red de Datos (IP nativa de enlace ascendente)
- **Total**: Conteo agregado de paquetes en todas las interfaces

Comportamiento Esperado:

- **RX N3 ≈ TX N6**: Los paquetes de enlace ascendente fluyen de RAN a la Red de Datos
- **RX N6 ≈ TX N3**: Los paquetes de enlace descendente fluyen de la Red de Datos a RAN
- Un desequilibrio significativo puede indicar:
  - Tráfico asimétrico (descargas >> cargas)
  - Caídas de paquetes o errores de reenvío
  - Errores de configuración de enrutamiento

Estadísticas de XDP

Las estadísticas de XDP (eXpress Data Path) muestran el rendimiento del procesamiento de paquetes a nivel de kernel.



Métricas:

- **Abortado**: El programa XDP encontró un error (debería ser siempre 0)

- **Descartar**: Paquetes descartados intencionalmente por el programa XDP
- **Pasar**: Paquetes pasados a la pila de red para un procesamiento adicional
- **Redirigir**: Paquetes redirigidos directamente a la interfaz de salida
- **TX**: Paquetes transmitidos a través de XDP

Interpretación:

- **Abortado > 0**: Problema crítico con el programa eBPF o compatibilidad del kernel
- **Descartar > 0**: Caídas basadas en políticas o paquetes inválidos
- **Pasar alto**: La mayoría de los paquetes procesados en la pila de red (normal)
- **Redirigir alto**: Paquetes reenviados directamente (rendimiento óptimo)

Estadísticas de Paquetes

Desglose detallado del protocolo de paquetes y contadores de procesamiento.

Contadores de Protocolo:

- **RX ARP**: Paquetes del Protocolo de Resolución de Direcciones
- **RX GTP ECHO**: Solicitud/Respuesta de Eco GTP-U (keepalive)
- **RX GTP OTHER**: Otros mensajes de control GTP
- **RX GTP PDU**: Datos de usuario encapsulados en GTP-U (tráfico principal)
- **RX GTP UNEXP**: Tipos de paquetes GTP inesperados
- **RX ICMP**: Protocolo de Mensajes de Control de Internet (ping, errores)
- **RX ICMP6**: Paquetes ICMPv6
- **RX IP4**: Paquetes IPv4
- **RX IP6**: Paquetes IPv6
- **RX OTHER**: Otros protocolos
- **RX TCP**: Paquetes del Protocolo de Control de Transmisión
- **RX UDP**: Paquetes del Protocolo de Datagramas de Usuario

Casos de Uso:

- **Monitorear el conteo de GTP-U PDU**: Indicador principal del tráfico de usuarios
- **Verificar ICMP para conectividad**: Pruebas de accesibilidad de red
- **Rastrear la proporción de TCP vs UDP**: Patrones de tráfico de aplicaciones
- **Detectar protocolos inesperados**: Problemas de seguridad o de configuración

---

Estadísticas de Rutas

Estadísticas de búsqueda FIB (Base de Información de Enrutamiento) para decisiones de enrutamiento.

Búsqueda FIB IPv4:

- **Caché**: Búsquedas de ruta en caché (ruta rápida)
- **OK**: Búsquedas de ruta exitosas

Búsqueda FIB IPv6:

- **Caché**: Búsquedas de ruta IPv6 en caché
- **OK**: Búsquedas de ruta IPv6 exitosas

Indicadores de Rendimiento:

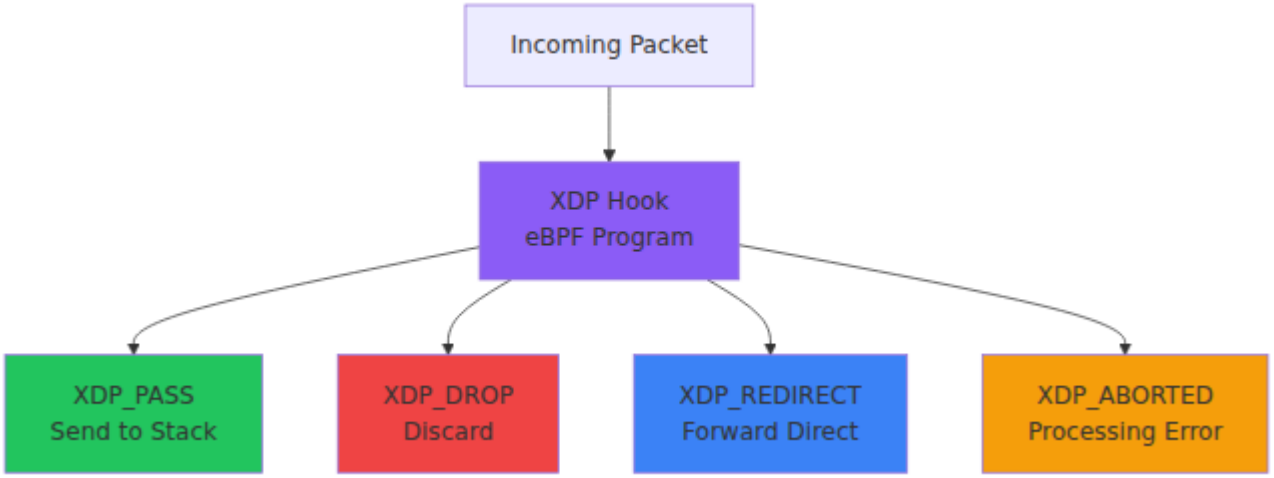
- **Alta Tasa de Acierto de Caché**: Indica buen rendimiento de la caché de enrutamiento
- **Alto Conteo de OK**: Confirma que las tablas de enrutamiento están correctamente configuradas
- **Bajas o Cero Búsquedas**: Puede indicar que el tráfico no fluye o que se está evitando el enrutamiento

---

Monitoreo de Capacidad

Capacidad del Mapa eBPF

El monitoreo de la capacidad del mapa eBPF previene fallos en el establecimiento de sesiones debido al agotamiento de recursos.



Mapas eBPF Críticos

far_map (Reglas de Acción de Reenvío):

- **Capacidad**: 131,070 entradas
- **Tamaño de Clave**: 4 B (ID FAR)
- **Tamaño de Valor**: 16 B (parámetros de reenvío)
- **Uso de Memoria**: ~2.6 MB
- **Criticidad**: Alta - Usado para todas las decisiones de reenvío de paquetes

pdr_map_downlin (PDR de Enlace Descendente - IPv4):

- **Capacidad**: 131,070 entradas
- **Tamaño de Clave**: 4 B (dirección IPv4 del UE)
- **Tamaño de Valor**: 208 B (info PDR)
- **Uso de Memoria**: ~27 MB
- **Criticidad**: Crítica - El establecimiento de sesiones falla si está lleno

pdr_map_downlin_ip6 (PDR de Enlace Descendente - IPv6):

- **Capacidad**: 131,070 entradas
- **Tamaño de Clave**: 16 B (dirección IPv6 del UE)
- **Tamaño de Valor**: 208 B (info PDR)
- **Uso de Memoria**: ~29 MB
- **Criticidad**: Crítica - El establecimiento de sesiones IPv6 falla si está lleno

pdr_map_teid_ip (PDR de Enlace Ascendente):

- **Capacidad**: 131,070 entradas
- **Tamaño de Clave**: 4 B (TEID)
- **Tamaño de Valor**: 208 B (info PDR)
- **Uso de Memoria**: ~27 MB
- **Criticidad**: Crítica - El tráfico de enlace ascendente falla si está lleno

qer_map (Reglas de Aplicación de QoS):

- **Capacidad**: 65,535 entradas
- **Tamaño de Clave**: 4 B (ID QER)
- **Tamaño de Valor**: 32 B (parámetros de QoS)
- **Uso de Memoria**: ~2.3 MB
- **Criticidad**: Media - Solo afecta la aplicación de QoS

urr_map (Reglas de Reporte de Uso):

- **Capacidad**: 131,070 entradas
- **Tamaño de Clave**: 4 B (ID URR)
- **Tamaño de Valor**: 16 B (contadores de volumen)
- **Uso de Memoria**: ~2.6 MB
- **Criticidad**: Baja - Solo afecta la facturación

Umbrales de Capacidad

Umbral	Acción Requerida
0-50% (Verde)	Operación normal - No se requiere acción
50-70% (Amarillo)	Precaución - Monitorear tendencias de crecimiento, planificar aumento de capacidad
70-90% (Ámbar)	Advertencia - Programar aumento de capacidad dentro de 1 semana
90-100% (Rojo)	Crítico - Se requiere acción inmediata, las nuevas sesiones fallarán

Procedimiento para Aumentar la Capacidad

Antes de aumentar la capacidad:

1. Revisar tendencias de uso actuales
2. Estimar la tasa de crecimiento futura
3. Calcular la capacidad requerida

Pasos para aumentar la capacidad del mapa:

1. Detener el servicio de OmniUPF
2. Actualizar el archivo de configuración de UPF con los nuevos tamaños de mapa
3. Reiniciar el servicio de OmniUPF
4. Verificar la nueva capacidad en la vista de Capacidad
5. Monitorear para el establecimiento exitoso de sesiones

Nota: Cambiar la capacidad del mapa eBPF requiere reiniciar el UPF y borra todas las sesiones existentes.

Métricas de Rendimiento

Para información detallada sobre todas las métricas de Prometheus expuestas por OmniUPF, consulte la **Referencia de Métricas**.

Tasa de Procesamiento de Paquetes

Cálculo:

```
Tasa de Paquetes (pps) = (Delta de Conteo de Paquetes) / (Delta de Tiempo en segundos)
```

Ejemplo:

- Paquetes RX iniciales: 7,000

- Después de 10 segundos: 17,000
- Tasa de Paquetes = (17,000 - 7,000) / 10 = 1,000 pps

Objetivos de Rendimiento:

- **UPF Pequeño**: 10,000 - 100,000 pps
- **UPF Mediano**: 100,000 - 1,000,000 pps
- **UPF Grande**: 1,000,000 - 10,000,000 pps

Indicadores de Cuello de Botella:

- Aumento en el conteo de abortos de XDP
- Alta utilización de CPU
- Aumento en caídas de paquetes
- Aumento en latencia

Cálculo de Rendimiento

Cálculo:

```
Rendimiento (Mbps) = (Delta de Conteo de Bytes × 8) / (Delta de
Tiempo en segundos × 1,000,000)
```

Ejemplo:

- Bytes RX iniciales: 500 MB
- Después de 60 segundos: 800 MB
- Rendimiento = (300 MB × 8) / (60 × 1,000,000) = 40 Mbps

Planificación de Capacidad:

- Monitorear tiempos de rendimiento máximo (por ejemplo, horas de la tarde)
- Comparar con la capacidad del enlace (velocidades de interfaz N3/N6)
- Planificar para 2x el rendimiento máximo para margen

Tasa de Caídas

Cálculo:

```
Tasa de Caídas (%) = (Paquetes Descartados / Total de Paquetes RX)
× 100
```

Umbrales Aceptables:

- **< 0.1%**: Excelente (pérdida de paquetes normal debido a errores)
- **0.1% - 1%**: Bueno (problemas menores o limitación de tasa)
- **1% - 5%**: Pobre (investigar problemas de QoS o capacidad)
- **> 5%**: Crítico (problema mayor de reenvío o capacidad)

Causas Comunes de Caídas:

- Limitación de tasa QER (MBR excedido)
- Fallos en la búsqueda del mapa eBPF
- TEIDs o IPs de UE inválidos
- Errores de enrutamiento

Alertas y Umbrales

Alertas Recomendadas

Alertas Críticas (se requiere respuesta inmediata):

- Capacidad del mapa eBPF > 90%
- Conteo de abortos de XDP > 0
- Tasa de caídas > 5%
- Fallo en la verificación de salud del UPF

Alertas de Advertencia (respuesta dentro de 1 hora):

- Capacidad del mapa eBPF > 70%

- Tasa de caídas > 1%
- Tasa de paquetes acercándose a la capacidad del enlace
- TTL del buffer excedido (paquetes mayores de 30s)

Alertas Informativas (monitorear tendencias):

- Capacidad del mapa eBPF > 50%
- Aumento en el conteo de paquetes en buffer
- Nuevas asociaciones PFCP establecidas/liberadas
- Umbrales de volumen URR excedidos

Configuración de Alertas

Las alertas se pueden configurar a través de:

1. **Métricas de Prometheus**: Exportar métricas para monitoreo externo (ver Referencia de Métricas para la lista completa)
2. **Monitoreo de Logs**: Analizar logs de OmniUPF para patrones de error
3. **Polling de API REST**: Consultar periódicamente los endpoints `/map_info`, `/packet_stats`
4. **Monitoreo de UI Web**: Monitoreo manual a través de las páginas de Estadísticas y Capacidad

---

Planificación de Capacidad

Estimación de Capacidad de Sesiones

Calcular sesiones máximas:

```
Sesiones Máximas = min(
  Capacidad del Mapa PDR / 2,  # PDR de enlace descendente +
ascendente por sesión
  Capacidad del Mapa FAR / 2,  # FAR de enlace descendente +
ascendente por sesión
  Capacidad del Mapa QER       # Opcional, un QER por sesión
)
```

Ejemplo:

- Capacidad del Mapa PDR: 131,070
- Capacidad del Mapa FAR: 131,070
- Capacidad del Mapa QER: 65,535

Sesiones Máximas = min(131,070 / 2, 131,070 / 2, 65,535) = **65,535 sesiones**

Capacidad de Memoria

Calcular la memoria total del mapa eBPF:

```
Memoria = Σ (Capacidad del Mapa × (Tamaño de Clave + Tamaño de
Valor))
```

Ejemplo de Configuración:

- Mapas PDR: 3 × 131,070 × 212 B = 83.3 MB
- Mapa FAR: 131,070 × 20 B = 2.6 MB
- Mapa QER: 65,535 × 36 B = 2.3 MB
- Mapa URR: 131,070 × 20 B = 2.6 MB
- **Total**: ~91 MB de memoria del kernel

Consideraciones de Memoria del Kernel:

- Asegurar límite de memoria bloqueada suficiente (`ulimit -l`)
- Reservar 2x el uso estimado para margen de seguridad
- Monitorear la disponibilidad de memoria del kernel

Capacidad de Tráfico

Calcular la capacidad de rendimiento requerida:

1. **Estimar el rendimiento promedio por sesión**:
    - Streaming de video: ~5 Mbps
    - Navegación web: ~1 Mbps
    - VoIP: ~0.1 Mbps
2. **Calcular el rendimiento agregado**:

    ```
    Rendimiento Total = Sesiones × Rendimiento Promedio por Sesión
    ```

3. **Agregar margen**:

    ```
    Capacidad Requerida = Rendimiento Total × 2  # 100% de margen
    ```

Ejemplo:

- 10,000 sesiones concurrentes
- Promedio de 2 Mbps por sesión
- Total: 20 Gbps
- Capacidad requerida: 40 Gbps (interfaces N3 + N6)

Planificación de Crecimiento

Análisis de Tendencias:

1. Registrar el conteo máximo de sesiones diarias
2. Calcular la tasa de crecimiento semanal
3. Extrapolar al límite de capacidad

Fórmula de Tasa de Crecimiento:

```
Semanas hasta Capacidad = (Capacidad - Uso Actual) / (Crecimiento Semanal)
```

Ejemplo:

- Sesiones actuales: 30,000
- Capacidad: 65,535 sesiones
- Crecimiento semanal: 2,000 sesiones
- Semanas hasta la capacidad: (65,535 - 30,000) / 2,000 = **17.8 semanas**

Acción: Planificar actualización de capacidad en 12 semanas (dejando un margen de 5 semanas).

Resolución de Problemas de Rendimiento

Alta Tasa de Caídas de Paquetes

Síntomas: Tasa de caídas > 1%, quejas de usuarios sobre mala conectividad

Diagnóstico:

1. Verificar Estadísticas → Estadísticas de Paquetes
2. Identificar si las caídas son específicas de un protocolo
3. Revisar Estadísticas de XDP para caídas vs. abortos de XDP

Causas Comunes:

- **Limitación de Tasa QER**: Verificar valores MBR de QER vs. tráfico real
- **TEIDs Inválidos**: Verificar que el TEID PDR de enlace ascendente coincida con la asignación de gNB
- **IPs de UE Desconocidas**: Verificar que exista un PDR de enlace descendente para la IP del UE
- **Desbordamiento de Buffer**: Verificar estadísticas de buffer

Resolución:

- Aumentar el MBR de QER si hay limitación de tasa
- Verificar que el SMF haya creado los PDR correctos
- Limpiar buffers si se detecta desbordamiento

Errores de Procesamiento de XDP

Síntomas: Abortos de XDP > 0

Diagnóstico:

1. Navegar a Estadísticas → Estadísticas de XDP
2. Verificar contador de abortos
3. Revisar logs de OmniUPF para errores de eBPF

Causas Comunes:

- Fallo de verificación del programa eBPF
- Incompatibilidad de versión del kernel
- Errores de acceso al mapa eBPF
- Corrupción de memoria

Resolución:

- Reiniciar el servicio de OmniUPF
- Verificar que la versión del kernel cumpla con los requisitos mínimos (Linux 5.4+)
- Revisar logs del programa eBPF
- Contactar soporte si el problema persiste

Agotamiento de Capacidad

Síntomas: Fallos en el establecimiento de sesiones, capacidad del mapa al 100%

Diagnóstico:

1. Navegar a la página de Capacidad
2. Identificar qué mapa está al 100%
3. Verificar si las sesiones están atascadas (no se están eliminando)

Mitigación Inmediata:

1. Identificar sesiones obsoletas (verificar página de Sesiones)
2. Solicitar al SMF que elimine sesiones antiguas
3. Limpiar buffers para liberar entradas de FAR

Resolución a Largo Plazo:

1. Aumentar la capacidad del mapa eBPF
2. Programar reinicio de UPF con mapas más grandes
3. Implementar políticas de limpieza de sesiones

Degradación del Rendimiento

Síntomas: Alta latencia, bajo rendimiento, saturación de CPU

Diagnóstico:

1. Verificar tasa de paquetes vs. línea base histórica
2. Revisar estadísticas de XDP para retrasos en el procesamiento
3. Monitorear la utilización de CPU en el host de UPF
4. Verificar la utilización de la interfaz N3/N6

Causas Comunes:

- Tráfico que excede la capacidad de UPF
- Núcleos de CPU insuficientes para el procesamiento de paquetes
- Cuello de botella en la interfaz de red
- Colisiones de hash en el mapa eBPF

Resolución:

- Escalar UPF horizontalmente (agregar más instancias)
- Actualizar CPU o habilitar RSS (Escalado Lateral de Recepción)
- Actualizar interfaces de red a mayor velocidad
- Ajustar la función hash del mapa eBPF

Documentación Relacionada

- **Referencia de Métricas** - Referencia completa de métricas de Prometheus
- **Guía de Operaciones de UPF** - Arquitectura y operaciones generales de UPF
- **Guía de Gestión de Reglas** - Configuración de PDR, FAR, QER, URR
- **Guía de Operaciones de UI Web** - Funciones de monitoreo del panel de control
- **Guía de Resolución de Problemas** - Problemas comunes y diagnósticos
- **Guía de Arquitectura** - Datapath eBPF y optimización de rendimiento

N9 Loopback: Ejecutando SGWU y PGWU en la Misma Instancia

Descripción General

OmniUPF admite la ejecución de funciones tanto de **SGWU (Gateway de Servicio del Plano de Usuario)** como de **PGWU (Gateway de PDN del Plano de Usuario)** en la **misma instancia** con **bucle N9 de cero latencia**. Este modo de implementación es ideal para:

- **Despliegues simplificados de EPC 4G** - Una sola instancia de UPF en lugar de dos
- **Optimización de costos** - Reducción de la complejidad de infraestructura y operativa
- **Computación en el borde** - Minimizar la latencia para escenarios de ruptura local
- **Entornos de laboratorio/pruebas** - Plano de usuario EPC completo en un solo servidor

Cuando se configura con la misma dirección IP para las interfaces N3 y N9, OmniUPF **detecta automáticamente** el tráfico que fluye entre los roles SGWU y PGWU y lo procesa **totalmente en eBPF** sin enviar paquetes a la interfaz de red.

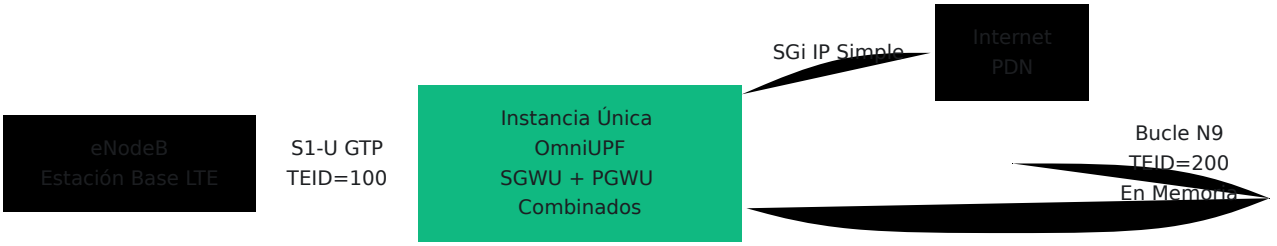
Cómo Funciona

Implementación Tradicional (Dos Instancias)

Flujo de Paquetes:

1. eNodeB → SGWU: Paquete GTP (TEID=100) llega a S1-U
2. SGWU: Coincide con PDR de subida, encapsula en un nuevo túnel GTP (TEID=200)
3. **Paquete enviado a través de la red física N9** a la instancia PGWU
4. PGWU: Recibe GTP (TEID=200), desencapsula, reenvía a Internet
5. **Total: 2 pasadas XDP + 1 salto de red**

Implementación de Bucle N9 (Instancia Única)



Flujo de Paquetes con Bucle N9:

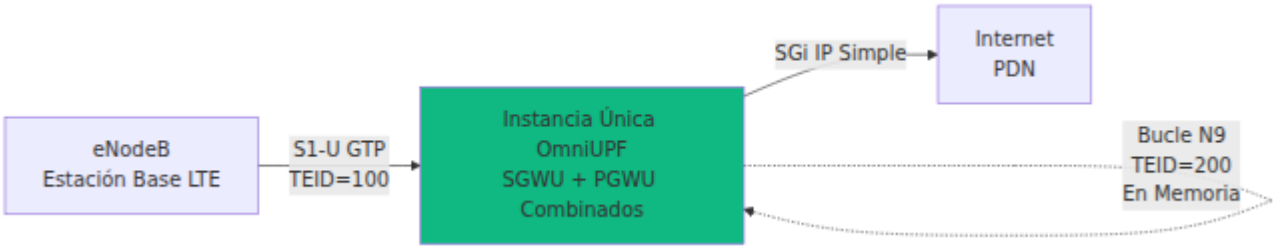
1. eNodeB → rol SGWU: Paquete GTP (TEID=100) llega a S1-U
2. rol SGWU: Coincide con PDR de subida
3. **Detección de bucle:** IP de destino = IP local (10.0.1.10)
4. **Procesamiento en el lugar:** Actualizar GTP TEID a 200 (sesión PGWU)
5. rol PGWU: Desencapsula, reenvía a Internet
6. **Total: 1 pasada XDP, cero saltos de red**

Beneficio de rendimiento: Reenvío interno sub-microsegundo frente a milisegundos para el viaje de ida y vuelta de la red

---

Detalles de Procesamiento de Paquetes

Flujo de Subida: eNodeB → SGWU → PGWU → Internet

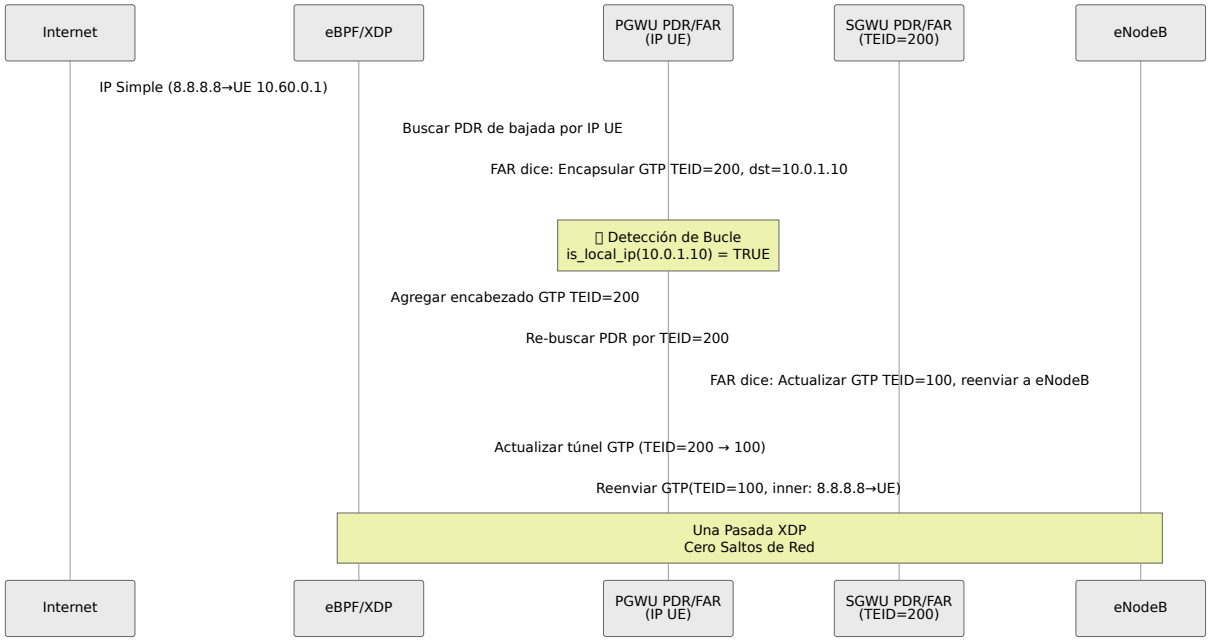


Ruta de Código eBPF: `ebpf/xdp/n3n6_entrypoint.c` líneas 349-403

Pasos Clave:

1. **Recibir:** Paquete GTP de eNodeB con TEID=100
2. **Coincidencia de PDR:** Buscar PDR de subida para la sesión SGWU (TEID=100)
3. **Acción FAR:** Encapsular en GTP con TEID=200, reenviar a 10.0.1.10
4. **Verificación de Bucle:** `is_local_ip(10.0.1.10)` devuelve TRUE
5. **Actualizar TEID:** Cambiar `ctx->gtp->teid` de 100 a 200 (en memoria del kernel)
6. **Re-Procesar:** Buscar PDR para TEID=200 (sesión PGWU)
7. **Acción FAR:** Eliminar encabezado GTP, reenviar a Internet
8. **Ruta:** Enviar paquete IP simple a la interfaz N6

Flujo de Bajada: Internet → PGWU → SGWU → eNodeB



Ruta de Código eBPF: `ebpf/xdp/n3n6_entrypoint.c` líneas 137-194 (IPv4), 265-322 (IPv6)

Pasos Clave:

1. **Recibir:** Paquete IP simple de Internet destinado a UE (10.60.0.1)
2. **Coincidencia de PDR:** Buscar PDR de bajada por IP UE (sesión PGWU)
3. **Acción FAR:** Encapsular en GTP con TEID=200, reenviar a 10.0.1.10
4. **Verificación de Bucle:** `is_local_ip(10.0.1.10)` devuelve TRUE
5. **Agregar GTP:** Encapsular paquete con TEID=200
6. **Re-Procesar:** Buscar PDR para TEID=200 (sesión SGWU)
7. **Acción FAR:** Actualizar túnel GTP a eNodeB TEID=100
8. **Ruta:** Enviar paquete GTP a la interfaz S1-U (eNodeB)

Configuración

Requisitos

Plano de Control:

- **SGWU-C**: Debe conectarse a la interfaz PFCP de OmniUPF (por ejemplo, `192.168.1.10:8805`)
- **PGWU-C**: Debe conectarse a la **misma** interfaz PFCP de OmniUPF

Red:

- **Una sola dirección IP** para ambas interfaces N3 y N9
- **Direcciones IP diferentes** para SGWU-C y PGWU-C (si se ejecutan en el mismo host, usar diferentes puertos)

Configuración de OmniUPF

`/etc/omniupf/runtime.exs`:

```
# Interfaces de red
xdp_interfaces = "eth0"              # Una sola interfaz para S1-U
y N9
xdp_attach_mode = "native"           # Usar nativo para mejor
rendimiento

# Interfaz PFCP
pfcp_address = "192.168.1.10"        # Dirección PFCP de OmniUPF
pfcp_port = 8805                     # Puerto PFCP
node_id = "192.168.1.10"             # ID de Nodo PFCP de OmniUPF

# Interfaces del Plano de Usuario
n3_address = "10.0.1.10"             # IP de la interfaz S1-U/N3
n9_address = n3_address               # IP de la interfaz N9
(IGUAL que N3)

# Grupos de Recursos
feature_ueip = true
ueip_pool = "10.60.0.0/16"           # Grupo de direcciones IP de
UE
feature_ftup = true
teid_pool_start = 1
teid_pool_end = 65_535

# Capacidad
max_sessions = 100_000               # Máximo de sesiones
concurrentes de UE

# API
api_port = 8080
```

Configuración Clave:

- `n3_address` y `n9_address` **DEBEN ser idénticos** para habilitar el bucle
- Dirección de escucha PFCP única para ambos planos de control
- Suficiente `max_sessions` para la carga combinada de SGWU + PGWU

Configuración del Plano de Control

Configuración de SGWU-C

```
# Apuntar a la interfaz PFCP de OmniUPF
upf_pfcp_address: "192.168.1.10:8805"

# Interfaz S1-U (igual que la n3_address de OmniUPF)
sgwu_s1u_address: "10.0.1.10"

# Interfaz N9 para reenvío a PGWU (igual que OmniUPF)
sgwu_n9_address: "10.0.1.10"
```

Configuración de PGWU-C

```
# Apuntar a la MISMA interfaz PFCP de OmniUPF
upf_pfcp_address: "192.168.1.10:8805"

# Interfaz N9 (recibe de SGWU)
pgwu_n9_address: "10.0.1.10"

# Interfaz SGi para conectividad a Internet
pgwu_sgi_address: "192.168.100.1"
```

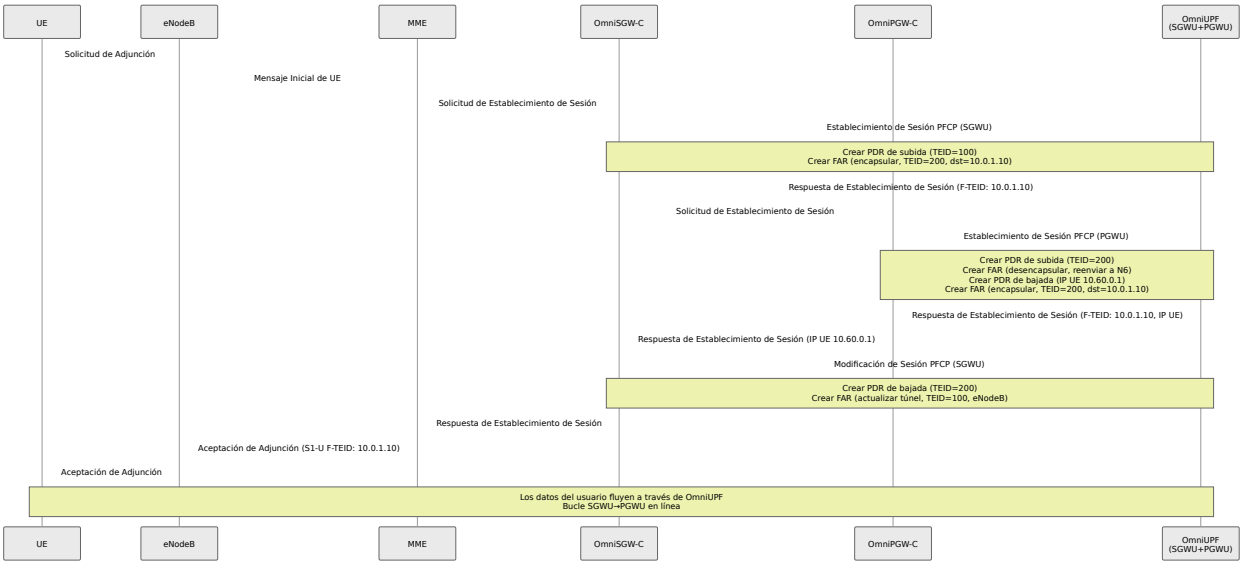
Importante:

- Ambos planos de control se conectan al **mismo punto final PFCP** (`:8805`)
- OmniUPF crea **asociaciones PFCP separadas** para SGWU-C y PGWU-C
- Las sesiones están aisladas por plano de control (seguimiento por ID de Nodo)

Ejemplo de Flujo de Sesión

Adjuntar UE y Establecimiento de Sesión PDU

Escenario: UE se adjunta a la red, establece sesión de datos



Sesiones PFCP Creadas:

Sesión SGWU (desde OmniSGW-C):

- **PDR de Subida:** Coincidir TEID=100 (desde eNodeB) → FAR: Encapsular TEID=200, dst=10.0.1.10
- **PDR de Bajada:** Coincidir TEID=200 (desde PGWU) → FAR: Actualizar túnel TEID=100, reenviar a eNodeB

Sesión PGWU (desde OmniPGW-C):

- **PDR de Subida:** Coincidir TEID=200 (desde SGWU) → FAR: Desencapsular, reenviar a Internet
- **PDR de Bajada:** Coincidir IP UE=10.60.0.1 → FAR: Encapsular TEID=200, dst=10.0.1.10

Monitoreo y Verificación

Verificar que el Bucle N9 esté Activo

Verificar Registros XDP:

```
# Ver salida de depuración eBPF en tiempo real
sudo cat /sys/kernel/debug/tracing/trace_pipe | grep loopback
```

Salida esperada:

```
upf: [n3] sesión para teid:100 -> 200 remoto:10.0.1.10
upf: [n9-loopback] detección de auto-reenvío, procesamiento en
línea TEID:200
upf: [n9-loopback] desencapsulado, enrutando a N6

upf: [n6] usar mapeo 10.60.0.1 -> teid:200
upf: [n6-loopback] detección de auto-reenvío de bajada,
procesamiento en línea TEID:200
upf: [n6-loopback] SGWU actualizando túnel GTP a eNodeB TEID:100
upf: [n6-loopback] reenviando a eNodeB
```

Monitorear Sesiones a través de la API REST

Listar Asociaciones PFCP:

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

Salida esperada:


```
{
  "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
}
```

Verificar dos asociaciones separadas (una para SGWU-C, una para PGWU-C)

Listar Sesiones Activas:

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

Salida esperada:

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

Cada UE tiene DOS sesiones:

- Sesión de SGWU-C (TEID=100, interfaz S1-U)
- Sesión de PGWU-C (TEID=200, interfaz N9)

Métricas de Rendimiento

Verificar Estadísticas de Paquetes:

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

Métricas clave:

- `xdp_processed`: Total de paquetes procesados en eBPF

- `xdp_pass`: Paquetes pasados a la pila de red (debería ser cero para tráfico de bucle)
- `xdp_redirect`: Paquetes reenviados a través de redirección XDP
- `xdp_tx`: Paquetes transmitidos (el tráfico de bucle utiliza esto)

Para tráfico de bucle N9:

- `xdp_pass` debería ser **mínimo** (solo tráfico no de bucle)
- `xdp_tx` o `xdp_redirect` cuentan el reenvío de bucle

Solución de Problemas

Tráfico N9 yendo a la Red en lugar de Bucle

Síntoma: Paquetes enviados a la interfaz de red, alta latencia

Causa Raíz: `n3_address` ≠ `n9_address`

Solución (en `runtime.exs`):

```
# INCORRECTO:
n3_address = "10.0.1.10"
n9_address = "10.0.1.20"   # IP diferente, ¡sin bucle!

# CORRECTO:
n3_address = "10.0.1.10"
n9_address = n3_address     # Misma IP, habilita el bucle
```

Verificación:

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

Debería mostrar:

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

PDR No Encontrado Después del Bucle

Síntoma: Los registros muestran `[n9-loopback] no PDR para TEID de destino`

Causa Raíz: Sesión PGWU no creada o desajuste de TEID

Diagnóstico:

1. **Verificar Sesiones PFCP:**

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

2. **Verificar Configuración de FAR:**

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

Solución: Asegurarse de que PGWU-C cree una sesión con TEID coincidente que SGWU-C utiliza para el reenvío N9

Alta Utilización de CPU

Síntoma: Utilización de CPU más alta de lo esperado

Causa Raíz: Programa eBPF procesando paquetes múltiples veces o búsquedas de mapa excesivas

Diagnóstico:

```
# Ver patrones de acceso al mapa eBPF
sudo bpftool map dump name pdr_map_teid_ip4 | wc -l
sudo bpftool map dump name far_map | wc -l
```

Solución:

- Aumentar `max_sessions` si el mapa está lleno (causa fallos en la búsqueda)
- Verificar que la limitación de tasa QER no esté causando caídas y retransmisiones
- Verificar si hay un almacenamiento excesivo de paquetes

Pérdida de Paquetes Durante la Transferencia

Síntoma: Paquetes perdidos durante la transferencia de eNodeB

Causa Raíz: Almacenamiento no configurado o límites de almacenamiento insuficientes

Configuración:

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

Verificación:

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

Beneficios del Bucle N9

Rendimiento

Métrica	Dos Instancias	Instancia Única (Bucle N9)	Mejora
Latencia	1-5 ms	< 1 µs	**1000x más rápido**
Rendimiento	Limitado por la red	Limitado por CPU/memoria	**2-3x más alto**
Uso de CPU	2× pasadas XDP + pila de red	1× pasada XDP	**Reducción del 40-50%**
Pérdida de Paquetes	Riesgo durante congestión de red	Cero (en memoria)	**Eliminado**

Operacional

- **Despliegue Simplificado:** Una sola instancia de OmniUPF en lugar de dos
- **Reducción de Infraestructura:** La mitad de los servidores, puertos de red, direcciones IP
- **Menor Complejidad:** Una sola configuración, un solo punto de monitoreo
- **Ahorros de Costos:** Reducción de hardware, energía, refrigeración, mantenimiento
- **Resolución de Problemas Más Fácil:** Trazado de paquetes único, salida de depuración eBPF única

Casos de Uso

Ideal Para:

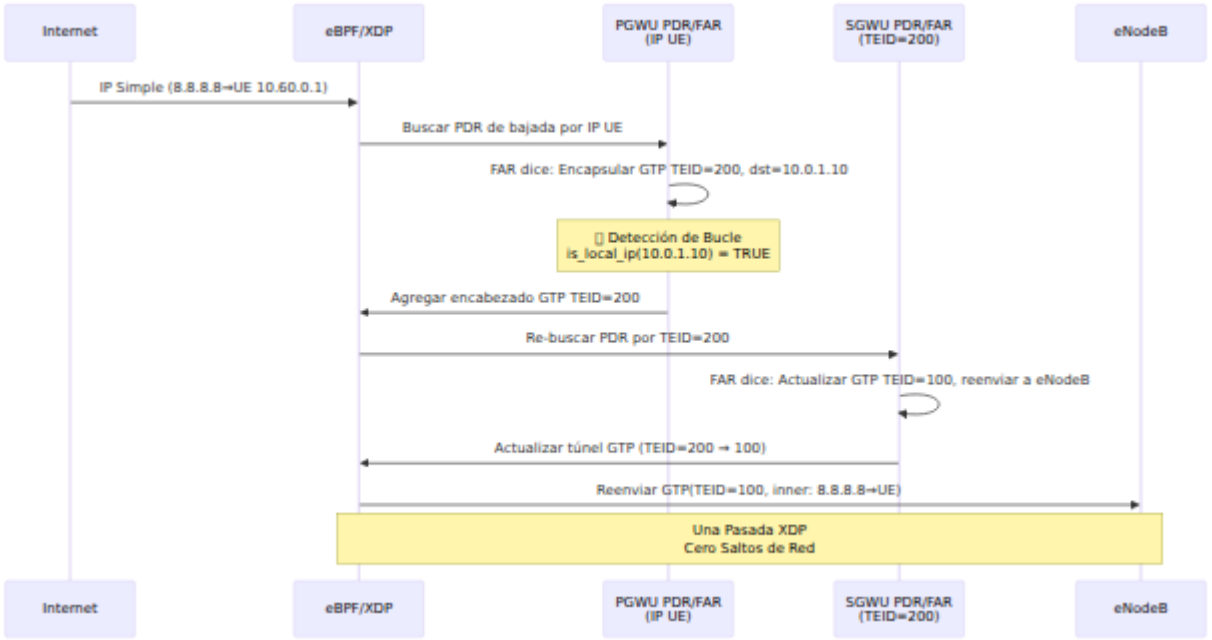
- **Computación en el Borde:** Minimizar latencia para ruptura local
- **Despliegues Pequeños/Medianos:** < 100K suscriptores
- **Laboratorio/Pruebas:** Plano de usuario EPC completo en una sola VM
- **Constrained Costs:** Presupuesto de hardware limitado

No Recomendado Para:

- **Redundancia Geográfica:** SGWU y PGWU en diferentes centros de datos
- **Escala Masiva:** > 1M suscriptores (considerar escalado horizontal)
- **Requisitos Regulatorios:** Separación obligada de SGW y PGW

Comparación con Otros Modos de Implementación

Instancia Única (Bucle N9) vs. Instancias Separadas



Resumen

El Bucle N9 permite **un plano de usuario EPC 4G de grado de operador en una sola instancia de OmniUPF** procesando el tráfico SGWU→PGWU completamente en eBPF sin saltos de red. Esto proporciona:

- **Latencia sub-microsegundo** para el reenvío entre gateways
- **Reducción del 40-50% en CPU** en comparación con instancias separadas
- **Operaciones simplificadas** - instancia única, configuración, monitoreo
- **Costo más bajo** - la mitad de la infraestructura
- **Cumplimiento total de 3GPP** - protocolos PFCP, GTP-U estándar

La configuración es automática cuando `n3_address == n9_address` - no se requieren banderas o configuraciones especiales. La ruta de datos eBPF de OmniUPF detecta condiciones de bucle y procesa paquetes en línea.

Para más información:

- **Configuración:** CONFIGURATION.md
- **Arquitectura:** ARCHITECTURE.md
- **Referencia de Métricas:** METRICS.md
- **Monitoreo:** MONITORING.md
- **Operaciones:** OPERATIONS.md
- **Solución de Problemas:** TROUBLESHOOTING.md

Referencia de Códigos de Causa PFCP

Descripción General

PFCP (Protocolo de Control de Reenvío de Paquetes) utiliza códigos de causa en mensajes de respuesta para indicar el resultado de las solicitudes. Este documento describe los códigos de causa implementados en OmniUPF y cuándo ocurren durante el procesamiento de mensajes PFCP.

Todos los códigos de causa se ajustan a las especificaciones de **3GPP TS 129.244** y se devuelven en los mensajes de respuesta PFCP para indicar éxito, fallo o condiciones de error específicas.

Monitoreo de Códigos de Causa

OmniUPF rastrea los resultados de los mensajes PFCP utilizando métricas de Prometheus. Cada respuesta PFCP incluye un código de causa que se registra en:

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

Esto permite el monitoreo de:

- **Tasas de éxito** por tipo de mensaje y nodo de plano de control
- **Patrones de error** que indican configuraciones incorrectas o problemas de protocolo
- **Salud de la asociación** basada en tasas de rechazo

Consulte Referencia de Métricas para la documentación completa de métricas PFCP.

Categorías de Códigos de Causa

Códigos de Éxito

Código	Nombre	Cuándo Ocurre
1	**RequestAccepted**	Solicitud procesada con éxito. Todos los IEs obligatorios presentes y válidos. Reglas creadas/modificadas/eliminadas con éxito.

Códigos de Error del Cliente

Código	Nombre	Cuándo Ocurre
64	**RequestRejected**	Rechazo general por errores no especificados. Utilizado cuando no se aplica un código de causa específico.
65	**SessionContextNotFound**	Modificación o eliminación de sesión solicitada para SEID desconocido. La sesión especificada no existe en este UPF.
66	**MandatoryIEMissing**	Elemento de Información requerido ausente. Ejemplos: NodeID faltante en la Configuración de Asociación, F-SEID faltante en el Establecimiento de Sesión, RecoveryTimeStamp faltante.
67	**ConditionalIEMissing**	IE condicionalmente requerido ausente basado en otros IEs presentes. Utilizado cuando los IEs dependen de la presencia de otros.
69	**MandatoryIEIncorrect**	IE requerido presente pero contiene datos inválidos. Ejemplos: Formato de

Código	Nombre	Cuándo Ocurre
		NodeID no parseable, valor de RecoveryTimeStamp inválido, F-SEID mal formado.
72	**NoEstablishedPFCPAssociation**	Operación de sesión intentada sin asociación activa. Debe establecerse la asociación PFCP antes de crear sesiones.
73	**RuleCreationModificationFailure**	Error al aplicar reglas PDR, FAR, QER o URR al datapath eBPF. Causas posibles: capacidad del mapa eBPF agotada, parámetros de regla inválidos, fallo en la asignación de recursos.

Códigos de Error del Servidor/Recurso

Código	Nombre	Cuándo Ocurre
74	**PFCPEntityInCongestion**	UPF experimentando alta carga o agotamiento de recursos. Temporalmente incapaz de procesar solicitudes.
75	**NoResourcesAvailable**	Recursos insuficientes para cumplir con la solicitud. Ejemplos: capacidad del mapa eBPF agotada, fallo en la asignación de memoria, pool de TEID agotado.
77	**SystemFailure**	Error interno crítico que impide el procesamiento de la solicitud. Ejemplos: fallo del programa eBPF, error en la interfaz del kernel, corrupción de base de datos.

Códigos de Característica No Soportada

Código	Nombre	Cuándo Ocurre
68	**InvalidLength**	El campo de longitud de IE no coincide con la longitud de datos real. Actualmente no utilizado en OmniUPF.
70	**InvalidForwardingPolicy**	Política de reenvío no soportada por UPF. Actualmente no utilizado en OmniUPF.
71	**InvalidFTEIDAllocationOption**	Opción de asignación de F-TEID no soportada. Actualmente no utilizado en OmniUPF.
76	**ServiceNotSupported**	Servicio o característica solicitada no implementada. Actualmente no utilizado en OmniUPF.
78	**RedirectionRequested**	UPF solicita redirección a otra instancia de UPF. Actualmente no utilizado en OmniUPF.

Escenarios Comunes y Causas

Fallos en la Configuración de Asociación

Escenario: NodeID Faltante

```
SMF → UPF: Solicitud de Configuración de Asociación (sin NodeID)
UPF → SMF: Respuesta de Configuración de Asociación (Causa:
MandatoryIEMissing)
```

Resolución: Asegúrese de que SMF incluya el IE NodeID en todas las Solicitudes de Configuración de Asociación.

Escenario: Formato de NodeID Inválido

```
SMF → UPF: Solicitud de Configuración de Asociación
(NodeID="inválido")
UPF → SMF: Respuesta de Configuración de Asociación (Causa:
MandatoryIEIncorrect)
```

Resolución: NodeID debe ser un FQDN válido o una dirección IPv4/IPv6.

Escenario: Timestamp de Recuperación Faltante

```
SMF → UPF: Solicitud de Configuración de Asociación (sin
RecoveryTimeStamp)
UPF → SMF: Respuesta de Configuración de Asociación (Causa:
MandatoryIEMissing)
```

Resolución: Incluir RecoveryTimeStamp en la Solicitud de Configuración de Asociación.

Fallos en el Establecimiento de Sesión

Escenario: No Asociación Establecida

```
SMF → UPF: Solicitud de Establecimiento de Sesión
UPF → SMF: Respuesta de Establecimiento de Sesión (Causa:
NoEstablishedPFCPAssociation)
```

Resolución: Establecer la asociación PFCP antes de crear sesiones.

Escenario: Fallo en la Creación de Reglas

```
SMF → UPF: Solicitud de Establecimiento de Sesión
UPF procesa FARs, QERs, URRs con éxito
UPF falla al crear PDR (mapa eBPF lleno)
UPF → SMF: Respuesta de Establecimiento de Sesión (Causa:
RuleCreationModificationFailure)
```

Resolución:

- Verifique la capacidad del mapa eBPF (consulte Monitoreo de Capacidad)
- Aumente los tamaños de los mapas en la configuración de UPF
- Reduzca el conteo de sesiones activas

Escenario: F-SEID Faltante

```
SMF → UPF: Solicitud de Establecimiento de Sesión (sin CP F-SEID)
UPF → SMF: Respuesta de Establecimiento de Sesión (Causa:
MandatoryIEMissing)
```

Resolución: Incluir CP F-SEID en la Solicitud de Establecimiento de Sesión.

Fallos en la Modificación de Sesión

Escenario: SEID Desconocido

```
SMF → UPF: Solicitud de Modificación de Sesión (SEID=12345)
UPF no tiene sesión con SEID 12345
UPF → SMF: Respuesta de Modificación de Sesión (Causa:
SessionContextNotFound)
```

Resolución:

- Verifique que el SEID coincida con el valor de la Respuesta de Establecimiento de Sesión
- Verifique si la sesión ya fue eliminada
- Asegúrese de usar la instancia correcta de UPF (escenarios de bucle N9)

Fallos en la Eliminación de Sesión

Escenario: SEID Desconocido

```
SMF → UPF: Solicitud de Eliminación de Sesión (SEID=67890)
UPF no tiene sesión con SEID 67890
UPF → SMF: Respuesta de Eliminación de Sesión (Causa:
SessionContextNotFound)
```

Resolución: El SEID puede haber sido eliminado previamente o nunca existió.

Solución de Problemas con Códigos de Causa

Usando Métricas de Prometheus

Consulta Prometheus para identificar patrones de error:

```
# Tasa de error por código de causa
rate(upf_pfcp_rx_errors{cause_code!="RequestAccepted"}[5m])

# Principales causas de rechazo
topk(5, sum by (cause_code) (upf_pfcp_rx_errors))

# Errores por par SMF
sum by (peer_address, cause_code)
(upf_pfcp_rx_errors{cause_code!="RequestAccepted"})

# Fallos en el establecimiento de sesión
upf_pfcp_rx_errors{message_name="SessionEstablishmentRequest",
cause_code!="RequestAccepted"}
```

Usando la Interfaz Web

Navegue a la página de **Sesiones** para ver:

- Conteo de sesiones activas por nodo de plano de control
- Tasas de éxito/fallo en el establecimiento de sesiones
- Errores recientes de sesión

Navegue a la página de **Capacidad** para diagnosticar:

- Utilización del mapa eBPF (causa raíz de RuleCreationModificationFailure)
- Indicadores de agotamiento de recursos

Consulte Guía de Interfaz Web para instrucciones detalladas de monitoreo.

Pasos Comunes de Depuración

Alta Tasa de MandatoryIEMissing:

1. Verifique la configuración de SMF para los IEs requeridos
2. Verifique la compatibilidad de la versión de la biblioteca PFCP
3. Revise los registros de SMF en busca de errores de construcción de IE

Frecuentes RuleCreationModificationFailure:

1. Verifique la capacidad del mapa eBPF: `GET /api/v1/map_info`
2. Monitoree el uso del mapa: `upf_ebpf_map_used / upf_ebpf_map_capacity`
3. Aumente los tamaños de los mapas en la configuración si > 70% utilizado
4. Consulte Planificación de Capacidad

Errores NoEstablishedPFCPAssociation:

1. Verifique que la asociación exista: `GET /api/v1/pfcp_associations`
2. Verifique la configuración del tiempo de espera del heartbeat
3. Revise los registros de configuración de la asociación
4. Asegúrese de que SMF y UPF puedan alcanzarse mutuamente

SessionContextNotFound en Modificación:

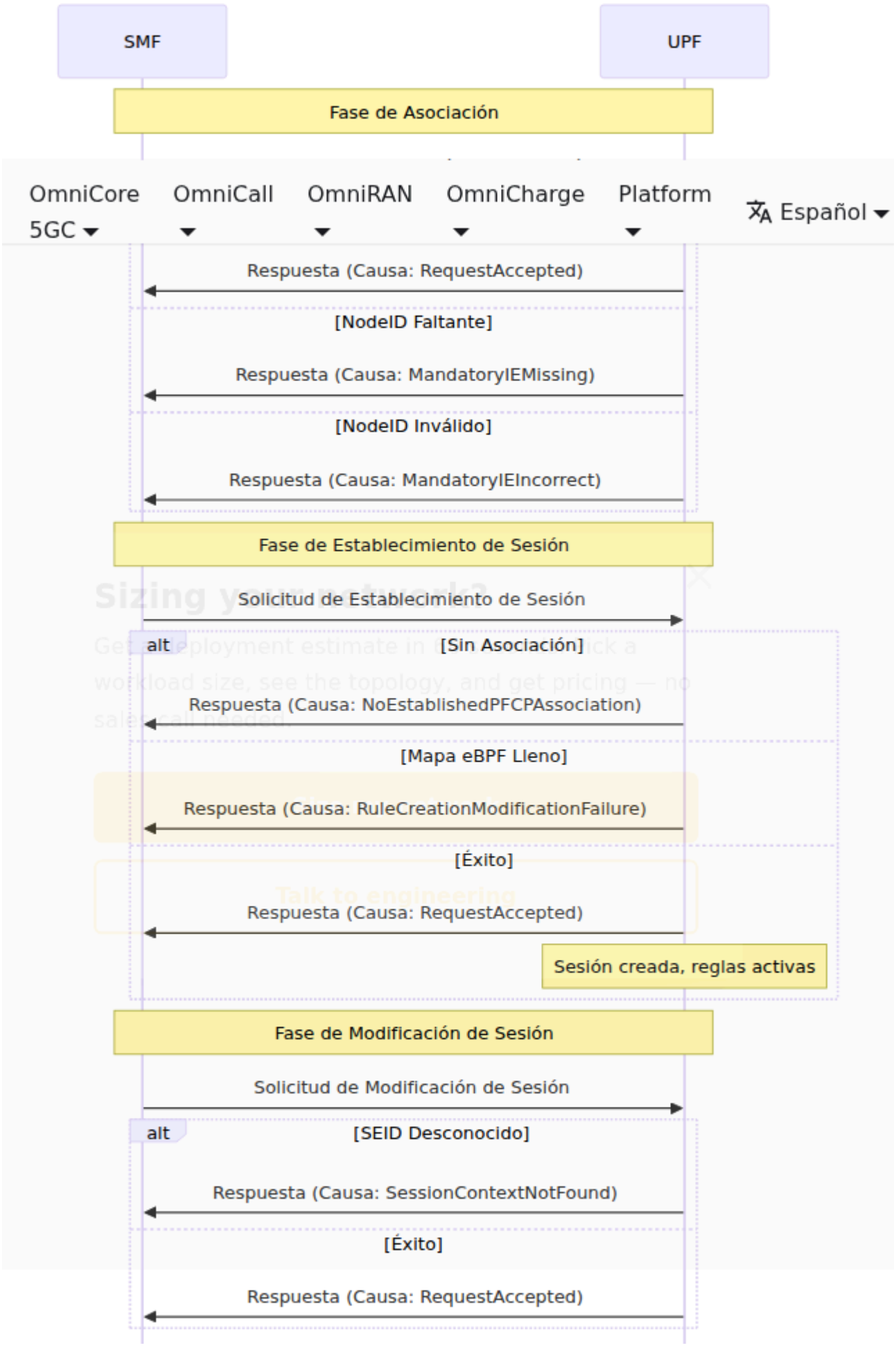
1. Verifique el SEID de la respuesta de establecimiento de sesión
2. Verifique si la sesión fue eliminada
3. Para bucle N9: Asegúrese de usar el endpoint correcto de UPF

4. Consulte sesiones activas: `GET /api/v1/pfcp_sessions`

Impacto del Código de Causa en las

Operaciones

Ciclo de Vida de la Sesión


SMF
UPF
Fase de Asociación
OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
Español
Respuesta (Causa: RequestAccepted)
[NodeID Faltante]
Respuesta (Causa: MandatoryIEMissing)
[NodeID Inválido]
Respuesta (Causa: MandatoryIEIncorrect)
Fase de Establecimiento de Sesión
Solicitud de Establecimiento de Sesión
alt
[Sin Asociación]
Respuesta (Causa: NoEstablishedPFCPAssociation)
[Mapa eBPF Lleno]
Respuesta (Causa: RuleCreationModificationFailure)
[Éxito]
Respuesta (Causa: RequestAccepted)
Sesión creada, reglas activas
Fase de Modificación de Sesión
Solicitud de Modificación de Sesión
alt
[SEID Desconocido]
Respuesta (Causa: SessionContextNotFound)
[Éxito]
Respuesta (Causa: RequestAccepted)

Métricas y Alertas

Alertas Recomendadas:

```
# Crítico: Alta tasa de rechazo
- alert: PfcpHighRejectionRate
  expr: |
    rate(upf_pfcp_rx_errors{cause_code!="RequestAccepted"}[5m]) > 0.1
  annotations:
    summary: "Alta tasa de rechazo PFCP: {{ $value }}/s"

# Advertencia: Problemas de capacidad
- alert: PfcpRuleCreationFailures
  expr: |

rate(upf_pfcp_rx_errors{cause_code="RuleCreationModificationFailure"}
[5m]) > 0
  annotations:
    summary: "Se detectaron fallos en la creación de reglas PFCP"

# Advertencia: Problemas de asociación
- alert: PfcpNoAssociation
  expr: |

rate(upf_pfcp_rx_errors{cause_code="NoEstablishedPFCPAssociation"}
[5m]) > 0
  annotations:
    summary: "Sesiones PFCP intentadas sin asociación"
```

Cumplimiento de Normas 3GPP

OmniUPF implementa códigos de causa de acuerdo con:

- **3GPP TS 129.244 v16.4.0** - Especificación PFCP
- **Sección 8.2.1** - Definición de IE de Causa
- **Sección 8.19** - Tabla de valores de Causa

Documentación Relacionada

- **Integración del Protocolo PFCP** - Arquitectura PFCP y manejo de mensajes
- **Referencia de Métricas** - Documentación de la métrica upf_pfcp_rx_errors
- **Guía de Monitoreo** - Monitoreo de capacidad y alertas
- **Guía de Solución de Problemas** - Problemas de asociación y sesión PFCP
- **Guía de Interfaz Web** - Monitoreo de sesiones y asociaciones

Guía de Gestión de Reglas

Tabla de Contenidos

1. Descripción General
2. Reglas de Detección de Paquetes (PDR)
3. Reglas de Acción de Reenvío (FAR)
4. Reglas de Aplicación de QoS (QER)
5. Reglas de Reporte de Uso (URR)
6. Relaciones de Reglas
7. Operaciones Comunes
8. Solución de Problemas

Descripción General

OmniUPF utiliza un conjunto de reglas interconectadas para clasificar, reenviar, dar forma y rastrear el tráfico del plano de usuario. Estas reglas son instaladas por el SMF a través de PFCP y almacenadas en mapas eBPF para un procesamiento de paquetes de alto rendimiento. Comprender estas reglas y sus relaciones es crítico para operar y solucionar problemas en el UPF.

Tipos de Reglas

Tipo de Regla	Propósito	Campo Clave	Instalado Por
PDR (Regla de Detección de Paquetes)	Clasificar paquetes en flujos	TEID o IP de UE	SMF a través de Establecimiento/Modificación de Sesión PFCP
FAR (Regla de Acción de Reenvío)	Determinar acción de reenvío	ID de FAR	SMF a través de Establecimiento/Modificación de Sesión PFCP
QER (Regla de Aplicación de QoS)	Aplicar límites de ancho de banda y marcado	ID de QER	SMF a través de Establecimiento/Modificación de Sesión PFCP
URR (Regla de Reporte de Uso)	Rastrear volúmenes de datos para cobro	ID de URR	SMF a través de Establecimiento/Modificación de Sesión PFCP

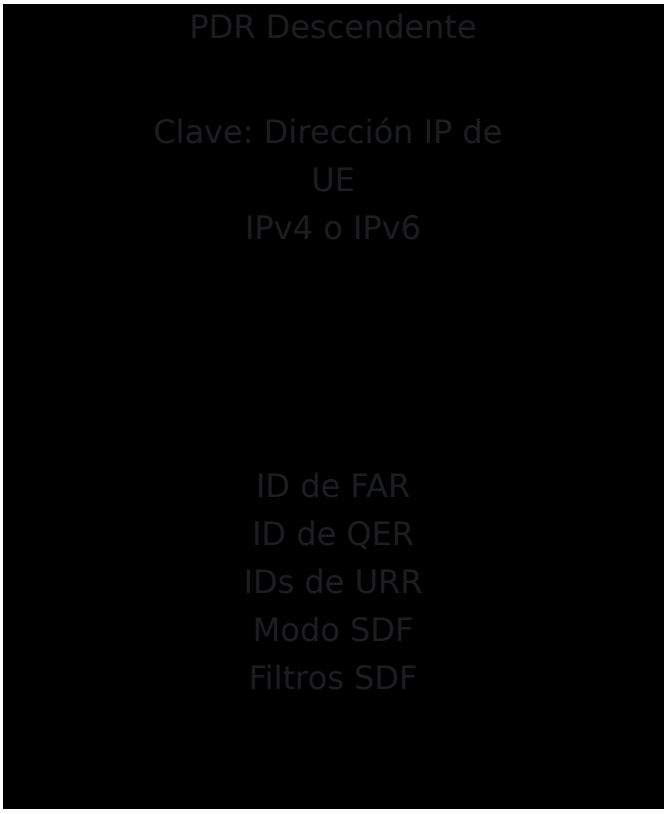
Flujo de Procesamiento de Reglas

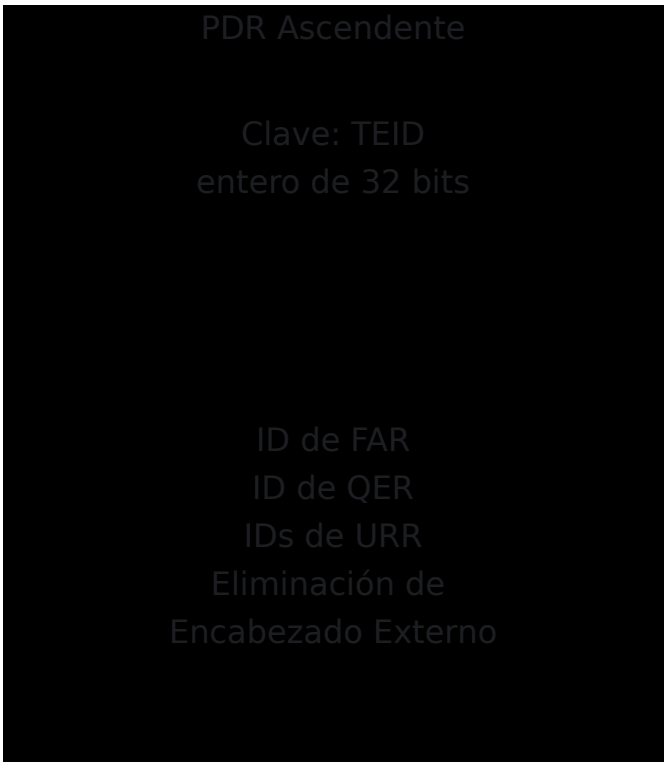
Reglas de Detección de Paquetes (PDR)

Propósito

Las PDR clasifican los paquetes entrantes en flujos de tráfico. Son el punto de entrada para todo el procesamiento de paquetes en el UPF.

Estructura de PDR





PDR Ascendentes

Las PDR ascendentes coinciden con los paquetes que llegan a la interfaz N3 desde la RAN.

Campo Clave: TEID (Identificador de Punto Final de Túnel)

- Entero sin signo de 32 bits
- Asignado por el SMF y señalizado al gNB
- Único por flujo de tráfico de UE

Campos de Valor:

- **ID de FAR**: Referencia a la regla de acción de reenvío
- **ID de QER**: Referencia a la regla de aplicación de QoS (opcional)
- **IDs de URR**: Lista de reglas de reporte de uso (opcional)
- **Eliminación de Encabezado Externo**: Bandera para eliminar la encapsulación GTP-U

Proceso de Búsqueda:

1. Extraer TEID del encabezado GTP-U
2. Búsqueda hash en el mapa eBPF `uplink_pdr_map`
3. Si se encuentra coincidencia, recuperar ID de FAR, ID de QER y IDs de URR
4. Si no hay coincidencia, descartar paquete

Ejemplo:

```
TEID: 5678
ID de FAR: 2
ID de QER: 1
Eliminación de Encabezado Externo: Falso
Modo SDF: Sin SDF
```

PDR Descendentes

Las PDR descendentes coinciden con los paquetes que llegan a la interfaz N6 desde la red de datos.

Campo Clave: Dirección IP de UE

- Dirección IPv4 (32 bits) o dirección IPv6 (128 bits)
- Asignada por el SMF durante el establecimiento de la sesión PDU
- Única por UE

Campos de Valor:

- **ID de FAR**: Referencia a la regla de acción de reenvío
- **ID de QER**: Referencia a la regla de aplicación de QoS (opcional)
- **IDs de URR**: Lista de reglas de reporte de uso (opcional)
- **Modo SDF**: Modo de filtro de Flujo de Datos de Servicio
  - `Sin SDF`: Sin filtrado, todo el tráfico coincide

    - `Solo SDF`: Solo el tráfico que coincide con SDF es reenviado
    - `SDF + Predeterminado`: El tráfico que coincide con SDF utiliza reglas específicas, el otro tráfico utiliza el FAR predeterminado
- **Filtros SDF**: Filtros específicos de la aplicación (puertos, protocolos, rangos de IP)

Proceso de Búsqueda:

1. Extraer IP de destino del encabezado del paquete
2. Búsqueda hash en `downlink_pdr_map` (IPv4) o `downlink_pdr_map_ip6` (IPv6)
3. Si se encuentra coincidencia, verificar filtros SDF (si están configurados)
4. Recuperar ID de FAR, ID de QER y IDs de URR
5. Si no hay coincidencia, descartar paquete

Ejemplo:

```
IP de UE: 10.45.0.1
ID de FAR: 1
ID de QER: 1
Eliminación de Encabezado Externo: Falso
Modo SDF: Sin SDF
```

Filtros SDF (Flujo de Datos de Servicio)

Los filtros SDF proporcionan clasificación de tráfico específica de la aplicación dentro de una PDR.

Casos de Uso:

- Diferenciar el tráfico de YouTube del de navegación web
- Aplicar diferentes QoS a VoIP frente a datos de mejor esfuerzo
- Enrutar aplicaciones específicas a través de diferentes rutas de red

Criterios de Filtro:

- **Protocolo**: TCP, UDP, ICMP
- **Rango de Puertos**: Puertos de destino (por ejemplo, 443 para HTTPS, 5060 para SIP)
- **Rango de Direcciones IP**: Redes de destino específicas
- **Descripción del Flujo**: Plantillas de flujo definidas por 3GPP

Ejemplo de Configuración de SDF:

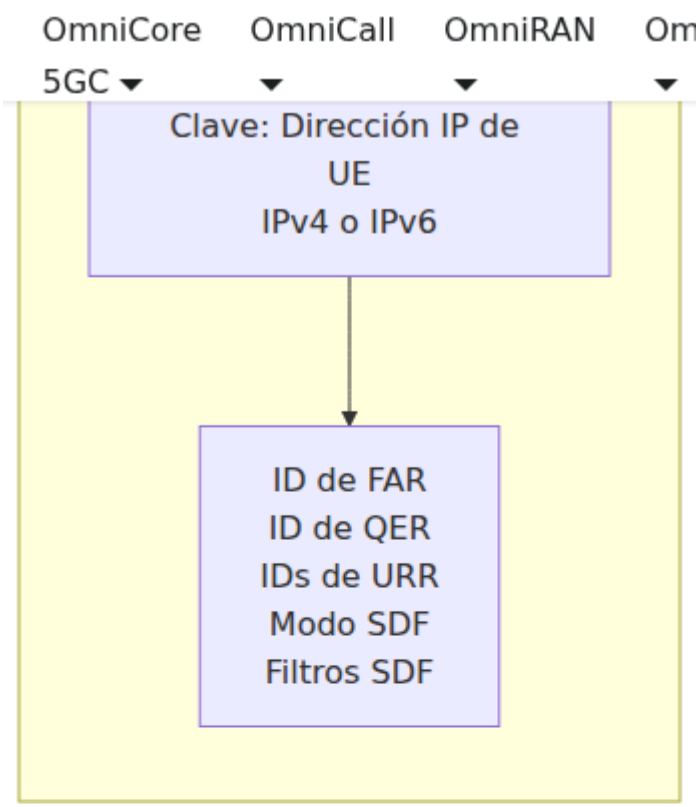
```
ID de PDR: 10
IP de UE: 10.45.0.1
Modo SDF: Solo SDF
Filtros SDF:
  - Protocolo: UDP, Puertos: 5060-5061 → ID de FAR 5 (FAR de VoIP)
  - Protocolo: TCP, Puerto: 443 → ID de FAR 1 (FAR Predeterminado)
```

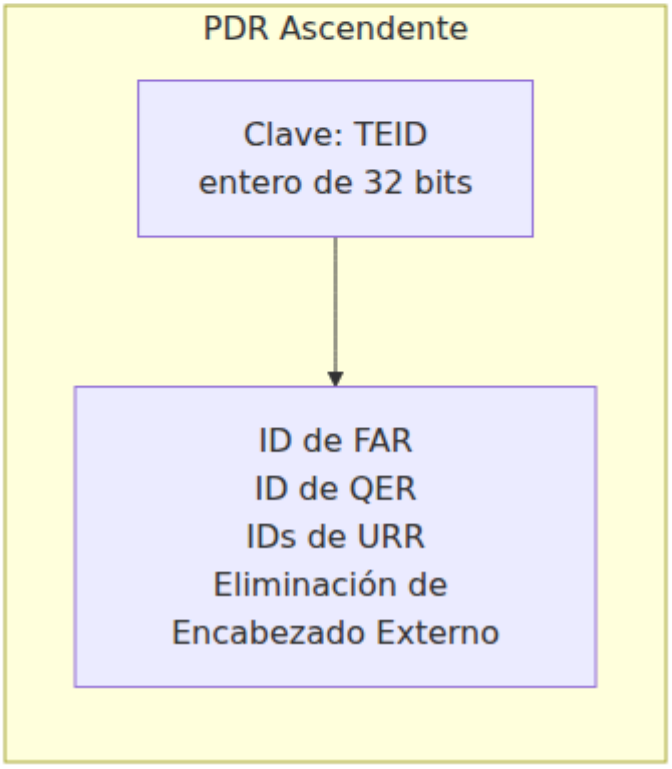
Reglas de Acción de Reenvío (FAR)

Propósito

Las FAR determinan qué hacer con los paquetes que coinciden con una PDR. Definen acciones de reenvío, parámetros de encapsulación GTP-U y puntos finales de destino.

Estructura de FAR





Banderas de Acción

Las acciones de FAR son banderas a nivel de bits que se pueden combinar:

Bandera	Bit	Valor	Descripción
REENVIAR	1	2	Reenviar paquete al destino
BUFFER	2	4	Almacenar paquete en buffer
DESCARTAR	0	1	Descartar paquete
NOTIFICAR	3	8	Enviar notificación al plano de control
DUPLICAR	4	16	Duplicar paquete a múltiples destinos

Combinaciones de Acción Comunes:

- `Acción: 2 (REENVIAR)` - Reenvío normal (más común)
- `Acción: 6 (REENVIAR + BUFFER)` - Reenviar y almacenar durante la transferencia
- `Acción: 4 (BUFFER)` - Solo almacenar (durante el cambio de ruta)
- `Acción: 1 (DESCARTAR)` - Descartar paquete (raro, generalmente para hacer cumplir políticas)

Control de Almacenamiento

La bandera BUFFER (bit 2) controla el almacenamiento de paquetes durante eventos de movilidad. El almacenamiento es una característica crítica del UPF que previene la pérdida de paquetes durante las transiciones de estado de UE.

Cuándo se Utiliza el Almacenamiento

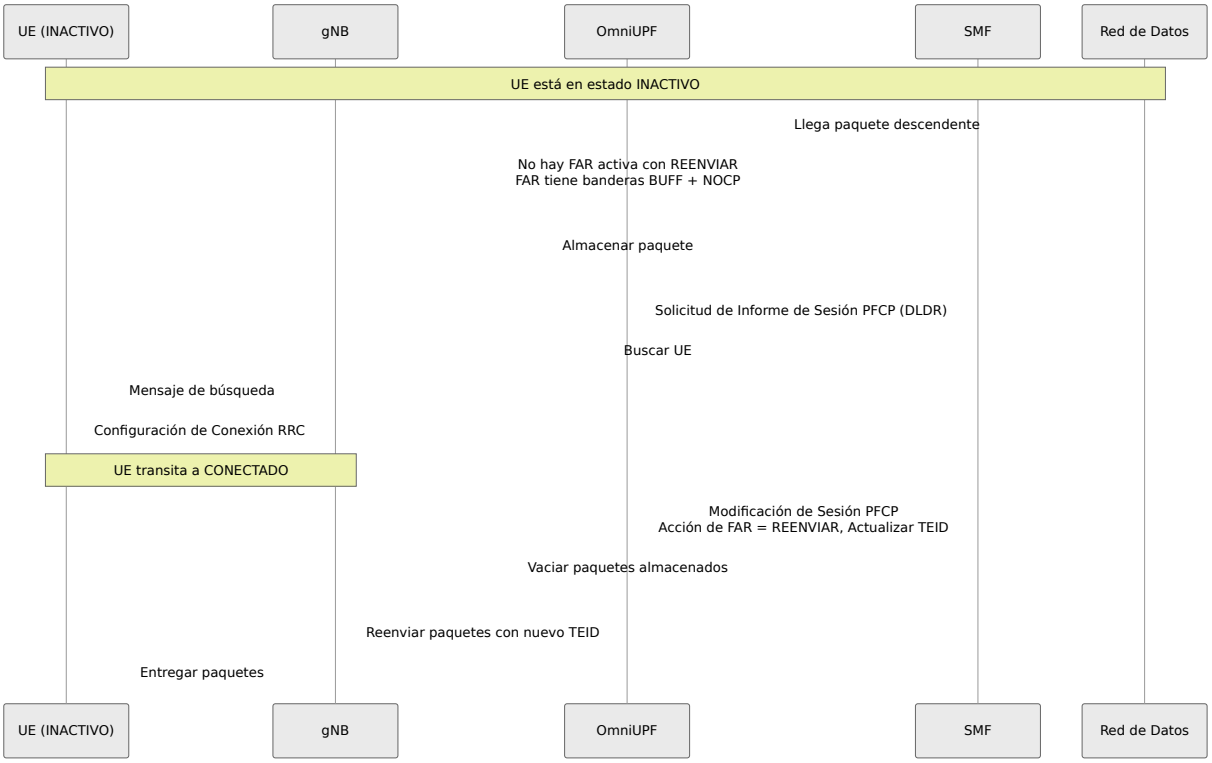
Transición de Inactivo a Conectado: Cuando los paquetes descendentes llegan para un UE en estado INACTIVO (no conectado a gNB), el UPF:

1. Almacena los paquetes

2. Envía una Notificación de Datos Descendentes (DLDR) al SMF
3. El SMF busca al UE para despertarlo y conectarlo
4. Una vez conectado, el SMF actualiza la FAR con la acción de REENVIAR
5. El UPF vacía los paquetes almacenados al UE

Transferencia (Conectado a Conectado): Durante la transferencia de gNB a gNB, el UPF almacena temporalmente los paquetes para prevenir pérdidas:

1. La conexión antigua de gNB se cierra
2. El SMF establece la acción de FAR a BUFFER
3. Los paquetes se encolan durante el cambio de ruta
4. El UE se conecta al nuevo gNB
5. El SMF actualiza la FAR con el nuevo TEID y la acción de REENVIAR
6. El UPF vacía los paquetes al nuevo gNB



Capacidad y Límites de Almacenamiento

Límites Globales de Almacenamiento:

- **Máx. Total de Paquetes**: 100,000 (configurable)
- **Máx. Total de Bytes**: Basado en la memoria disponible

- **TTL (Tiempo de Vida)**: 60 segundos (configurable)
- **Paquetes que exceden TTL**: Se descartan automáticamente

Límites por FAR:

- **Máx. Paquetes por FAR**: 10,000 (configurable)
- **Propósito**: Prevenir que una sola FAR agote la capacidad de almacenamiento

Comportamiento de Desbordamiento de Almacenamiento:

- Cuando se alcanza el límite global o por FAR, se descartan nuevos paquetes
- Las métricas rastrean descartes con `reason="global_limit"` o `reason="far_limit"`
- Los paquetes más antiguos NO se expulsan automáticamente (solo se descartan explícitamente al expirar TTL)

Notificación de Datos Descendentes (DLDR)

Cuando el UPF almacena un paquete para un UE INACTIVO, envía una Solicitud de Informe de Sesión PFCP al SMF:

Contenidos de DLDR:

- **Tipo de Informe**: Informe de Datos Descendentes (DLDR)
- **ID de FAR**: La FAR que activó el almacenamiento
- **Información del Servicio de Datos Descendentes**: QFI opcional, Indicador de Política de Búsqueda

Acciones del SMF en DLDR:

1. Buscar al UE a través de AMF → gNB
2. Esperar a que el UE establezca la conexión RRC
3. Enviar Solicitud de Modificación de Sesión PFCP para actualizar la FAR
4. La acción de FAR cambia de `BUFF+NOCP` a `REENVIAR`
5. El UPF vacía los paquetes almacenados

Métricas para DLDR:

- `upf_dldr_sent_total`: Total de DLDR enviados
- `upf_dldr_send_errors`: DLDR fallidos
- `upf_buffer_notify_to_flush_duration_seconds`: Latencia desde DLDR hasta el vaciado

Ver Referencia de Métricas para la lista completa.

Operaciones de Almacenamiento

Habilitar Almacenamiento (Establecer bandera BUFF):

- Acción de FAR `|= 0x04` (establecer bit 2)
- Ejemplo: `Acción: 2 (REENVIAR)` → `Acción: 6 (REENVIAR+BUFF)`
- Usado durante la preparación para la transferencia

Modo Solo Almacenamiento (BUFF sin REENVIAR):

- Acción de FAR `= 0x04` (solo BUFF)
- Los paquetes se almacenan pero NO se reenvían
- Usado para el estado INACTIVO de UE (pendiente de búsqueda)

Deshabilitar Almacenamiento (Limpiar bandera BUFF):

- Acción de FAR `&= ~0x04` (limpiar bit 2)
- Ejemplo: `Acción: 6 (REENVIAR+BUFF)` → `Acción: 2 (REENVIAR)`
- Los paquetes almacenados permanecen hasta que se vacíen o se limpien

Vaciar Buffer:

- Reenviar todos los paquetes almacenados utilizando las reglas de FAR **actuales**
- Los paquetes se reenvían con TEID/destino actualizado
- El buffer se vacía después de un vaciado exitoso
- La FAR debe tener la acción de REENVIAR establecida

Limpiar Buffer:

- Descartar todos los paquetes almacenados sin reenviar

- Usar cuando la transferencia falla o la sesión se elimina
- Las métricas rastrean con `reason="cleared"`

Monitoreo de Paquetes Almacenados

Página de Buffers (Interfaz Web): Navegar a **Buffers** para ver:

- Total de paquetes almacenados
- Total de bytes almacenados
- Número de FARs con paquetes almacenados
- Conteos de paquetes por FAR
- Marca de tiempo del paquete más antiguo
- Habilitar/Deshabilitar almacenamiento por FAR
- Operaciones de vaciado o limpieza

Indicadores Clave:

- **Paquetes > 10 segundos antiguos**: Posible retraso en la búsqueda
- **Paquetes > 30 segundos antiguos**: Probable fallo en la búsqueda, limpiar buffer
- **Alto conteo de paquetes**: Verificar sesiones atascadas o fallos en la búsqueda

Métricas de Prometheus:

- `upf_buffer_packets_current`: Paquetes almacenados actuales
- `upf_buffer_bytes_current`: Bytes almacenados actuales
- `upf_buffer_fars_active`: FARs con paquetes almacenados
- `upf_buffer_packets_dropped{reason}`: Conteos de paquetes descartados

Ver Referencia de Métricas para métricas completas de buffer.

Escenarios Comunes de Almacenamiento

Escenario 1: UE INACTIVO Datos Descendentes

```
Estado Inicial:
- UE en modo INACTIVO (sin conexión a gNB)
- Acción de FAR: 0x04 (solo BUFF)

Llegada de Datos:
1. DN envía paquete descendente
2. UPF coincide con PDR, aplica FAR
3. FAR tiene bandera BUFF → paquete almacenado
4. UPF envía DLDR al SMF
5. SMF busca al UE
6. UE se conecta al gNB
7. SMF modifica FAR: Acción = 0x02 (REENVIAR)
8. UPF vacía paquetes almacenados con nuevo TEID
```

Escenario 2: Preparación para Transferencia

```
Estado Inicial:
- UE conectado a gNB-1 (TEID 1234)
- Acción de FAR: 0x02 (REENVIAR)

Proceso de Transferencia:
1. SMF modifica FAR: Acción = 0x06 (REENVIAR+BUFF)
2. Paquetes reenviados a gNB-1 Y almacenados
3. UE cambia a gNB-2
4. SMF modifica FAR: TEID = 5678, Acción = 0x02 (REENVIAR)
5. UPF vacía paquetes almacenados a gNB-2 con nuevo TEID
6. Sin pérdida de paquetes durante la transferencia
```

Escenario 3: Cambio de Ruta

```
Estado Inicial:
- UE conectado, flujo de datos activo

Cambio de Ruta:
1. SMF modifica FAR: Acción = 0x04 (solo BUFF)
2. Todos los paquetes entrantes se almacenan (no se reenvían)
3. La red reconfigura la ruta
4. SMF modifica FAR: Acción = 0x02 (REENVIAR), nuevo destino
5. UPF vacía todos los paquetes almacenados a la nueva ruta
```

Creación de Encabezado Externo

Determina si se debe agregar la encapsulación GTP-U.

FAR Ascendente (N3 → N6):

- Creación de Encabezado Externo: Falso
- Acción: Eliminar GTP-U, reenviar paquete IP nativo

FAR Descendente (N6 → N3):

- Creación de Encabezado Externo: Verdadero
- IP Remota: dirección IP del gNB (por ejemplo, 200.198.5.10)
- TEID: ID de túnel para tráfico de UE
- Acción: Agregar encabezado GTP-U, reenviar al gNB

Búsqueda de FAR en la Interfaz Web

La página de Gestión de Reglas proporciona búsqueda de FAR por ID:

Pasos:

1. Navegar a Reglas → pestaña FARs
2. Ingresar ID de FAR en el campo de búsqueda
3. Hacer clic en "Buscar" para ver detalles de FAR

Información Mostrada:

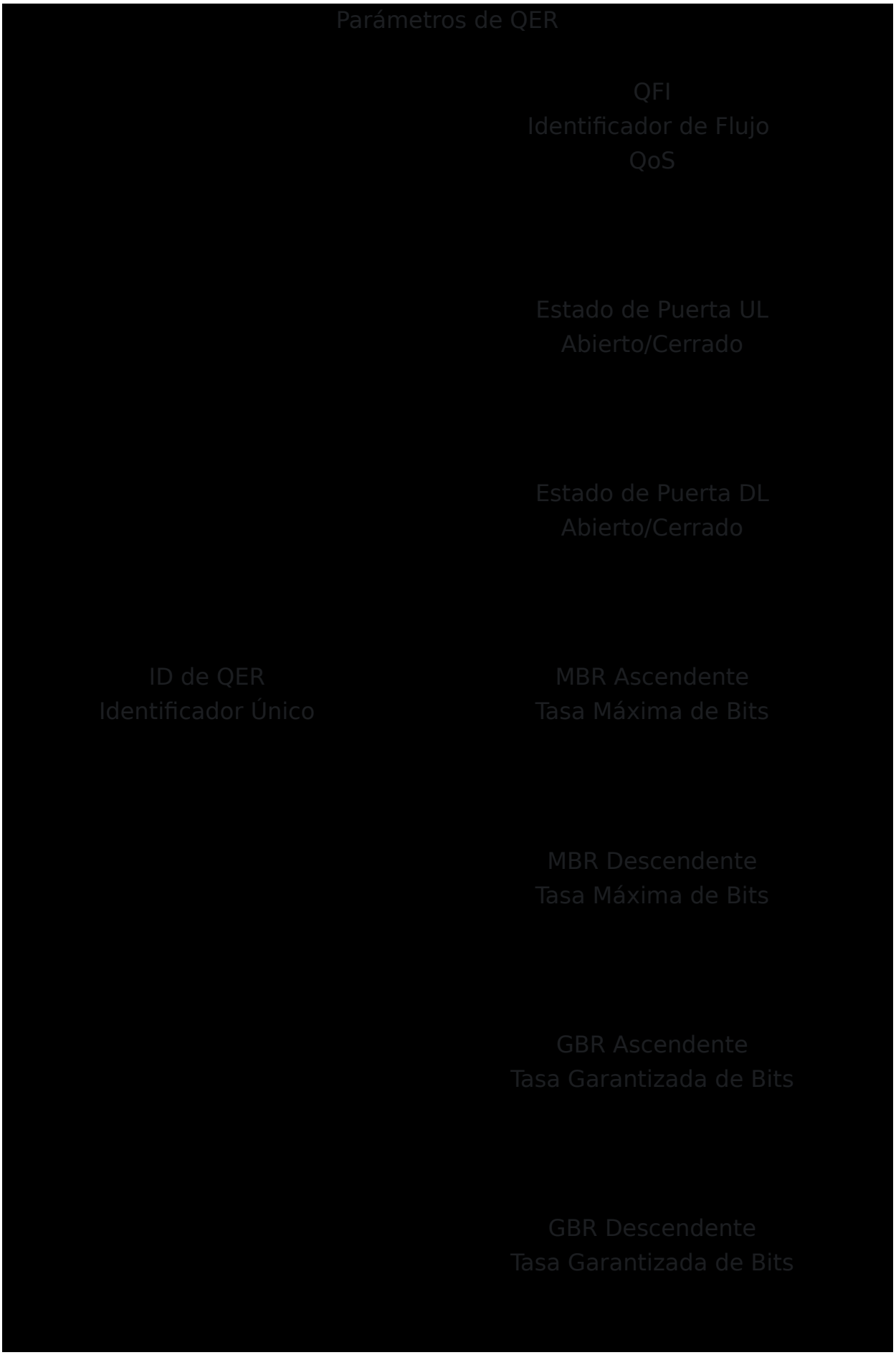
- ID de FAR
- Acción (numérica + banderas decodificadas)
- Estado de almacenamiento (ON/OFF)
- Creación de Encabezado Externo
- Dirección IP remota (con representación entera)
- TEID
- Marcado de Nivel de Transporte

Reglas de Aplicación de QoS (QER)

Propósito

Las QER aplican parámetros de Calidad de Servicio a flujos de tráfico, incluidos límites de ancho de banda y marcado de paquetes.

Estructura de QER


Parámetros de QER
QFI
Identificador de Flujo
QoS
Estado de Puerta UL
Abierto/Cerrado
Estado de Puerta DL
Abierto/Cerrado
ID de QER
Identificador Único
MBR Ascendente
Tasa Máxima de Bits
MBR Descendente
Tasa Máxima de Bits
GBR Ascendente
Tasa Garantizada de Bits
GBR Descendente
Tasa Garantizada de Bits

Parámetros de QoS

QFI (Identificador de Flujo QoS):

- Identificador de 6 bits para flujos QoS 5G
- Los valores 1-9 están estandarizados (por ejemplo, QFI 9 = portadora predeterminada)
- Usado para el marcado de paquetes en 5GC

Estado de Puerta:

- **Abierto (0)**: Tráfico permitido
- **Cerrado (no cero)**: Tráfico bloqueado

Tasa Máxima de Bits (MBR):

- Ancho de banda máximo permitido para el flujo de tráfico
- Especificado en kbps
- **MBR = 0**: Sin límite de tasa (ilimitado)
- El tráfico que excede el MBR se descarta

Tasa Garantizada de Bits (GBR):

- Ancho de banda mínimo garantizado para el flujo de tráfico
- Especificado en kbps
- **GBR = 0**: Mejor esfuerzo (sin garantía)
- **GBR > 0**: Flujo priorizado con ancho de banda garantizado

Tipos de Flujos QoS

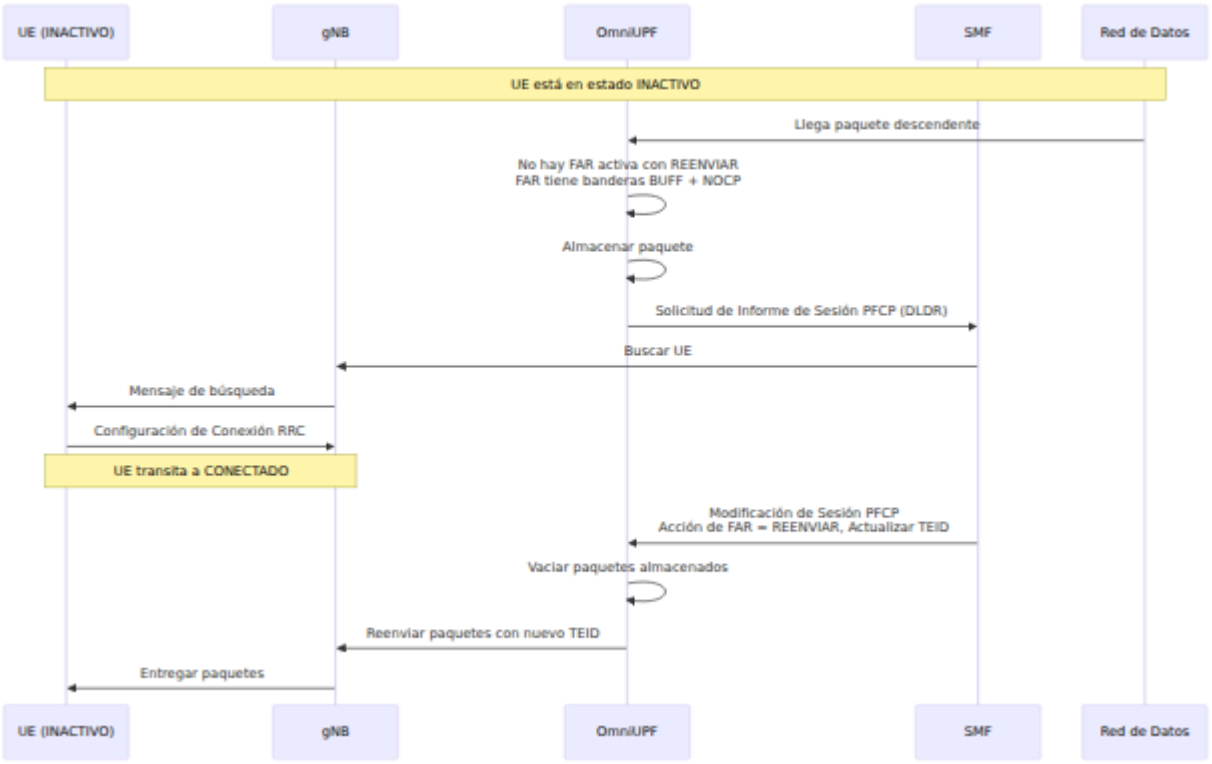
Flujos de Mejor Esfuerzo (GBR = 0):

```
ID de QER: 1
QFI: 9
MBR Ascendente: 100000 kbps (100 Mbps)
MBR Descendente: 100000 kbps (100 Mbps)
GBR Ascendente: 0 kbps
GBR Descendente: 0 kbps
```

Flujos Garantizados (GBR > 0):

```
ID de QER: 2
QFI: 1
MBR Ascendente: 10000 kbps (10 Mbps)
MBR Descendente: 10000 kbps (10 Mbps)
GBR Ascendente: 5000 kbps (5 Mbps)
GBR Descendente: 5000 kbps (5 Mbps)
```

Algoritmo de Aplicación de QoS



Mecanismo de Aplicación de MBR

OmniUPF aplica límites de MBR (Tasa Máxima de Bits) utilizando un **limitador de tasa de ventana deslizante** implementado en la ruta de datos eBPF. Este algoritmo opera con precisión de nanosegundos directamente en la capa XDP, asegurando un rendimiento a la tasa de línea sin cambios de contexto del kernel.

Cómo Funciona

Algoritmo: Limitación de Tasa de Ventana Deslizante

Para cada paquete, el UPF realiza las siguientes verificaciones:

1. **Verificación del Estado de la Puerta**: Si el estado de la puerta es `CERRADA` (no cero), descartar el paquete inmediatamente
2. **Verificación de MBR**: Si MBR = 0, omitir la limitación de tasa (ancho de banda ilimitado)
3. **Cálculo del Tiempo de Transmisión**:

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

4. **Verificación de Ventana**: Si el tiempo actual está dentro de la ventana deslizante de 5 ms, descartar el paquete
5. **Avance de Ventana**: Si el paquete es permitido, avanzar la ventana por `tx_time`

Ejemplo de Cálculo:

Supongamos:

- MBR = 100,000 kbps (100 Mbps)
- Tamaño del paquete = 1500 bytes
- Tamaño de la ventana = 5,000,000 ns (5 ms)

```
Paso 1: Calcular el tiempo de transmisión a 100 Mbps
tx_time = (1500 bytes × 8 bits/byte) × (1,000,000,000 ns/sec) /
100,000,000 bps
        = 12,000,000,000 / 100,000,000
        = 120 ns

Paso 2: Verificar si el paquete cabe en la ventana
current_time = 1000000000 ns
window_start = 999990000 ns
if (window_start + tx_time > current_time):
    DESCARTAR paquete (excedería el límite de tasa)

Paso 3: Si se permite, avanzar la ventana
window_start = window_start + 120 ns
PASAR paquete
```

Comportamiento de Ventana Deslizante

Tamaño de Ventana de 5ms:

- El algoritmo utiliza una ventana deslizante de 5 milisegundos
- La ventana se restablece automáticamente si está inactiva durante más de 5 ms

- Previene la inanición de ráfagas mientras aplica la tasa promedio

Manejo de Ráfagas:

- Se permiten pequeñas ráfagas dentro de la ventana de 5 ms
- El tráfico sostenido por encima de MBR se limita en tasa
- Más preciso que algoritmos de cubo de tokens simples

Limitación de Tasa por Dirección:

- MBR ascendente utiliza la marca de tiempo `qer->ul_start`
- MBR descendente utiliza la marca de tiempo `qer->dl_start`
- Cada dirección se limita en tasa de manera independiente

Puntos de Aplicación de Limitación de MBR

Ascendente (N3 → N6):

1. El paquete llega a la interfaz N3 (desde gNB)
2. Búsqueda de PDR por TEID
3. Búsqueda de QER por ID de QER
4. Verificar `ul_gate_status` → descartar si está cerrado
5. Aplicar `limit_rate_sliding_window()` con `ul_maximum_bitrate`
6. Si se permite, reenviar a N6 y actualizar contadores de URR

Descendente (N6 → N3):

1. El paquete llega a la interfaz N6 (desde la Red de Datos)
2. Búsqueda de PDR por dirección IP de UE
3. Búsqueda de QER por ID de QER
4. Verificar `dl_gate_status` → descartar si está cerrado
5. Aplicar `limit_rate_sliding_window()` con `dl_maximum_bitrate`
6. Si se permite, agregar encabezado GTP-U y reenviar a N3

Bucle N9 (SGWU ↔ PGWU):

- Tanto las QER ascendentes como descendentes pueden aplicarse en escenarios de bucle N9

- Cada QER se verifica de manera independiente en los límites de SGWU y PGWU

MBR vs. Rendimiento Observado

Por qué el rendimiento observado puede diferir del MBR:

- **Sobrecarga de Protocolo**: GTP-U, UDP, encabezados IP añaden ~50-60 bytes por paquete
- **Variación en el Tamaño de Paquete**: Paquetes más pequeños = más sobrecarga, menor eficiencia
- **Precisión de Limitación de Tasa**: La aplicación ocurre por paquete, no por byte
- **Comportamiento de Restablecimiento de Ventana**: Períodos inactivos de 5 ms permiten ráfagas breves por encima de MBR

Ejemplo:

```
MBR Configurado: 100 Mbps
Rendimiento Observado: ~95-98 Mbps (debido a la sobrecarga de GTP-U/UDP/IP)
```

Cómo Verificar la Limitación de Tasa:

1. Verificar contadores de volumen de URR a lo largo del tiempo: `upf_urr_*_volume_bytes`
2. Calcular el rendimiento: `(volume_delta_bytes × 8) / time_delta_seconds / 1000` = kbps
3. Comparar con el MBR configurado en QER

GBR (Tasa Garantizada de Bits)

Importante: OmniUPF **no** aplica actualmente los mínimos de GBR. GBR se almacena en la QER pero no se utiliza para la priorización del tráfico o el control de admisión.

Comportamiento de GBR:

- Los valores de GBR se aceptan del SMF a través de PFCP
- GBR se almacena en el mapa de QER y es visible a través de la API
- **Sin reserva de ancho de banda** o priorización del tráfico basada en GBR
- GBR sirve como metadatos para rastrear el tipo de flujo (mejor esfuerzo vs. garantizado)

Mejora Futura:

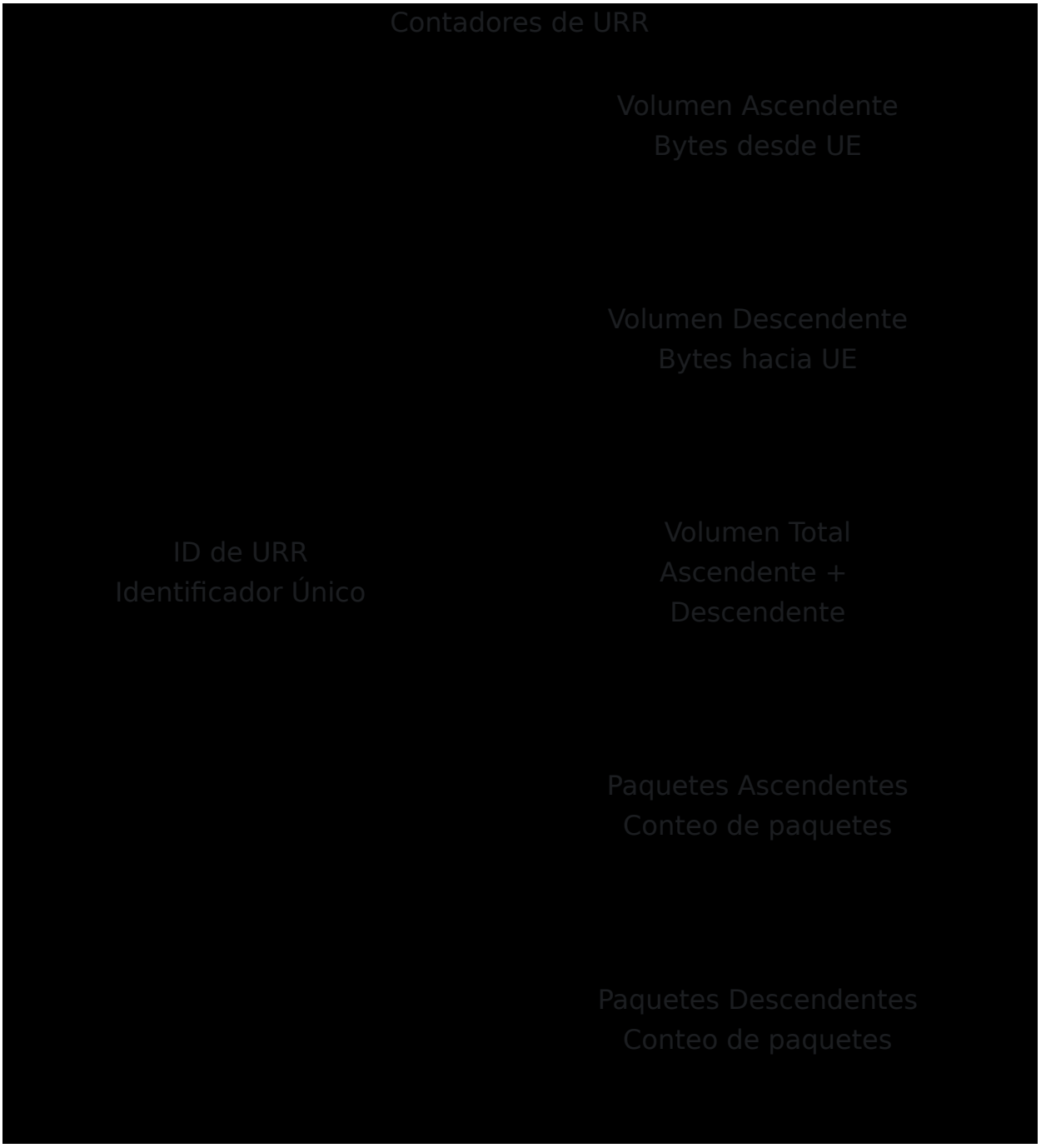
- La aplicación de GBR requiere programación de tráfico o encolado ponderado
- Puede implementarse utilizando capacidades de QoS de eBPF en futuras versiones

Reglas de Reporte de Uso (URR)

Propósito

Las URR rastrean volúmenes de datos para cobro, análisis y aplicación de políticas. Mantienen contadores de paquetes y bytes que se informan al SMF para los registros de cobro.

Estructura de URR



Seguimiento de Volumen

Volumen Ascendente:

- Bytes transmitidos desde UE a la Red de Datos
- Medido después de la desencapsulación de GTP-U
- Incluye encabezado IP y carga útil

Volumen Descendente:

- Bytes transmitidos desde la Red de Datos hacia UE
- Medido antes de la encapsulación de GTP-U
- Incluye encabezado IP y carga útil

Volumen Total:

- Suma de volúmenes ascendentes y descendentes
- Utilizado para el reporte de uso total

Disparadores de Reporte de Uso

Las URR pueden activar informes basados en:

Umbral de Volumen:

- Informar cuando el volumen exceda el límite configurado
- Ejemplo: Informar cada 1 GB de uso

Umbral de Tiempo:

- Informar en intervalos periódicos
- Ejemplo: Informar cada 5 minutos

Basado en Eventos:

- Informar sobre la terminación de la sesión
- Informar sobre el cambio de QoS
- Informar sobre la transferencia

Formato de Visualización de Volumen

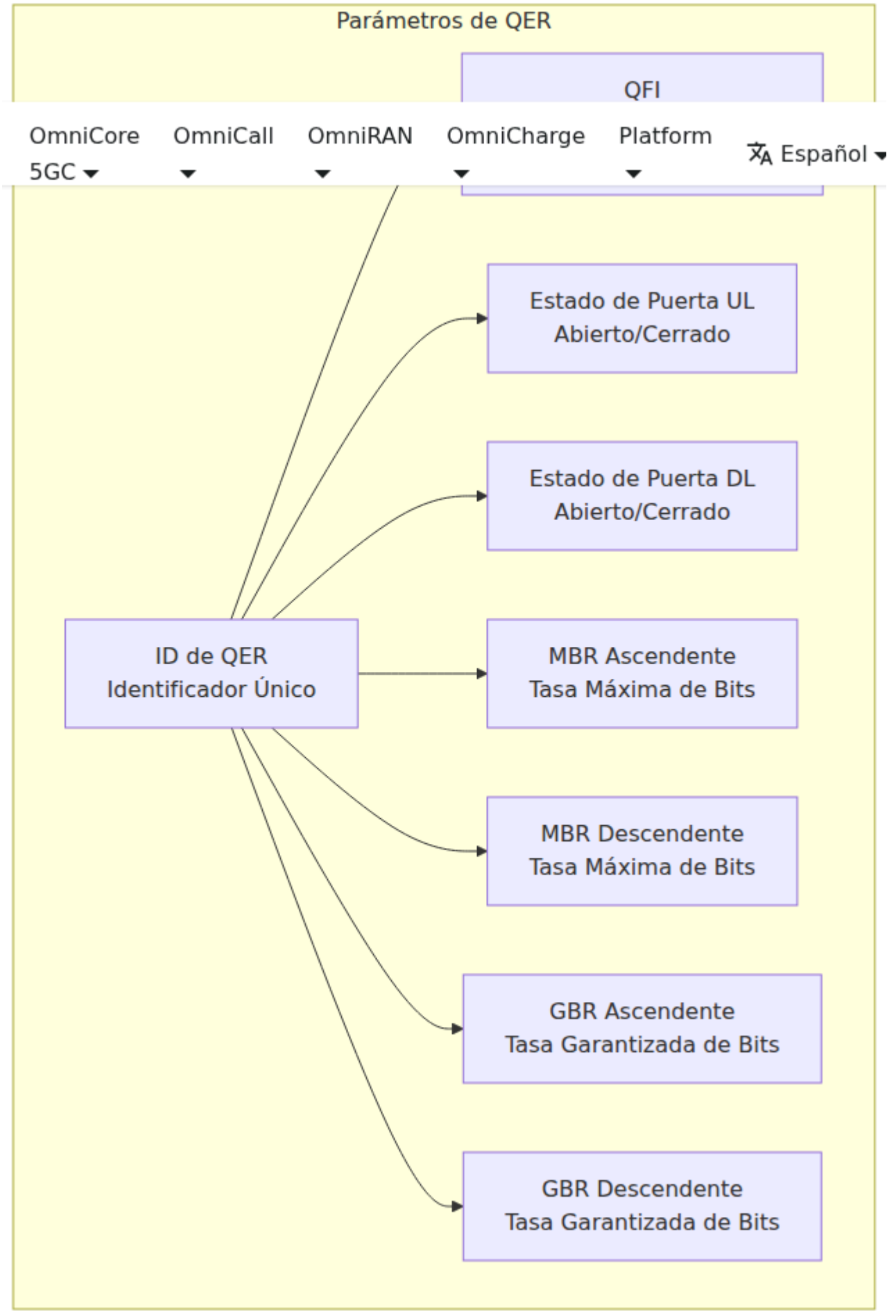
La interfaz web formatea automáticamente el volumen en unidades legibles para humanos:

Bytes	Visualización
0 - 1023	B (Bytes)
1024 - 1048575	KB (Kilobytes)
1048576 - 1073741823	MB (Megabytes)
1073741824 - 1099511627775	GB (Gigabytes)
1099511627776+	TB (Terabytes)

Ejemplo:

```
ID de URR: 0
Volumen Ascendente: 12.3 KB
Volumen Descendente: 9.0 KB
Volumen Total: 21.3 KB
```

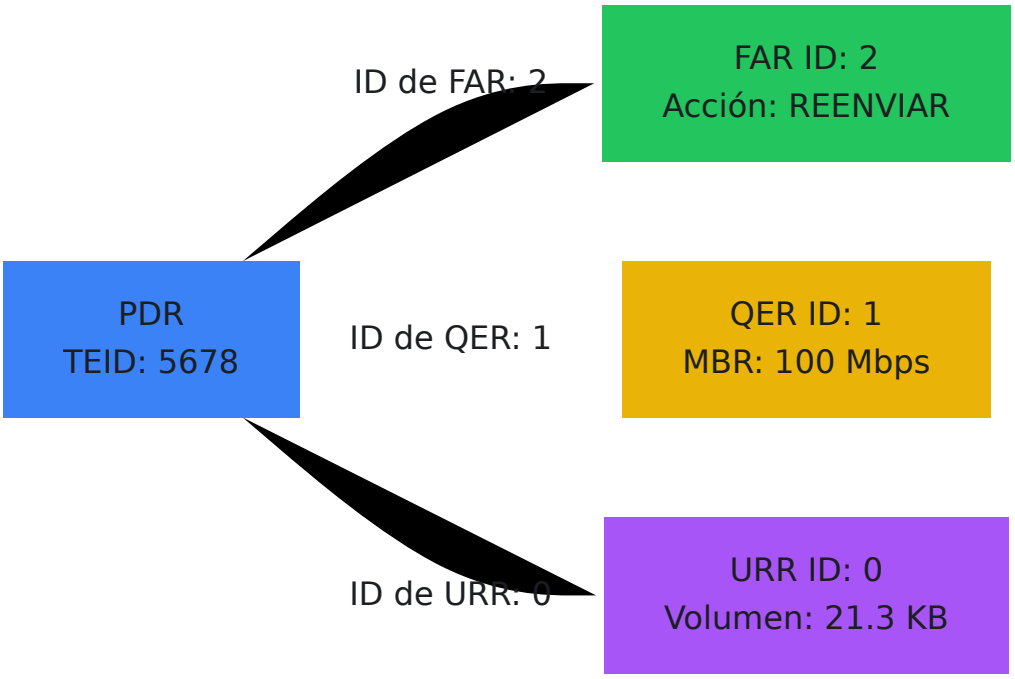
Flujo de Reporte de URR


Parámetros de QER
QFI
OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
Español
Estado de Puerta UL
Abierto/Cerrado
Estado de Puerta DL
Abierto/Cerrado
ID de QER
Identificador Único
MBR Ascendente
Tasa Máxima de Bits
MBR Descendente
Tasa Máxima de Bits
GBR Ascendente
Tasa Garantizada de Bits
GBR Descendente
Tasa Garantizada de Bits

Relaciones de Reglas

Cadena PDR → FAR → QER → URR

Cada PDR hace referencia a una FAR, que puede hacer referencia a una QER y una o más URRs.



Ejemplo de Configuración de Sesión

PDR Ascendente:

```
TEID: 5678
ID de FAR: 2
ID de QER: 1
IDs de URR: [0]
Eliminación de Encabezado Externo: Falso
```

PDR Descendente:

```
IP de UE: 10.45.0.1
ID de FAR: 1
ID de QER: 1
IDs de URR: [0]
Modo SDF: Sin SDF
```

FAR ID 1 (Descendente):

```
Acción: 2 (REENVIAR)
Creación de Encabezado Externo: Verdadero
IP Remota: 200.198.5.10
TEID: 5678
```

FAR ID 2 (Ascendente):

```
Acción: 2 (REENVIAR)
Creación de Encabezado Externo: Falso
```

QER ID 1:

```
QFI: 9
MBR Ascendente: 100000 kbps
MBR Descendente: 100000 kbps
GBR Ascendente: 0 kbps
GBR Descendente: 0 kbps
```

URR ID 0:

```
Volumen Ascendente: 12.3 KB
Volumen Descendente: 9.0 KB
Volumen Total: 21.3 KB
```

Operaciones Comunes

Ver Reglas para una Sesión

A través de la Página de Sesiones:

1. Navegar a Sesiones
2. Encontrar UE por IP o TEID
3. Hacer clic en "Expandir" para ver todas las reglas (PDR, FAR, QER, URR)

A través de la Página de Reglas:

1. Navegar a Reglas
2. Utilizar búsqueda por TEID (ascendente) o IP de UE (descendente) en la pestaña PDR
3. Anotar el ID de FAR, ID de QER, IDs de URR
4. Cambiar a las pestañas FAR/QER/URR para ver las reglas referenciadas

Habilitar/Deshabilitar Almacenamiento

Escenario: Durante la transferencia, almacenar paquetes para prevenir pérdidas

Pasos:

1. Navegar a Reglas → FARs
2. Ingresar ID de FAR en el campo de búsqueda
3. Hacer clic en "Buscar"
4. Si el almacenamiento está APAGADO, hacer clic en "Habilitar Almacenamiento"
5. Verificar que el bit 2 de la acción de FAR esté establecido (el valor de acción aumenta en 4)

Alternativa a través de la Página de Buffers:

1. Navegar a Buffers
2. Ver FARs con almacenamiento habilitado

3. Hacer clic en "Deshabilitar Buffer" cuando la transferencia se complete

Monitorear Cumplimiento de QoS

Verificar si el tráfico está siendo limitado en tasa:

1. Navegar a Reglas → QERs
2. Encontrar ID de QER asociado con la sesión de UE
3. Anotar valores de MBR Ascendente y MBR Descendente
4. Comparar con la tasa de crecimiento del volumen de URR

Calcular Rendimiento Promedio:

```
Rendimiento (kbps) = (Delta de Volumen en bytes × 8) / (Delta de
Tiempo en segundos × 1000)
```

Si el rendimiento se acerca al MBR, el tráfico está siendo limitado en tasa.

Rastrear Uso de Datos

Monitorear volúmenes de URR:

1. Navegar a Reglas → URRs
2. Ver volúmenes ascendentes, descendentes y totales
3. Ordenar por Volumen Total para encontrar los usuarios más altos
4. Actualizar periódicamente para observar el crecimiento del volumen

Casos de Uso:

- Verificar la integración de cobro
- Detectar uso anormal de datos
- Planificar capacidad basada en patrones de tráfico

Solución de Problemas

No Flujo de Tráfico

Verificar PDR:

1. Verificar que la PDR exista para TEID (ascendente) o IP de UE (descendente)
2. Confirmar que el ID de FAR sea válido
3. Verificar que los filtros SDF no estén bloqueando el tráfico

Verificar FAR:

1. Verificar que la acción de FAR sea REENVIAR (no DESCARTAR o solo BUFFER)
2. Confirmar que la creación de encabezado externo coincida con la dirección
3. Verificar que la IP Remota y el TEID sean correctos para el descenso

Verificar QER:

1. Verificar que el Estado de la Puerta esté Abierto (0)
2. Verificar que el MBR no sea demasiado restrictivo

Paquetes Siendo Descartados

Verificar Limitación de Tasa de QER:

1. Navegar a Reglas → QERs
2. Verificar que el MBR sea adecuado para la carga de tráfico
3. Verificar que el crecimiento del volumen de URR coincida con el rendimiento esperado

Verificar Acción de FAR:

1. Navegar a Reglas → FARs
2. Verificar que la acción sea REENVIAR, no DESCARTAR
3. Verificar que el almacenamiento no esté atascado en modo solo BUFFER

Problemas de Almacenamiento

Paquetes atascados en el buffer:

1. Navegar a la página de Buffers
2. Verificar la marca de tiempo del paquete más antiguo
3. Si > 30 segundos, la transferencia puede haber fallado
4. Limpiar o vaciar manualmente el buffer
5. Deshabilitar el almacenamiento en FAR

Desbordamiento de buffer:

1. Verificar total de paquetes vs. Máx. Total (predeterminado 100,000)
2. Verificar paquetes por FAR vs. Máx. Por FAR (predeterminado 10,000)
3. Limpiar buffers si están llenos
4. Investigar por qué el almacenamiento no se deshabilitó

URR No Rastreando

Contadores de volumen en cero:

1. Verificar que la PDR haga referencia al ID de URR
2. Verificar que los paquetes estén coincidiendo con la PDR
3. Verificar que la FAR esté reenviando (no descartando) paquetes
4. Confirmar que el ID de URR exista en el mapa de URR

Volumen no reportando al SMF:

1. Verificar la configuración de Informe de Sesión PFCP
2. Verificar los disparadores de reporte de URR (umbrales de volumen/tiempo)
3. Revisar registros de mensajes de Informe de Sesión PFCP

Documentación Relacionada

- **Guía de Operaciones de UPF** - Descripción general de la arquitectura y componentes de OmniUPF

- **Guía de Operaciones de Interfaz Web** - Uso del panel de control para visualización de reglas
- **Guía de Monitoreo** - Estadísticas y monitoreo de capacidad
- **Guía de Solución de Problemas** - Problemas comunes y diagnósticos

Guía de Solución de Problemas de OmniUPF

Tabla de Contenidos

1. Descripción General
2. Herramientas de Diagnóstico
3. Problemas de Instalación
4. Problemas de Configuración
5. Problemas de Asociación PFCP
6. Problemas de Procesamiento de Paquetes
7. Problemas de XDP y eBPF
8. Problemas de Rendimiento
9. Problemas Específicos del Hipervisor
10. Problemas de NIC y Controlador
11. Fallos en el Establecimiento de Sesiones
12. Problemas de Búfer

Descripción General

Esta guía proporciona procedimientos sistemáticos de solución de problemas para problemas comunes de OmniUPF. Cada sección incluye síntomas, pasos de diagnóstico, causas raíz y procedimientos de resolución.

Lista de Verificación Rápida de Diagnóstico

Antes de realizar una solución de problemas profunda, verifique:

```
# 1. Verifique que OmniUPF esté en ejecución
systemctl status omniupf

# 2. Verifique la asociación PFCP
curl http://localhost:8080/api/v1/upf_pipeline

# 3. Verifique que los mapas eBPF estén cargados
ls /sys/fs/bpf/

# 4. Verifique que el programa XDP esté adjunto
ip link show | grep -i xdp

# 5. Verifique los registros del kernel en busca de errores
dmesg | tail -50
journalctl -u omniupf -n 50
```

Herramientas de Diagnóstico

API REST de OmniUPF

Verificar el estado del UPF:

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

Verificar asociaciones PFCP:

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

Verificar el conteo de sesiones:

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

Verificar la capacidad del mapa eBPF:

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

Verificar estadísticas de paquetes:

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

Verificar estadísticas de XDP:

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

Inspección de Mapas eBPF

Listar todos los mapas eBPF:

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

Mostrar detalles del mapa:

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

Contar entradas en el mapa:

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

Inspección del Programa XDP

Verificar si el programa XDP está adjunto:

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

Listar todos los programas XDP:

```
bpftool net list
```

Mostrar detalles del programa XDP:

```
bpftool prog show
```

Volcar estadísticas de XDP:

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

Depuración de Red

Capturar tráfico PFCP en N4 (plano de control):

```
# PFCP no es procesado por XDP, tcpdump funciona normalmente
tcpdump -i eth0 -n udp port 8805 -w /tmp/pfcp_traffic.pcap
```

Capturar tráfico GTP-U en N3 (requiere captura fuera de banda):

```
# ADVERTENCIA: El tcpdump estándar en el host UPF NO PUEDE
capturar paquetes procesados por XDP!
# XDP procesa GTP-U antes de que la pila de red del kernel vea los
paquetes.

# Use captura fuera de banda en su lugar:
# 1. TAP de red entre gNB y UPF
# 2. Espejo de puerto de switch/SPAN para copiar tráfico N3
# 3. Espejo de puerto de switch virtual a VM de analizador

# En el host de análisis/monitoreo (NO en UPF):
# tcpdump -i <mirror_interface> -n udp port 2152 -w
/tmp/n3_capture.pcap

# O use la API de estadísticas para conteos de paquetes:
curl http://localhost:8080/api/v1/packet_stats
curl http://localhost:8080/api/v1/n3n6_stats
```

Monitorear contadores de paquetes:

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

Verificar tabla de enrutamiento:

```
ip route show
ip route get 10.45.0.100  # Verificar ruta para IP de UE
```

Verificar tabla ARP:

```
ip neigh show
```

Problemas de Instalación

Problema: "sistema de archivos eBPF no montado"

Síntomas:

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

Causa: sistema de archivos eBPF no montado

Resolución:

```
# Montar sistema de archivos eBPF
sudo mount bpffs /sys/fs/bpf -t bpf

# Hacer persistente (agregar a /etc/fstab)
echo "bpffs /sys/fs/bpf bpf defaults 0 0" | sudo tee -a /etc/fstab

# Verificar montaje
mount | grep bpf
```

Problema: Versión del kernel demasiado antigua

Síntomas:

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

Causa: versión del kernel de Linux por debajo del requisito mínimo

Resolución:

```
# Verificar versión del kernel
uname -r

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

# Verificar nuevo kernel
uname -r  # Debería ser >= 5.15.0
```

Problema: Dependencia de libbpf faltante

Síntomas:

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

Causa: biblioteca libbpf no instalada

Resolución:

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

# Verificar instalación
ldconfig -p | grep libbpf
```

Problemas de Configuración

Problema: Archivo de configuración inválido

Síntomas:

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

Causa: error de sintaxis YAML en el archivo de configuración

Resolución:

```
# Validar sintaxis YAML
cat config.yml | python3 -c "import yaml, sys;
yaml.safe_load(sys.stdin)"

# Problemas comunes:
# - Sangrado incorrecto (usar espacios, no tabulaciones)
# - Faltan dos puntos después de las claves
# - Cadenas no entrecomilladas con caracteres especiales
# - Elementos de lista sin guiones

# Ejemplo de YAML correcto:
cat > config.yml <<EOF
interface_name: [eth0]
xdp_attach_mode: generic
api_address: :8080
pfcp_address: :8805
EOF
```

Problema: Nombre de interfaz no encontrado

Síntomas:

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

Causa: la interfaz configurada no existe

Resolución:

```
# Listar todas las interfaces de red
ip link show

# Verificar estado de la interfaz
ip addr show eth0

# Si la interfaz tiene un nombre diferente, actualizar config.yml:
interface_name: [ens1f0]  # Usar el nombre real de la interfaz

# Para VMs, verificar el esquema de nombres de interfaz
ls /sys/class/net/
```

Problema: Puerto ya en uso

Síntomas:

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

Causa: el puerto 8080, 8805 o 9090 ya está vinculado por otro proceso

Resolución:

```
# Encontrar proceso que usa el puerto
sudo lsof -i :8080
sudo netstat -tulpn | grep :8080

# Matar proceso en conflicto
sudo kill <PID>

# O cambiar el puerto de OmniUPF en la configuración
api_address: :8081
pfcp_address: :8806
metrics_address: :9091
```

Problema: ID de nodo PFCP inválido

Síntomas:

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

Causa: el ID de nodo PFCP no es una dirección IPv4 válida

Resolución:

```yaml
# Correcto: usar dirección IP (no nombre de host)
pfcp_node_id: 10.100.50.241

# Incorrecto:
# pfcp_node_id: localhost
# pfcp_node_id: upf.example.com
```

Problemas de Asociación PFCP

Problema: No se establecieron asociaciones PFCP

Síntomas:

- La interfaz web muestra "No hay asociaciones"
- Los registros de SMF muestran "Fallo en la configuración de la asociación PFCP"

Diagnóstico:

```
# 1. Verificar si el servidor PFCP está escuchando
sudo netstat -ulpn | grep 8805

# 2. Verificar reglas de firewall
sudo iptables -L -n | grep 8805
sudo ufw status

# 3. Capturar tráfico PFCP
tcpdump -i any -n udp port 8805 -vv

# 4. Verificar asociaciones PFCP a través de la API
curl http://localhost:8080/api/v1/upf_pipeline
```

Causas Comunes y Resoluciones:

Firewall bloqueando PFCP

Resolución:

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

ID de nodo PFCP incorrecto

Resolución:

```
# Establecer ID de nodo PFCP en la IP correcta de la interfaz N4
pfcp_node_id: 10.100.50.241  # Debe coincidir con la IP en la red N4
```

Red inalcanzable para SMF

Resolución:

```
# Probar conectividad con SMF
ping <SMF_IP>

# Verificar enrutamiento hacia SMF
ip route get <SMF_IP>

# Agregar ruta si falta
sudo ip route add <SMF_NETWORK>/24 via <GATEWAY>
```

SMF configurado con IP de UPF incorrecta

Resolución:

- Verificar la configuración de SMF para la dirección de UPF
- Asegurarse de que SMF tenga configurada la IP de `pfcp_node_id` de UPF
- Verificar que SMF pueda enrutar a la red N4 de UPF

Problema: Fallos en el latido PFCP

Síntomas:

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

Diagnóstico:

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

# Monitorear registros de latidos
journalctl -u omniupf -f | grep heartbeat
```

Causas y Resoluciones:

Pérdida de paquetes en la red

Resolución:

```
# Verificar pérdida de paquetes hacia SMF
ping -c 100 <SMF_IP> | grep loss

# Si hay alta pérdida, investigar la red:
# - Verificar estado del enlace
# - Verificar salud del switch/router
# - Verificar congestión
```

Intervalo de latido demasiado agresivo

Resolución:

```yaml
# Aumentar el intervalo de latido
heartbeat_interval: 30  # Aumentar de 5 a 30 segundos
heartbeat_retries: 5    # Aumentar reintentos
heartbeat_timeout: 10   # Aumentar tiempo de espera
```

Problemas de Procesamiento de Paquetes

Problema: No fluyen paquetes (contadores RX/TX en 0)

Síntomas:

- La página de estadísticas muestra 0 paquetes RX/TX
- UE no puede establecer sesión de datos

Diagnóstico:

```
# 1. Verificar si el programa XDP está adjunto
ip link show eth0 | grep xdp

# 2. Verificar que la interfaz esté activa
ip link show eth0

# 3. Verificar estadísticas de paquetes (consciente de XDP)
# Nota: tcpdump no puede ver paquetes GTP-U procesados por XDP
curl http://localhost:8080/api/v1/packet_stats
```

Resoluciones:

Programa XDP no adjunto

Resolución:

```
# Reiniciar OmniUPF para volver a adjuntar XDP
sudo systemctl restart omniupf

# Verificar adjunto
ip link show eth0 | grep xdp
bpftool net list
```

Interfaz caída o sin enlace

Resolución:

```
# Activar interfaz
sudo ip link set eth0 up

# Verificar estado del enlace
ethtool eth0 | grep "Link detected"

# Si el enlace está caído, verificar conexión física o
configuración de red de VM
```

Interfaz configurada incorrectamente

Resolución:

```
# Actualizar config.yml con la interfaz correcta
interface_name: [ens1f0]  # Usar el nombre real de la interfaz de
'ip link show'
```

Problema: Paquetes recibidos pero no reenviados (alta tasa de caída)

Síntomas:

- Contadores RX en aumento pero contadores TX no
- Tasa de caída > 1%

Diagnóstico:

```
# Verificar estadísticas de caída
curl http://localhost:8080/api/v1/xdp_stats | jq '.drop'

# Verificar estadísticas de ruta
curl http://localhost:8080/api/v1/packet_stats | jq '.route_stats'

# Monitorear caídas de paquetes
watch -n 1 'curl -s http://localhost:8080/api/v1/packet_stats | jq
".total_rx, .total_tx, .total_drop"'
```

Causas Comunes:

Sin coincidencia de PDR (TEID o IP de UE desconocidos)

Resolución:

```
# Verificar si existen sesiones
curl http://localhost:8080/api/v1/sessions

# Si no hay sesiones, verificar:
# - La asociación PFCP está establecida
# - SMF ha creado sesiones
# - El establecimiento de la sesión fue exitoso

# Verificar entradas en el mapa PDR
bpftool map dump name pdr_map_teid_ip | grep -c key
bpftool map dump name pdr_map_downlin | grep -c key
```

Fallos de enrutamiento

Resolución:

```
# Verificar fallos de búsqueda FIB
curl http://localhost:8080/api/v1/packet_stats | jq '.route_stats'

# Probar enrutamiento para IP de UE
ip route get 10.45.0.100

# Agregar ruta faltante
sudo ip route add 10.45.0.0/16 dev eth1  # Ruta del grupo de UE a
N6
```

Limitación de tasa QER

Síntomas:

- Rendimiento inferior al esperado
- Tráfico limitado a una tasa específica
- Contadores de volumen URR muestran comportamiento de meseta
- Contadores de caída de XDP en aumento durante ráfagas de tráfico

Diagnóstico:

1. **Verificar MBR configurado para la sesión**:

```
# Encontrar el ID de QER de la sesión
curl http://localhost:8080/api/v1/pfcp_sessions | jq '.data[] |
select(.ue_ip == "10.45.0.1")'

# Buscar la configuración de QER
curl http://localhost:8080/api/v1/qer_map | jq '.data[] |
select(.qer_id == 1)'
```

2. **Verificar estado de la puerta**:

```
# El estado de la puerta debe ser 0 (ABIERTA) tanto para uplink
como para downlink
curl http://localhost:8080/api/v1/qer_map | jq '.data[] |
{qer_id, ul_gate: .ul_gate_status, dl_gate: .dl_gate_status}'
```

3. **Calcular rendimiento real desde URR**:

```
# Consultar contadores de volumen URR en dos puntos en el
tiempo
curl http://localhost:8080/api/v1/urr_map | jq '.data[] |
select(.urr_id == 0)'

# Calcular rendimiento (manual):
# throughput_kbps = (volume_delta_bytes × 8) /
time_delta_seconds / 1000
```

4. **Comparar MBR vs. rendimiento real**:
    - Rendimiento esperado ≈ 95-98% de MBR (debido a la sobrecarga del protocolo)
    - Si el rendimiento está significativamente por debajo de MBR, verificar otros cuellos de botella
    - Si el rendimiento coincide exactamente con MBR, la limitación de tasa está funcionando como se esperaba

Resolución:

- **Si MBR es demasiado bajo**: Solicitar a SMF que actualice QER con un MBR más alto a través de PFCP Session Modification
- **Si la puerta está cerrada**: Investigar por qué SMF cerró la puerta (política, cuota o error)
- **Si la limitación de tasa es inesperada**: Verificar la configuración de política de SMF y el perfil QoS

Comprendiendo la Aplicación de MBR:

OmniUPF utiliza un algoritmo de ventana deslizante para hacer cumplir los límites de MBR con precisión de nanosegundos en la ruta de datos eBPF. Consulte Guía de Gestión de Reglas - Mecanismo de Aplicación de MBR para una explicación detallada de:

- Cómo el tamaño y la tasa de los paquetes determinan las decisiones de caída
- Por qué el rendimiento observado difiere del MBR configurado
- Limitación de tasa por dirección (uplink/downlink)
- Comportamiento de ventana deslizante de 5 ms

Escenarios Comunes:

- **Llamadas VoIP cayendo**: Verificar si el MBR es suficiente para la tasa de bits del códec (G.711 = ~80 kbps)
- **Buffering en streaming de video**: Asegurarse de que MBR > tasa de bits de video + sobrecarga (1080p = ~5-10 Mbps)
- **Tráfico de ráfagas**: Se permiten pequeñas ráfagas dentro de la ventana de 5 ms, el tráfico sostenido está limitado por la tasa

---

Problema: Tráfico unidireccional (uplink funciona, downlink no)

Síntomas:

- Paquetes RX N3 pero no paquetes TX N3 (problema de downlink)
- Paquetes RX N6 pero no paquetes TX N6 (problema de uplink)

Diagnóstico:

```
# Verificar estadísticas de interfaz N3/N6 (método consciente de
XDP)
curl http://localhost:8080/api/v1/n3n6_stats
curl http://localhost:8080/api/v1/packet_stats

# Nota: tcpdump estándar no puede capturar tráfico GTP-U procesado
por XDP
# Use la API de estadísticas o xdpdump para análisis de tráfico
# Consulte la sección "Captura de Paquetes con XDP" para más
detalles
```

Fallo de Uplink (RX N3, sin TX N6):

Causa: Sin acción FAR o problema de enrutamiento hacia N6

Resolución:

```
# Verificar que FAR tenga acción FORWARD
curl http://localhost:8080/api/v1/sessions | jq '.[].fars[] |
select(.applied_action == 2)'

# Verificar que exista ruta N6
ip route get 8.8.8.8  # Probar ruta hacia internet

# Agregar ruta predeterminada si falta
sudo ip route add default via <N6_GATEWAY> dev eth1
```

Fallo de Downlink (RX N6, sin TX N3):

Causa: Sin PDR de downlink o falta de encapsulación GTP

Resolución:

```
# Verificar que exista PDR de downlink para la IP de UE
curl http://localhost:8080/api/v1/sessions | jq '.[].pdrs[] |
select(.pdi.ue_ip_address)'

# Verificar que FAR tenga OUTER_HEADER_CREATION
curl http://localhost:8080/api/v1/sessions | jq '.[].fars[] |
.outer_header_creation'

# Verificar conectividad con gNB
ping <GNB_N3_IP>
```

Problemas de XDP y eBPF

Para obtener detalles sobre la configuración de XDP, selección de modo y solución de problemas, consulte la Guía de Modos XDP.

Problema: El programa XDP no se pudo cargar

Síntomas:

```
ERRO[0000] failed to load XDP program: invalid argument
```

Diagnóstico:

```
# Verificar soporte de XDP en el kernel
grep XDP /boot/config-$(uname -r)

# Debería mostrar:
# CONFIG_XDP_SOCKETS=y
# CONFIG_BPF=y
# CONFIG_BPF_SYSCALL=y

# Verificar dmesg para error detallado
dmesg | grep -i bpf
```

Causas y Resoluciones:

El kernel carece de soporte para XDP

Resolución:

```
# Recompilar el kernel con soporte para XDP o actualizar a un
kernel más nuevo
# Ubuntu 22.04+ tiene XDP habilitado por defecto
sudo apt install linux-generic-hwe-22.04
sudo reboot
```

Fallo de verificación del programa XDP

Resolución:

```
# Verificar registros de OmniUPF para errores de verificador
journalctl -u omniupf | grep verifier

# Problemas comunes:
# - Complejidad de eBPF excede límites (aumentar límites del
kernel)
# - Acceso a memoria inválido (error en el código eBPF)

# Aumentar el nivel de registro del verificador eBPF para
depuración
sudo sysctl kernel.bpf_stats_enabled=1
```

Problema: Contador de abortos de XDP en aumento

Síntomas:

- Las estadísticas de XDP muestran abortos > 0
- Aumento de caídas de paquetes

Diagnóstico:

```
# Verificar contador de abortos de XDP
curl http://localhost:8080/api/v1/xdp_stats | jq '.aborted'

# Monitorear estadísticas de XDP
watch -n 1 'curl -s http://localhost:8080/api/v1/xdp_stats'
```

Causa: el programa eBPF encontró un error en tiempo de ejecución

Resolución:

```
# Verificar registros del kernel para errores de eBPF
dmesg | grep -i bpf

# Reiniciar OmniUPF para recargar el programa eBPF
sudo systemctl restart omniupf

# Si el problema persiste, habilitar registro eBPF (requiere
recompilación):
# Compilar OmniUPF con BPF_ENABLE_LOG=1
```

Problema: Mapa eBPF lleno (capacidad agotada)

Síntomas:

- Fallos en el establecimiento de sesiones
- Capacidad del mapa al 100%

Diagnóstico:

```
# Verificar capacidad del mapa
curl http://localhost:8080/api/v1/map_info | jq '.[] | {map_name,
capacity, used, usage_percent}'

# Identificar mapas llenos
curl http://localhost:8080/api/v1/map_info | jq '.[] |
select(.usage_percent > 90)'
```

Mitigación Inmediata:

```
# 1. Identificar sesiones obsoletas
curl http://localhost:8080/api/v1/sessions | jq '.[] | {seid,
uplink_teid, created_at}'

# 2. Solicitar a SMF que elimine sesiones antiguas
# (a través de la interfaz de administración de SMF o API)

# 3. Monitorear disminución del uso del mapa
watch -n 5 'curl -s http://localhost:8080/api/v1/map_info | jq ".
[] | select(.map_name==\"pdr_map_downlin\") | .usage_percent"'
```

Resolución a Largo Plazo:

```
# Aumentar capacidad del mapa en config.yml
max_sessions: 200000  # Aumentar de 100000

# O establecer tamaños individuales de mapa
pdr_map_size: 400000
far_map_size: 400000
qer_map_size: 200000
```

Importante: Cambiar tamaños de mapa requiere reiniciar OmniUPF y **borra todas las sesiones existentes**.

Problemas de Rendimiento

Problema: Bajo rendimiento (por debajo de lo esperado)

Síntomas:

- Rendimiento < 1 Gbps a pesar de NIC capaz
- Alta utilización de CPU

Diagnóstico:

```
# Verificar tasa de paquetes
curl http://localhost:8080/api/v1/packet_stats | jq '.total_rx,
.total_tx'

# Verificar estadísticas de NIC
ethtool -S eth0 | grep -i drop

# Verificar modo XDP
ip link show eth0 | grep xdp
```

Resoluciones:

Usando modo XDP genérico

Resolución:

```
# Cambiar a modo nativo para mejor rendimiento
xdp_attach_mode: native  # Requiere NIC/controlador compatible con
XDP
```

Cuello de botella de un solo núcleo

Resolución:

```
# Habilitar RSS (Recepción de Distribución de Carga) en NIC
ethtool -L eth0 combined 4  # Usar 4 colas RX/TX

# Verificar RSS habilitado
ethtool -l eth0

# Fijar interrupciones a CPUs específicas
# Ver /proc/interrupts y usar irqbalance o afinidad manual
```

Bloat de búfer

Resolución:

```yaml
# Reducir límites de búfer para disminuir latencia
buffer_max_packets: 5000
buffer_packet_ttl: 15
```

Problema: Alta latencia

Síntomas:

- Latencia de ping > 50ms
- Degradación de la experiencia del usuario

Diagnóstico:

```
# Probar latencia a UE
ping -c 100 <UE_IP> | grep avg

# Verificar paquetes en búfer
curl http://localhost:8080/api/v1/upf_buffer_info | jq
'.total_packets_buffered'

# Verificar rendimiento de caché de ruta
curl http://localhost:8080/api/v1/packet_stats | jq '.route_stats'
```

Resoluciones:

Paquetes siendo almacenados en búfer excesivamente

Resolución:

```
# Verificar por qué los paquetes están en búfer
curl http://localhost:8080/api/v1/upf_buffer_info | jq '.buffers[]
| {far_id, packet_count, direction}'

# Limpiar búferes si están atascados
# (reiniciar OmniUPF o activar modificación de sesión PFCP para
aplicar FAR)
```

Latencia de búsqueda FIB

Resolución:

```
# Asegurarse de que la caché de ruta esté habilitada (opción de
tiempo de compilación)
# Compilar con BPF_ENABLE_ROUTE_CACHE=1

# Optimizar tabla de enrutamiento
# Usar menos rutas, más específicas en lugar de muchas rutas
pequeñas
```

Problema: Caídas de paquetes bajo carga

Síntomas:

- La tasa de caída aumenta con el tráfico
- Errores RX en NIC

Diagnóstico:

```
# Verificar errores en NIC
ethtool -S eth0 | grep -E "drop|error|miss"

# Verificar tamaño del búfer de anillo
ethtool -g eth0

# Monitorear caídas en tiempo real
watch -n 1 'ethtool -S eth0 | grep -E "drop|miss"'
```

Resolución:

```
# Aumentar tamaño del búfer de anillo RX
ethtool -G eth0 rx 4096

# Aumentar tamaño del búfer de anillo TX
ethtool -G eth0 tx 4096

# Verificar nuevas configuraciones
ethtool -g eth0
```

Problemas Específicos del Hipervisor

Para instrucciones de configuración paso a paso del hipervisor, consulte la Guía de Modos XDP.

Proxmox: XDP no funciona en VM

Síntomas:

- No se puede adjuntar el programa XDP en modo nativo
- Solo funciona el modo genérico

Causa: VM usando red en puente sin SR-IOV

Resolución:

Opción 1: Usar modo genérico (más simple)

```
xdp_attach_mode: generic
```

Opción 2: Configurar paso a través de SR-IOV

```
# En el host Proxmox:
# 1. Habilitar IOMMU
nano /etc/default/grub
# Agregar: intel_iommu=on iommu=pt
update-grub
reboot

# 2. Crear VFs
echo 4 > /sys/class/net/eth0/device/sriov_numvfs

# 3. Asignar VF a VM en la interfaz de usuario de Proxmox
# Hardware → Agregar → Dispositivo PCI → Seleccionar VF

# En VM:
interface_name: [ens1f0]  # VF de SR-IOV
xdp_attach_mode: native
```

VMware: Modo promiscuo requerido

Síntomas:

- Paquetes no recibidos por OmniUPF

Causa: vSwitch bloqueando direcciones MAC no coincidentes

Resolución:

```
# Habilitar modo promiscuo en vSwitch (en vSphere Client):
# 1. Seleccionar vSwitch → Editar Configuración
# 2. Seguridad → Modo Promiscuo: Aceptar
# 3. Seguridad → Cambios de Dirección MAC: Aceptar
# 4. Seguridad → Transmitidos Forjados: Aceptar
```

VirtualBox: Rendimiento muy bajo

Síntomas:

- Rendimiento < 100 Mbps

Causa: VirtualBox no soporta SR-IOV o XDP nativo

Resolución:

```
# Usar modo genérico (única opción)
xdp_attach_mode: generic

# Optimizar configuraciones de VirtualBox:
# - Usar adaptador VirtIO-Net (si está disponible)
# - Habilitar "Permitir Todo" en modo promiscuo
# - Asignar más núcleos de CPU a la VM
# - Usar red en puente en lugar de NAT

# Considerar migrar a KVM/Proxmox para mejor rendimiento
```

Problemas de NIC y Controlador

Problema: El controlador NIC no soporta XDP

Síntomas:

```
ERRO[0000] failed to attach XDP program: operation not supported
```

Diagnóstico:

```
# Verificar controlador NIC
ethtool -i eth0 | grep driver

# Verificar si el controlador soporta XDP
modinfo <driver_name> | grep -i xdp

# Listar interfaces capaces de XDP
ip link show | grep -B 1 "xdpgeneric\|xdpdrv\|xdpoffload"
```

Resolución:

Opción 1: Usar modo genérico

```
xdp_attach_mode: generic
```

Opción 2: Actualizar controlador NIC

```
# Verificar actualizaciones de controlador (Ubuntu)
sudo apt update
sudo apt install linux-modules-extra-$(uname -r)

# O instalar controlador específico del proveedor
# Ejemplo para Intel:
# Descargar de https://downloadcenter.intel.com/
```

Opción 3: Reemplazar NIC

```
# Usar NIC capaz de XDP:
# - Intel X710, E810
# - Mellanox ConnectX-5, ConnectX-6
# - Broadcom BCM57xxx (controlador bnxt_en)
```

Problema: El controlador falla o causa pánicos en el kernel

Síntomas:

- Pánico en el kernel después de adjuntar XDP
- NIC deja de responder

Diagnóstico:

```
# Verificar registros del kernel
dmesg | tail -100

# Verificar errores de controlador
journalctl -k | grep -E "BUG:|panic:"
```

Resolución:

```
# 1. Actualizar kernel y controladores
sudo apt update
sudo apt upgrade
sudo reboot

# 2. Deshabilitar offload de XDP (usar solo nativo)
xdp_attach_mode: native

# 3. Usar modo genérico como solución alternativa
xdp_attach_mode: generic

# 4. Informar error al proveedor de NIC o al equipo del kernel de
Linux
```

Fallos en el Establecimiento de Sesiones

Problema: Fallo en el establecimiento de sesiones

Síntomas:

- SMF informa fallo en el establecimiento de sesiones
- UE no puede establecer sesión PDU

Consulte Referencia de Códigos de Causa PFCP para escenarios comunes de fallo y resoluciones.

Diagnóstico:

```
# Verificar registros de OmniUPF para errores de sesión
journalctl -u omniupf | grep -i "session establishment"

# Verificar conteo de sesiones PFCP
curl http://localhost:8080/api/v1/sessions | jq 'length'

# Capturar tráfico PFCP durante el establecimiento de sesión
tcpdump -i any -n udp port 8805 -w /tmp/pfcp_session.pcap
```

Causas Comunes:

Capacidad del mapa llena

Resolución:

```
# Verificar uso del mapa
curl http://localhost:8080/api/v1/map_info | jq '.[] |
select(.usage_percent > 90)'

# Aumentar capacidad (ver sección de mapa eBPF lleno arriba)
```

Parámetros de PDR/FAR inválidos

Resolución:

```
# Verificar registros de OmniUPF para errores de validación
journalctl -u omniupf | grep -E "invalid|error" | tail -20

# Problemas comunes:
# - Dirección IP de UE inválida (0.0.0.0 o duplicada)
# - TEID inválido (0 o duplicado)
# - FAR faltante para PDR
# - Acción FAR inválida

# Verificar configuración de SMF y parámetros de sesión
```

Función no soportada (UEIP/FTUP)

Resolución:

```
# Habilitar funciones requeridas si es necesario
feature_ueip: true  # Asignación de IP de UE por UPF
ueip_pool: 10.60.0.0/16

feature_ftup: true  # Asignación de F-TEID por UPF
teid_pool: 100000
```

Problemas de Búfer

Problema: Paquetes atascados en el búfer

Síntomas:

- Contador de paquetes en búfer en aumento
- Paquetes no entregados después de la transferencia

Diagnóstico:

```
# Verificar estadísticas de búfer
curl http://localhost:8080/api/v1/upf_buffer_info

# Verificar búferes individuales de FAR
curl http://localhost:8080/api/v1/upf_buffer_info | jq '.buffers[]
| {far_id, packet_count, oldest_packet_ms}'

# Monitorear tamaño del búfer
watch -n 5 'curl -s http://localhost:8080/api/v1/upf_buffer_info |
jq ".total_packets_buffered"'
```

Causas y Resoluciones:

FAR nunca actualizado a FORWARD

Causa: SMF nunca envió PFCP Session Modification para aplicar FAR

Resolución:

```
# Verificar estado de FAR
curl http://localhost:8080/api/v1/sessions | jq '.[].fars[] |
{far_id, applied_action}'

# Acción BUFF = 1 (almacenamiento)
# Acción FORW = 2 (reenviar)

# Si está atascado en estado BUFF, solicitar a SMF que:
# - Envíe PFCP Session Modification Request
# - Actualice FAR con acción FORW
```

TTL del búfer expirado

Causa: Paquetes expiraron antes de la actualización de FAR

Resolución:

```
# Aumentar TTL del búfer
buffer_packet_ttl: 60  # Aumentar de 30 a 60 segundos
```

Desbordamiento del búfer

Causa: Demasiados paquetes almacenados por FAR

Resolución:

```
# Aumentar límites de búfer
buffer_max_packets: 20000  # Por FAR
buffer_max_total: 200000   # Límite global
```

Depuración Avanzada

Habilitar Registro de Depuración

```
logging_level: debug  # trace | debug | info | warn | error
```

```
# Reiniciar OmniUPF con registro de depuración
sudo systemctl restart omniupf

# Monitorear registros en tiempo real
journalctl -u omniupf -f --output cat
```

Trazado de Programas eBPF

```
# Trazar ejecución de programas eBPF (requiere bpftrace)
sudo bpftrace -e 'tracepoint:xdp:* { @[probe] = count(); }'

# Trazar operaciones de mapa
sudo bpftrace -e 'tracepoint:bpf:bpf_map_lookup_elem {
printf("%s\n", str(args->map_name)); }'
```

Captura de Paquetes con XDP

Entendiendo las Limitaciones de Captura de Paquetes de XDP:

XDP procesa paquetes **antes** de que la pila de red del kernel, por lo que el `tcpdump` estándar **no puede ver el tráfico procesado por XDP**. Los paquetes GTP-U (puerto UDP 2152) en N3 son procesados por XDP y no aparecerán en tcpdump en el host UPF.

Métodos Recomendados para Análisis de Tráfico:

```
# Método 1: Usar API de estadísticas para monitoreo (RECOMENDADO)
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

# Método 2: Capturar tráfico PFCP (no afectado por XDP)
tcpdump -i any -n udp port 8805 -w /tmp/pfcp.pcap

# Método 3: Captura de paquetes fuera de banda (RECOMENDADO para
GTP-U)
# Usar TAP de red o espejo de puerto de switch para capturar
tráfico
# Ejemplos:
# - TAP físico entre gNB y UPF
# - Espejo de puerto de switch copiando tráfico N3 al analizador
# - Espejo de puerto de switch virtual en hipervisor
#
# En el host de captura (NO el UPF):
# tcpdump -i <mirror_interface> -n udp port 2152 -w
/tmp/n3_mirror.pcap
```

Ejemplos de Configuración de Captura Fuera de Banda:

Red Física:

```
# Usar un TAP de red o configurar espejo de puerto de switch
# Ejemplo: configuración SPAN de switch Cisco
(config)# monitor session 1 source interface Gi1/0/1
(config)# monitor session 1 destination interface Gi1/0/24

# En el host de monitoreo conectado a Gi1/0/24:
tcpdump -i eth0 -n udp port 2152 -w /tmp/n3_capture.pcap
```

Entorno Virtual (VMware, KVM, etc.):

```
# Configurar espejo de puerto de switch virtual para enviar
tráfico de UPF a VM de analizador
# Ejemplo: puente de Linux con tcpdump en VM diferente
# En hipervisor, espejar la interfaz N3 de UPF a la interfaz de
analizador

# En VM de analizador:
tcpdump -i eth1 -n udp port 2152 -w /tmp/n3_virtual.pcap
```

Por qué se Requiere Captura Fuera de Banda:

- XDP omite completamente la pila de red del kernel
- Los paquetes son procesados en el controlador NIC o hardware
- tcpdump basado en host ve los paquetes DESPUÉS del procesamiento de XDP (demasiado tarde)
- La captura fuera de banda ve el tráfico en crudo antes del procesamiento de UPF

Lo Que PUEDE Capturar en el Host UPF:

- □ Tráfico PFCP (UDP 8805) - plano de control, no procesado por XDP
- □ Respuestas de API y métricas
- □ Tráfico GTP-U (UDP 2152) - plano de datos, procesado por XDP

Obtener Ayuda

Si los pasos de solución de problemas no resuelven su problema:

1. **Recopilar información de diagnóstico**:

```
# Información del sistema
uname -a
cat /etc/os-release

# Información de 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

# Registros
journalctl -u omniupf --since "1 hour ago" > /tmp/omniupf.log
dmesg > /tmp/dmesg.log

# Información de red
ip addr > /tmp/network.txt
ip route >> /tmp/network.txt
ethtool eth0 >> /tmp/network.txt
```

2. **Informar el problema** con:
    - Versión de OmniUPF
    - Versión del kernel de Linux
    - Diagrama de topología de red
    - Archivo de configuración (redactar información sensible)
    - Extractos de registro relevantes
    - Pasos para reproducir

Documentación Relacionada

- **Guía de Configuración** - Parámetros de configuración y ejemplos
- **Guía de Arquitectura** - Internos de eBPF/XDP y ajuste de rendimiento
- **Guía de Monitoreo** - Estadísticas, capacidad y alertas
- **Referencia de Métricas** - Métricas de Prometheus para solución de problemas
- **Códigos de Causa PFCP** - Códigos de error PFCP y solución de problemas
- **Guía de Gestión de Reglas** - Conceptos de PDR, FAR, QER, URR

- **Guía de Operaciones** - Arquitectura y descripción general de UPF

OmniUPF Jardín Cerrado / Redirección Fuera de Crédito

Tabla de Contenidos

1. Descripción General
2. Arquitectura
3. Flujo de Señalización PFCP
4. Detección de Portal Captivo
5. Configuración
6. Gestión de Lista Blanca
7. URLs de Redirección por Sesión
8. API
9. Métricas de Prometheus
10. Solución de Problemas

Descripción General

La función de Jardín Cerrado proporciona **aplicación nativa de la falta de crédito** directamente en el UPF, eliminando la necesidad de sistemas de aplicación externos (listas de direcciones MikroTik, reglas de mangle, DNAT).

Cuando un suscriptor se queda sin crédito, el SMF envía una Modificación de Sesión PFCP con un FAR que contiene `redirect_information`. OmniUPF intercepta todo el tráfico de esa sesión en el espacio de usuarios y aplica una experiencia de portal cautivo:

- **Las consultas DNS** son suplantadas para devolver la IP del servidor del portal, activando la detección de portal cautivo en todos los dispositivos

principales (Apple, Android, Windows)

- **El tráfico hacia el portal y las IPs en la lista blanca** (procesadores de pago, servicios CAPTCHA) se reenvía normalmente para que el suscriptor pueda recargar
- **Todo el demás tráfico** se descarta silenciosamente

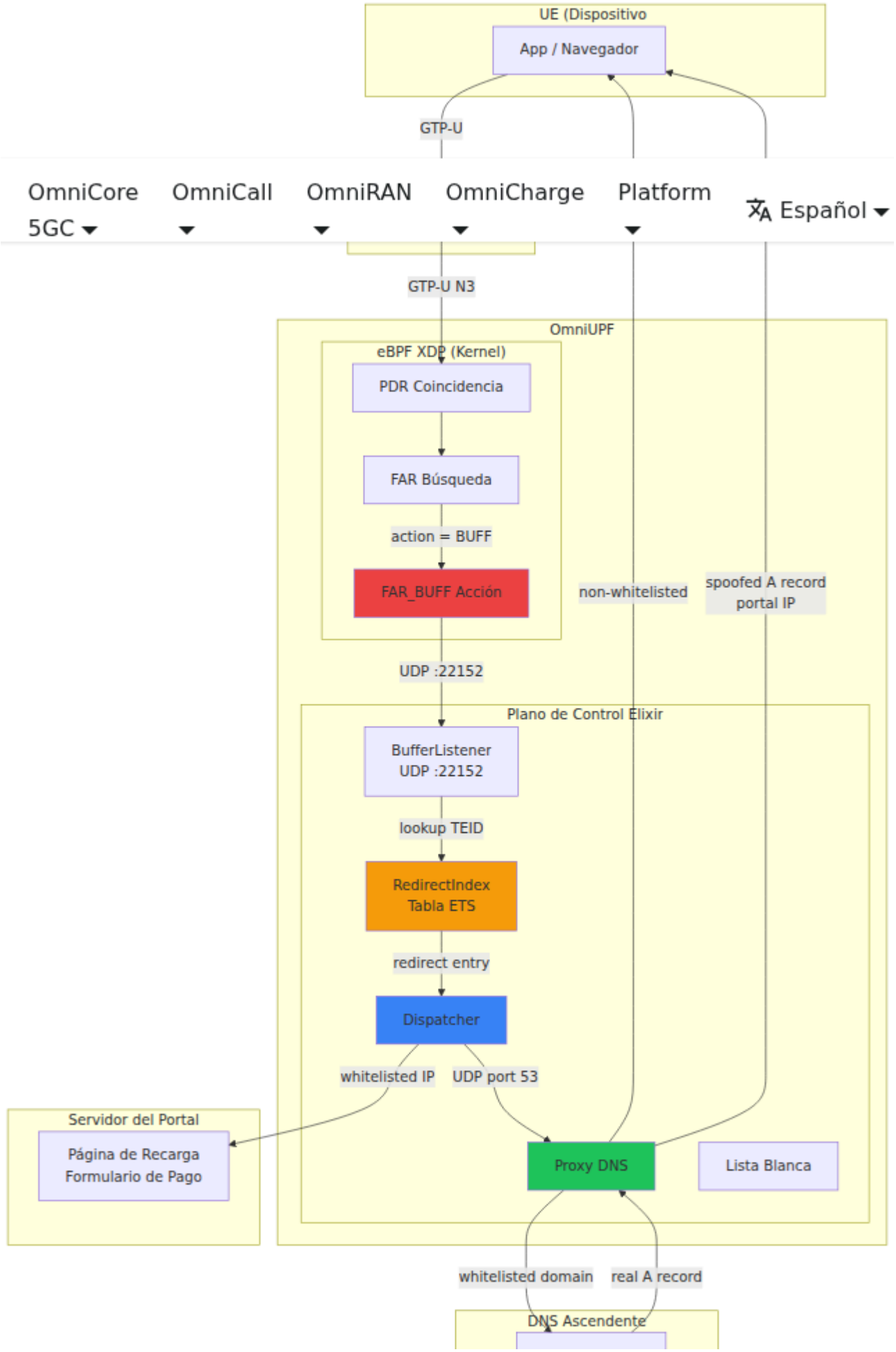
El suscriptor ve un aviso de portal cautivo y es dirigido a una página de recarga/pago. Una vez que se restaura el crédito, el SMF actualiza el FAR de nuevo a FORW y el reenvío normal se reanuda inmediatamente.

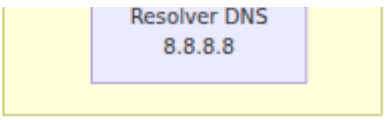
Puntos Clave de Diseño

- **No se requieren cambios en eBPF** -- reutiliza la acción existente `FAR_BUFF` para redirigir paquetes al espacio de usuarios
- **Redirección por sesión** -- cada sesión puede tener una IP de portal y una URL de redirección diferentes, determinadas por el IE `redirect_information` del SMF
- **La lista blanca vive en el UPF** -- el SMF solo dice "redirigir esta sesión"; el UPF decide qué tráfico permitir
- **Intercepción solo de enlace ascendente** -- el FAR de enlace descendente permanece FORW para que las respuestas del portal lleguen al UE a través de la ruta de encapsulación GTP normal

---

Arquitectura


UE (Dispositivo
App / Navegador
GTP-U
OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
Español
GTP-U N3
OmniUPF
eBPF XDP (Kernel)
PDR Coincidencia
FAR Búsqueda
action = BUFF
FAR_BUFF Acción
non-whitelisted
spoofed A record
portal IP
UDP :22152
Plano de Control Elixir
BufferListener
UDP :22152
lookup TEID
RedirectIndex
Tabla ETS
redirect entry
Dispatcher
whitelisted IP
UDP port 53
Servidor del Portal
Página de Recarga
Formulario de Pago
Proxy DNS
Lista Blanca
whitelisted domain
real A record
DNS Ascendente



Flujo de Paquetes

1. **UE envía paquete de enlace ascendente** (consulta DNS, solicitud HTTP, etc.) a través de GTP-U al UPF
2. **Coincidencia de PDR eBPF** encuentra el PDR coincidente, busca el FAR
3. **La acción del FAR es BUFF** (sobrescrita de FORW cuando la redirección está activa) -- eBPF envía el paquete al BufferListener en el puerto UDP 22152
4. **BufferListener** extrae el TEID, verifica la tabla ETS RedirectIndex
5. Si el TEID está en el RedirectIndex: **Dispatcher** procesa el paquete IP interno
6. **Árbol de decisión**:
   - Consulta DNS para dominio no en la lista blanca: suplantar A record con IP del portal
   - Consulta DNS para dominio en la lista blanca: reenviar al resolver real, almacenar IPs resueltas
   - Tráfico hacia IP del portal o IP en la lista blanca: reenviar a través de socket crudo
   - Todo lo demás: descartar silenciosamente
7. **Respuestas DNS/GTP-U** se envían de vuelta al UE a través de la ruta GTP-U de enlace descendente (encapsuladas con el TEID DL y enviadas al gNB)

Manejo de Loopback N9 (Sesión Dual SGW + PGW)

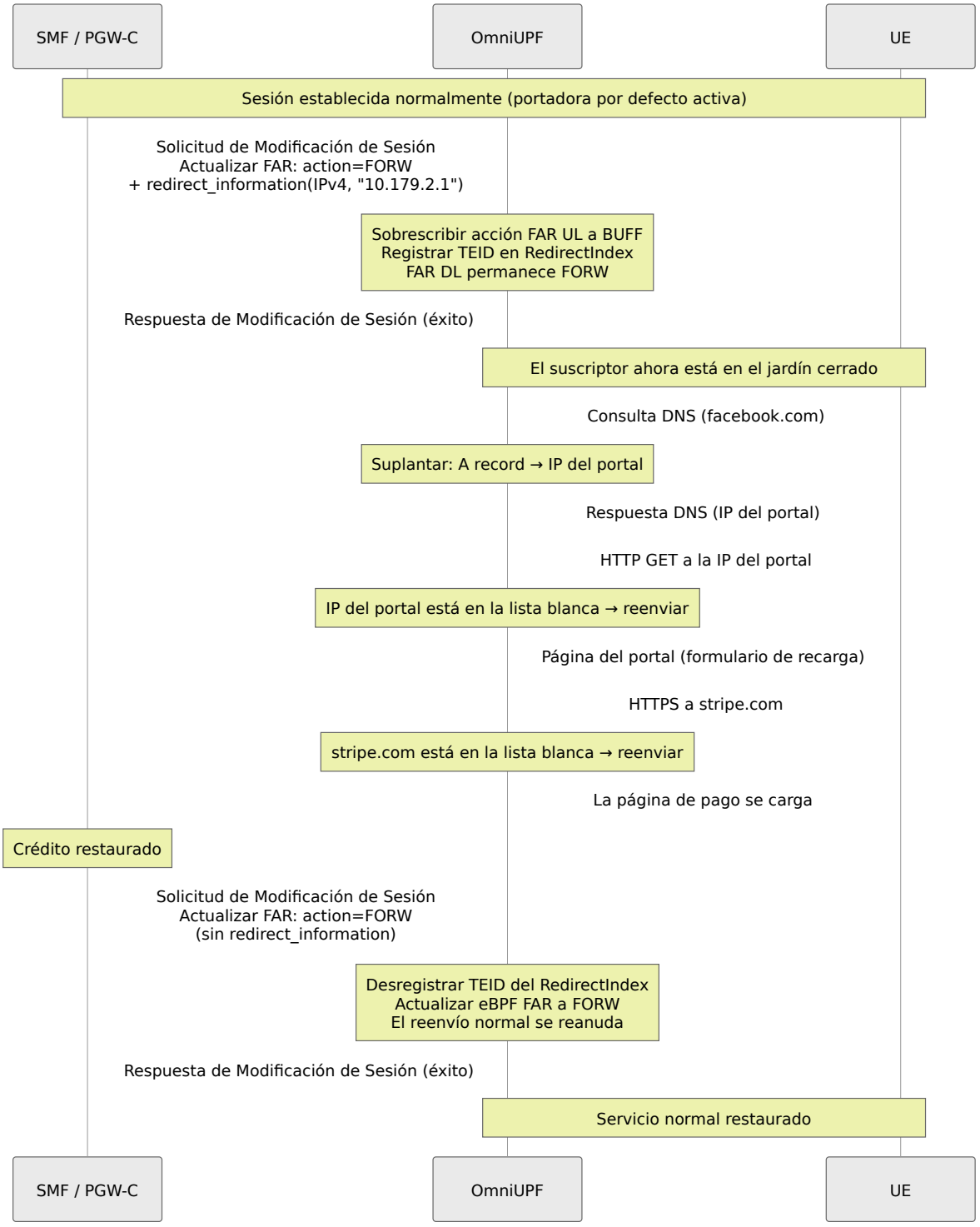
En implementaciones EPC 4G, OmniUPF a menudo actúa como **SGW-U y PGW-U simultáneamente** en el mismo nodo. El SGW-C y el PGW-C establecen cada uno una sesión PFCP separada. La sesión SGW tiene un FAR N9 que reenvía paquetes al TEID de enlace ascendente de la sesión PGW (un loopback a través de la interfaz N3). La sesión PGW realiza la aplicación real de la política del suscriptor.

Al activar una redirección de jardín cerrado a través de `POST /v1/walled_garden`, la lógica de activación detecta automáticamente esta topología:

1. **TEID(s) de sesión PGW** se identifican a partir de los PDRs de enlace ascendente del SEID objetivo.
2. El código escanea todas las sesiones PFCP para encontrar una sesión (la sesión SGW) que tenga un FAR cuyo `teid` coincida con uno de los TEIDs de PDR UL de PGW y tenga `outer_header_creation` configurado. Ese FAR es el FAR de avance N9 que apunta al PGW.
3. **Ambos** los FARs UL de PGW y los FARs UL que miran hacia el gNB de SGW se sobrescriben a `BUFF` en el mapa eBPF. Esto es necesario porque el programa eBPF ve el paquete a medida que llega desde el eNB con el TEID de SGW, no con el TEID de PGW.
4. Solo los **PDRs UL que miran hacia el gNB de SGW** se registran en el RedirectIndex (no los PDRs UL de PGW). El TEID de PDR UL de SGW es lo que eBPF verá en el paquete entrante desde el eNB.
5. Para la **ruta DL** (enviando respuestas de vuelta al UE), el Dispatcher utiliza el **FAR DL de la sesión SGW** (el FAR que reenvía hacia el eNB, con `remoteip != n3_address`). Este FAR contiene el TEID actual de eNB y la IP de gNB. El TEID DL se busca en vivo desde el estado de la sesión en el momento de la respuesta, no se almacena en caché en el momento de la activación — esto maneja el caso donde el TEID de eNB es asignado por una Modificación de Sesión posterior a la activación de la redirección.

En una topología UPF directa (sin SGW, SMF habla directamente con UPF), solo se involucran los propios FARs y TEIDs UL de la sesión — el código de detección de SGW no encuentra nada y los PDRs UL de la sesión principal se registran directamente.

Flujo de Señalización PFCP



IEs PFCP Involucrados

El SMF activa una redirección incluyendo `redirect_information` en los `forwarding_parameters` del FAR:

IE	Descripción
`apply_action`	Establecido en FORW por SMF (UPF lo sobrescribe a BUFF internamente)
`redirect_information`	Contiene el tipo de redirección y la dirección
`forwarding_parameters`	Contiene el `redirect_information` y `outer_header_creation`

Tipos de Información de Redirección (según 3GPP TS 29.244 Tabla 8.2.20-1):

Tipo	Valor	Comportamiento del UPF
IPv4	0	Utiliza la cadena de dirección IPv4 proporcionada como la IP del portal
IPv6	1	Utiliza la cadena de dirección IPv6 proporcionada como la IP del portal
URL	2	Resuelve el nombre de host de la URL a una IP; usa eso como la IP del portal. La ruta de la URL no es utilizada por el UPF -- se almacena en la API solo para visibilidad. El servidor web del portal es responsable de manejar cualquier enrutamiento basado en la ruta.
SIP URI	3	No soportado actualmente

Detección de Portal Captivo

El jardín cerrado activa **detección automática de portal cautivo** en todas las plataformas de dispositivos principales al suplantar respuestas DNS.

Cuando un dispositivo se conecta e intenta verificar la conectividad a internet, consulta dominios bien conocidos. La respuesta DNS suplantada redirige estas verificaciones a la IP del portal, que el dispositivo interpreta como un portal cautivo.

Dominios de Detección de Plataforma

Plataforma	Dominio de Detección	Respuesta
Apple (iOS/macOS)	`captive.apple.com`	HTTP 200 con `<HTML><HEAD>&lt` `</HEAD><BODY>Succes`
Android	`connectivitycheck.gstatic.com`	HTTP 204
Windows	`www.msftconnecttest.com`	HTTP 200 con `Microsoft`
Samsung	`connectivitycheck.samsung.com`	HTTP 200

Cuando estos dominios se resuelven a la IP del portal en lugar de sus direcciones reales, el dispositivo detecta un portal cautivo y presenta la página del portal al usuario ya sea como:

- Una notificación/ventana emergente (iOS, Android)
- Un redireccionamiento automático del navegador (Windows)

El servidor del portal en la IP del portal configurada debe servir respuestas apropiadas para estas URLs de detección y luego redirigir a la página de recarga/pago.

Configuración

La configuración del jardín cerrado vive en el archivo de configuración en tiempo de ejecución. En una instalación de producción (`.deb` package), el archivo de configuración está en `/etc/omniupf/runtime.exs`. El script de inicio

de la versión (`rel/env.sh`) verifica si este archivo está presente y, si es así, establece `RELEASE_CONFIG_DIR=/etc/omniupf` para que la versión de Erlang lo use en lugar del `config/runtime.exs` empaquetado. El UPF debe reiniciarse después de los cambios de configuración.

```
#
=========================================================================
# Jardín Cerrado / Redirección Fuera de Crédito
#
=========================================================================

# Habilitar la aplicación de redirección del jardín cerrado
walled_garden_enabled = true

# IP del servidor del portal (portal cautivo / página de recarga)
walled_garden_portal_ip = "10.179.2.1"

# Resolver DNS ascendente para búsquedas de dominios en la lista blar
walled_garden_dns_resolver = "8.8.8.8"

# Dominios en la lista blanca (los suscriptores pueden acceder a esto
mientras están redirigidos)
# Soporta comodines: "*.stripe.com" coincide con "api.stripe.com"
walled_garden_whitelist = [
  "stripe.com",
  "*.stripe.com",
  "js.stripe.com",
  "hcaptcha.com",
  "*.hcaptcha.com",
  "newassets.hcaptcha.com",
]
```

Parámetros

Parámetro	Tipo	Requerido	Predeterminado
`walled_garden_enabled`	Booleano	No	`false`
`walled_garden_portal_ip`	Cadena (IPv4)	Sí (si está habilitado)	`10.179.2.1`
`walled_garden_dns_resolver`	Cadena (IPv4)	No	`8.8.8.8`

Parámetro	Tipo	Requerido	Predeterminado
`walled_garden_whitelist`	Lista de Cadenas	No	`[]`

Gestión de Lista Blanca

La lista blanca controla qué dominios (y sus IPs resueltas) pueden alcanzar los suscriptores mientras están redirigidos. Esto se configura en el UPF, no en el SMF -- el SMF solo activa la redirección.

Sintaxis de Patrones

Patrón	Coincide	No Coincide
`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` (exacto), `evilstripe.com`
`hcaptcha.com`	`hcaptcha.com`, `newassets.hcaptcha.com`	`evihcaptcha.com`

Los patrones utilizan **anclaje de subdominio**: `stripe.com` coincide con el dominio en sí y cualquier subdominio (`foo.stripe.com`), pero no con dominios que simplemente contienen la cadena (`evilstripe.com`). La coincidencia no distingue entre mayúsculas y minúsculas.

Caché de IP

Cuando el proxy DNS reenvía una consulta para un dominio en la lista blanca, las direcciones IP resueltas se **almacenan automáticamente** en la lista blanca. Esto significa:

1. El suscriptor consulta `api.stripe.com`
2. El proxy DNS reenvía al resolver real, recibe `104.18.7.25`
3. `104.18.7.25` se añade a la caché de IPs en la lista blanca
4. El tráfico HTTP/HTTPS subsiguiente hacia `104.18.7.25` se reenvía (no se descarta)

La IP del portal siempre está en la lista blanca independientemente de la configuración.

Dominios Recomendados para la Lista Blanca

Para un portal de recarga típico con procesamiento de pagos de Stripe y hCaptcha:

```
walled_garden_whitelist = [
  # Procesador de pagos
  "stripe.com",
  "*.stripe.com",

  # Servicio CAPTCHA
  "hcaptcha.com",
  "*.hcaptcha.com",

  # Fuentes de Google (si el portal las utiliza)
  "fonts.googleapis.com",
  "fonts.gstatic.com",

  # CDN para activos del portal (si se aloja externamente)
  "cdn.example.com",
]
```

URLs de Redirección por Sesión

Cada sesión PFCP puede tener un **objetivo de redirección diferente**. El SMF controla esto a través del IE `redirect_information` en el FAR:

- **Tipo IPv4**: La cadena IP proporcionada se analiza y se utiliza como la IP del portal para esa sesión
- **Tipo IPv6**: La cadena IPv6 proporcionada se analiza y se utiliza como la IP del portal para esa sesión
- **Tipo URL**: El nombre de host se extrae y se resuelve a través de DNS en el momento de la creación del FAR. La IP resuelta se utiliza como la IP del portal. La ruta de la URL no es utilizada por el UPF -- se almacena solo para visibilidad de la API.

Esto permite escenarios donde diferentes niveles de suscriptores, MVNOs o planes de servicio redirigen a diferentes portales:

Sesión	redirect_information del SMF	IP del Portal Utilizada	UR
Sesión A	`IPv4: 10.179.2.1`	`10.179.2.1`	`10.179.2`
Sesión B	`IPv6: 2001:db8::1`	`2001:db8::1`	`2001:db8`
Sesión C	`URL: https://topup.mvno.com/recharge`	IP resuelta de `topup.mvno.com`	`https://`

El UPF almacena la IP del portal por sesión y la URL de redirección, que son visibles a través de la API.

API

La ruta base para todos los puntos finales del jardín cerrado es `/v1/walled_garden` (sin prefijo `/api`). Estos puntos finales son servidos por el servidor HTTP Phoenix en el `api_port` configurado (predeterminado: 8080).

GET /v1/walled_garden

Devuelve todas las sesiones de redirección del jardín cerrado activas, IPs en la lista blanca, rangos CIDR en caché y detalles de la sesión.

Respuesta:

```
{
  "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"]
}
```


Campos de Respuesta:

Campo	Descripción
`redirect_count`	Número de sesiones activas en el jardín cerrado
`redirects[].teid`	TEID de enlace ascendente que está siendo interceptado (hex)
`redirects[].session_seid`	SEID de sesión PFCP
`redirects[].portal_ip`	IP del portal para esta sesión específica
`redirects[].redirect_url`	URL de redirección del SMF (si es tipo URL), o null
`redirects[].ue_ip`	Dirección IP del UE
`redirects[].gnb_ip`	IP de gNB para respuestas GTP-U
`redirects[].dl_teid`	TEID de enlace descendente para encapsulación GTP-U
`redirects[].far_global_id`	ID global interno del FAR
`whitelisted_ips`	Todas las IPs actualmente en la lista blanca (portal + resoluciones DNS en caché)
`whitelisted_cidrs`	Rangos CIDR añadidos a través de la API de lista blanca

POST /v1/walled_garden

Activa la redirección del jardín cerrado en una sesión PFCP existente por SEID. Este es el camino activado por el operador/API — el camino normal es a través de la Modificación de Sesión PFCP con `redirect_information` en un FAR. La

ruta de la API es útil para pruebas y para operadores que necesitan redirigir manualmente una sesión sin la participación del SMF.

Cuerpo de la solicitud:

```
{
  "seid": 1,
  "url": "http://10.179.2.1/"
}
```

Campo	Tipo	Requerido	Descripción
`seid`	Entero	Sí	SEID local de la sesión PFCP a redirigir
`url`	Cadena	Sí	Objetivo de redirección — una dirección IPv4/IPv6 o una URL. Si se proporciona una URL, se resuelve el nombre de host y se utiliza la IP resultante como la IP del portal. La cadena completa de la URL se almacena para visibilidad de la API.

Respuesta (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"]
  }
}
```

El campo `sgw_seid` es no nulo cuando se detectó un loopback N9 (sesión emparejada SGW+PGW) y su TEID de enlace ascendente también fue BUFF'd.

`ul_teids` lista todos los TEIDs de enlace ascendente que se registraron en el índice de redirección.

Respuestas de error:

Estado	Significado
404	Sesión no encontrada (SEID no coincide con ninguna sesión PFCP activa)
400	Parámetros requeridos faltantes

DELETE /v1/walled_garden/:seid

Desactiva la redirección del jardín cerrado para una sesión, restaurando el reenvío normal. Todos los FARs de enlace ascendente para la sesión (incluyendo los FARs de sesión emparejada SGW en una topología de loopback N9) se restablecen a `action=FORW` en el mapa eBPF, y los TEIDs se desregistran del índice de redirección.

Parámetro de ruta: `:seid` — el SEID local de la sesión a desactivar.

Respuesta (200 OK):

```
{"status": "redirect removed", "info": {"seid": 1}}
```

Respuestas de error:

Estado	Significado
404	Sesión no encontrada

GET /v1/walled_garden/whitelist

Devuelve la lista blanca actual: IPs individuales en caché (IP del portal y IPs resueltas por DNS) y cualquier rango CIDR añadido a través de la API.

Respuesta:

```
{
  "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

Añade una dirección IP o rango CIDR a la lista blanca en tiempo de ejecución, sin reiniciar el UPF. Los cambios son solo en memoria y no persisten entre reinicios — añade entradas permanentes a `walled_garden_whitelist` en `runtime.exs`.

Cuerpo de la solicitud (añadir una IP):

```
{"ip": "203.0.113.10"}
```

Cuerpo de la solicitud (añadir un rango CIDR):

```
{"cidr": "192.168.100.0/24"}
```

Exactamente uno de `ip` o `cidr` debe estar presente. Añadir un rango CIDR significa que cualquier tráfico a IPs dentro de ese rango es reenviado por el dispatcher sin necesidad de una entrada de caché DNS por IP.

Respuesta (200 OK — IP añadida):

```
{"status": "added", "ip": "203.0.113.10"}
```

Respuesta (200 OK — CIDR añadido):

```
{"status": "added", "cidr": "192.168.100.0/24"}
```

Respuestas de error:

Estado	Significado
400	IP inválida, CIDR inválido o campos del cuerpo faltantes

Métricas de Prometheus

Medidores

Métrica: `upf_walled_garden_active_redirects` **Tipo**: Medidor **Descripción**: Número de sesiones actualmente en estado de redirección del jardín cerrado

Consultas de ejemplo:

```
# Conteo actual de redirecciones
upf_walled_garden_active_redirects
```

Contadores

Métrica: `upf_walled_garden_packets_intercepted_total` **Tipo**: Contador **Descripción**: Total de paquetes interceptados por el jardín cerrado (todo el tráfico de enlace ascendente de sesiones redirigidas)

Métrica: `upf_walled_garden_packets_dropped_total` **Tipo**: Contador **Descripción**: Total de paquetes descartados por el jardín cerrado (tráfico no en

la lista blanca, no DNS)

Métrica: `upf_walled_garden_packets_forwarded_total` **Tipo**: Contador **Etiquetas**:

- `dst_ip` - IP de destino del paquete reenviado **Descripción**: Paquetes reenviados a través del jardín cerrado a destinos en la lista blanca. Etiquetado por IP de destino para visibilidad por destino.

Métrica: `upf_walled_garden_bytes_forwarded_total` **Tipo**: Contador **Etiquetas**:

- `dst_ip` - IP de destino del tráfico reenviado **Descripción**: Bytes reenviados a través del jardín cerrado por IP de destino. Utiliza esto para identificar qué servicios en la lista blanca acceden los suscriptores mientras están redirigidos.

Métrica: `upf_walled_garden_dns_spoofed_total` **Tipo**: Contador **Etiquetas**:

- `domain` - El dominio que fue suplantado **Descripción**: Consultas DNS suplantadas por el jardín cerrado. Etiquetado por dominio consultado.

Métrica: `upf_walled_garden_dns_forwarded_total` **Tipo**: Contador **Etiquetas**:

- `domain` - El dominio en la lista blanca que fue reenviado **Descripción**: Consultas DNS reenviadas al resolver real (dominios en la lista blanca). Etiquetado por dominio.

Consultas de Ejemplo

```
# Sesiones activas del jardín cerrado
upf_walled_garden_active_redirects

# Tasa de interceptación (paquetes/segundo)
rate(upf_walled_garden_packets_intercepted_total[5m])

# Tasa de descartes (debería ser la mayoría de los interceptados)
rate(upf_walled_garden_packets_dropped_total[5m])

# Tráfico reenviado por destino (bytes/segundo)
sum by (dst_ip)
(rate(upf_walled_garden_bytes_forwarded_total[5m]))

# Top 5 destinos en la lista blanca por volumen de tráfico
topk(5, sum by (dst_ip)
(rate(upf_walled_garden_bytes_forwarded_total[5m])))

# Dominios suplantados más consultados
topk(10, sum by (domain)
(rate(upf_walled_garden_dns_spoofed_total[5m])))

# Tasa de búsqueda de dominios en la lista blanca
sum by (domain) (rate(upf_walled_garden_dns_forwarded_total[5m]))

# Proporción de descartes vs reenviados
sum(rate(upf_walled_garden_packets_dropped_total[5m]))
  / sum(rate(upf_walled_garden_packets_intercepted_total[5m]))
```

Solución de Problemas

El Portal Cautivo No Aparece en el Dispositivo

Síntomas: El suscriptor es redirigido (visible en la API) pero el dispositivo no muestra el aviso del portal cautivo.

Causas posibles:

- El servidor del portal no responde en la IP del portal configurada
- El servidor del portal no maneja las URLs de detección específicas de la plataforma
- La respuesta DNS no llega al UE (verificar la ruta GTP-U)

Resolución:

1. Verifica que el servidor del portal sea accesible: `curl http://<portal_ip>/`
2. Confirma que el portal maneje las URLs de detección (por ejemplo, `GET /hotspot-detect.html` para Apple)
3. Verifica `GET /v1/walled_garden` para confirmar que la sesión esté registrada
4. Verifica las métricas de Prometheus: `upf_walled_garden_dns_spoofed_total` debería estar incrementando
5. Verifica que la ruta GTP-U de enlace descendente esté funcionando (el FAR DL debería permanecer FORW)

La Página de Pago No Carga

Síntomas: El suscriptor ve el portal cautivo pero no puede acceder a la página de pago (Stripe, etc.).

Causas posibles:

- Dominio del procesador de pagos no está en la lista blanca
- Procesador de pagos utilizando una IP de CDN que no fue almacenada en caché
- Patrón de la lista blanca no coincide correctamente con subdominios

Resolución:

1. Verifica `GET /v1/walled_garden` — verifica que `whitelisted_ips` incluya las IPs del procesador de pagos
2. Verifica Prometheus: `upf_walled_garden_dns_forwarded_total{domain="stripe.com"}` debería mostrar búsquedas

3. Añade dominios faltantes a la lista blanca (común: `js.stripe.com`, `m.stripe.network`)
4. Verifica `upf_walled_garden_bytes_forwarded_total` por `dst_ip` para ver qué tráfico está fluyendo realmente

La Redirección No Se Activa

Síntomas: El SMF envía la Modificación de Sesión con `redirect_information` pero el tráfico del suscriptor no es interceptado.

Causas posibles:

- `walled_garden_enabled` está en `false`
- El IE `redirect_information` está en el FAR incorrecto (debe estar en el FAR **de enlace ascendente**)
- La acción del FAR no se está sobrescribiendo a BUFF

Resolución:

1. Verifica la configuración: `walled_garden_enabled = true`
2. Verifica `GET /v1/pfcp_sessions` — observa la acción del FAR para la sesión. El FAR de enlace ascendente debería mostrar acción `0x04` (BUFF)
3. Verifica `GET /v1/walled_garden` — el TEID de la sesión debería aparecer en la lista de redirección
4. Verifica los registros del UPF para mensajes de `redirect_info` durante la modificación de la sesión

La Redirección No Se Limpia Después de la Recarga

Síntomas: El suscriptor ha recargado pero el tráfico sigue siendo interceptado.

Causas posibles:

- El SMF no ha enviado la Modificación de Sesión para eliminar la redirección
- La transición de BUFF->FORW no se está detectando

Resolución:

1. Verifica `GET /v1/walled_garden` — ¿la sesión aún está listada?
2. Verifica `GET /v1/pfcp_sessions` — verifica que la acción del FAR se haya actualizado de nuevo a `0x02` (FORW)
3. Verifica los registros del SMF para confirmar que envió la Modificación de Sesión
4. Verifica los registros del UPF para el mensaje "redirect removed, unregistering walled garden"

TEID de Enlace Descendente Incorrecto (Obsoleto Después de la Modificación de Sesión)

Síntomas: Respuestas DNS suplantadas o tráfico reenviado no llegan al UE; los paquetes GTP-U se envían a la IP correcta del gNB pero el eNB los descarta como TEID desconocido.

Causa: En topologías 4G/SGW+PGW, el TEID de eNB en el FAR DL de SGW puede ser cero o un TEID de marcador de posición en el momento en que se activa la redirección del jardín cerrado (por ejemplo, el intercambio de `Initial Context Setup` ocurre después del Establecimiento de Sesión). Cuando el eNB asigna su TEID y lo envía de vuelta a través de la Modificación de Sesión, el FAR DL en la sesión SGW se actualiza — pero si el Dispatcher había almacenado en caché el valor antiguo, usaría el TEID incorrecto.

Resolución: El Dispatcher resuelve el TEID DL **en vivo** en el momento de la respuesta buscando el FAR DL actual de la sesión SGW (el FAR con `outer_header_creation` configurado y `remoteip != n3_address`). Solo recurre al valor almacenado en el momento de la activación si la búsqueda de la sesión falla. Por lo tanto, un TEID obsoleto debería corregirse por sí mismo tan pronto como se envíe la siguiente respuesta DNS o reenviada. Si aún ves el TEID incorrecto:

1. Verifica `GET /v1/walled_garden` — verifica que `dl_teid` en la entrada de redirección parezca plausible (no cero).
2. Verifica `GET /v1/pfcp_sessions` — observa las entradas del FAR de la sesión SGW; el FAR que mira hacia el gNB debería tener el `teid` y `remoteip` actuales.

3. Si la sesión SGW ha sido eliminada (por ejemplo, por traspaso o liberación), la entrada de redirección permanecerá en el RedirectIndex pero la búsqueda en vivo fallará. En este caso, utiliza `DELETE /v1/walled_garden/:seid` para limpiar la entrada obsoleta y permitir que el SMF restablezca.

El Portal Devuelve 304 No Modificado

Síntomas: El navegador del suscriptor muestra una página en blanco o la página del portal cautivo se carga una vez pero las visitas posteriores parecen vacías.

Causa: Las respuestas HTTP 304 (No Modificado) se envían cuando el navegador tiene una versión en caché de la página y el servidor confirma que nada ha cambiado. Para los flujos de portal cautivo, algunos servidores web de portal envían 304 en respuesta a las solicitudes de detección de plataforma (`/hotspot-detect.html`, `/generate_204`, etc.) si el navegador envía encabezados `If-Modified-Since` o `If-None-Match`. Algunas implementaciones de portal cautivo también envían 304 para la propia página de redirección.

Resolución:

1. El UPF reenvía respuestas HTTP del servidor del portal de forma transparente — no modifica los códigos de respuesta. El problema está en la configuración del servidor web del portal.
2. En el servidor del portal, asegúrate de que los puntos finales de detección de plataforma devuelvan el código de estado correcto (200, no 304) con el contenido del cuerpo esperado (ver Detección de Portal Captivo).
3. Configura el servidor del portal para enviar `Cache-Control: no-store` y omitir los encabezados `ETag`/`Last-Modified` en los puntos finales de detección para que los navegadores no los almacenen en caché.
4. Verifica con: `curl -v -H "If-None-Match: foo" http://<portal_ip>/hotspot-detect.html` — la respuesta debería ser 200, no 304.

Reenvío de Socket Crudo

Cuando el Dispatcher reenvía un paquete a una IP en la lista blanca, utiliza un **socket crudo** (`IPPROTO_RAW`, número de protocolo 255) a través del módulo `:socket` de Erlang. Se abre un nuevo socket crudo por paquete, se envía el paquete IP interno completo (tal como se recibió de eBPF) con `sendto`, y el socket se cierra inmediatamente.

Cómo funciona esto: El paquete IP interno del payload GTP-U ya tiene un encabezado IP válido con la IP del UE como origen y la IP del servidor de destino como destino. Al inyectar este paquete a través de un socket crudo utilizando `IPPROTO_RAW`, el kernel lo enruta en función de la IP de destino utilizando la tabla de enrutamiento del host. La interfaz N6 del UPF debe tener una ruta hacia el servidor del portal/en la lista blanca para que esto funcione.

Problemas comunes:

Síntoma	Causa Probable	Solución
Reenvío falla silenciosamente	`EPERM` — el socket crudo requiere root o `CAP_NET_RAW`	Asegúrate de que OmniUPF se ejecute como root o tenga la capacidad `CAP_NET_RAW`
Paquetes reenviados pero ninguna respuesta llega al UE	Ruta N6 hacia el portal faltante	Añade ruta a la subred del portal en el host UPF
`Walled garden: raw socket open failed` en los registros	Falta de capacidad o restricción del kernel	Verifica el servicio `systemd` `AmbientCapabilities=CAP_NET_RAW`
El reenvío funciona pero el UE obtiene una IP incorrecta como origen	NAT en N6 reescribe el origen	Asegúrate de que las respuestas del portal a la IP del UE sean enrutadas de vuelta a través del UPF

Verifica los registros del UPF para mensajes de `Walled garden forward failed` y `raw socket open failed`. Utiliza las métricas de Prometheus `upf_walled_garden_packets_forwarded_total` y `upf_walled_garden_bytes_forwarded_total` para confirmar que el tráfico esté fluyendo.

Advertencia de Alta Cardinalidad para Prometheus

Nota: Las etiquetas `dst_ip` y `domain` en las métricas del jardín cerrado pueden producir alta cardinalidad si se consultan muchos destinos o dominios únicos. En implementaciones grandes, considera usar reglas de grabación para agregar estas métricas:

```
# Regla de grabación para agregar por subred /24 en lugar de IPs
individuales
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+"
  )
)
```

Guía de Operaciones de la Interfaz Web

Tabla de Contenidos

1. Descripción General
2. Acceso al Panel de Control
3. Vista de Sesiones
4. Gestión de Reglas
5. Gestión de Buffers
6. Panel de Estadísticas
7. Monitoreo de Capacidad
8. Vista de Configuración
9. Vista de Rutas
10. Vista de Capacidades XDP
11. Visor de Registros

Descripción General

La Interfaz Web de OmniUPF proporciona un panel de control integral para el monitoreo y gestión en tiempo real de la Función de Plano de Usuario. La interfaz está construida sobre Phoenix LiveView y proporciona:

- **Visibilidad en tiempo real** de sesiones PFCP y conexiones PDU activas
- **Inspección de reglas** para PDR, FAR, QER y URR en todas las sesiones
- **Gestión de buffers** para el almacenamiento de paquetes durante eventos de movilidad
- **Monitoreo de estadísticas** para el procesamiento de paquetes, rutas e interfaces
- **Seguimiento de capacidad** para el uso y límites del mapa eBPF
- **Visualización de registros en vivo** para la solución de problemas

Arquitectura

El panel de control se comunica con múltiples instancias de OmniUPF a través de su API REST para:

- Consultar sesiones y asociaciones PFCP
- Inspeccionar reglas de detección y reenvío de paquetes
- Monitorear buffers de paquetes y su estado
- Acceder a estadísticas y métricas de rendimiento en tiempo real
- Rastrear la capacidad y utilización del mapa eBPF

Acceso al Panel de Control

Acceso por Defecto

El panel de control es accesible a través de HTTPS en el servidor de gestión de OmniUPF:

```
https://<upf-server>:443/
```

Puerto por Defecto: 443 (HTTPS con certificado autofirmado)

Configuración

El panel de control requiere la configuración del host de OmniUPF en `config/config.exs`:

Se pueden configurar múltiples instancias de UPF para implementaciones de múltiples instancias:

La configuración `upf_hosts` define qué instancias de OmniUPF están disponibles en el menú desplegable del selector de host en toda la interfaz.

Navegación

El panel de control proporciona pestañas de navegación para cada área operativa:

- **Sesiones** - `/sessions` - Sesiones y asociaciones PFCP
- **Reglas** - `/rules` - Inspección de reglas PDR, FAR, QER, URR
- **Buffers** - `/buffers` - Monitoreo y control de buffers de paquetes
- **Estadísticas** - `/statistics` - Estadísticas de paquetes, rutas, XDP e interfaces
- **Capacidad** - `/capacity` - Uso y monitoreo de capacidad del mapa eBPF
- **Config** - `/upf_config` - Configuración de UPF y direcciones del plano de datos
- **Rutas** - `/routes` - Rutas de UE y sesiones de protocolo de enrutamiento (OSPF, BGP)
- **Capacidades XDP** - `/xdp_capabilities` - Soporte de modo XDP y capacidades de rendimiento
- **Registros** - `/logs` - Transmisión de registros en vivo

Vista de Sesiones

URL: `/sessions`

Características

La vista de Sesiones muestra todas las sesiones PFCP activas y asociaciones de las instancias de OmniUPF seleccionadas.

Resumen de Asociaciones PFCP

Muestra todas las asociaciones PFCP activas (conexiones de control desde SMF/PGW-C):

Columna	Descripción
ID de Nodo	Identificador de nodo SMF o PGW-C (FQDN o IP)
Dirección	Dirección IP de SMF/PGW-C para comunicación PFCP
Siguiente ID de Sesión	Siguiente ID de sesión PFCP disponible para esta asociación

Propósito:

- Verificar la conectividad de SMF a UPF
- Monitorear el número de conexiones de plano de control
- Rastrear la asignación de ID de sesión por asociación

Tabla de Sesiones Activas

Muestra todas las sesiones PFCP que representan sesiones PDU activas de UE:

Columna	Descripción
SEID Local	Identificador de punto final de sesión asignado por UPF
SEID Remoto	Identificador de punto final de sesión asignado por SMF
IP de UE	Dirección IP IPv4 o IPv6 del equipo de usuario
TEID	Identificador de punto final de túnel GTP-U para tráfico ascendente
PDRs	Número de reglas de detección de paquetes en la sesión
FARs	Número de reglas de acción de reenvío en la sesión
QERs	Número de reglas de aplicación de QoS en la sesión
URRs	Número de reglas de informes de uso en la sesión
Acciones	Botón de expandir para ver información detallada de la regla

Características:

- **Filtrar por IP**: Encontrar sesiones para una dirección IP específica de UE
- **Filtrar por TEID**: Encontrar sesiones por ID de punto final de túnel
- **Expandir sesión**: Ver detalles completos de PDR/FAR/QER/URR en JSON
- **Actualización automática**: Se actualiza cada 10 segundos

Vista de Sesión Expandida:

Cuando haces clic en "Expandir" en una sesión, la vista muestra:

- **Reglas de Detección de Paquetes (PDRs)**: JSON completo con TEID, IP de UE, ID de FAR, ID de QER, filtros SDF

    - **Los IDs de PDR son clicables** - Haz clic para navegar a la pestaña Reglas y ver los detalles completos de PDR
    - PDRs ascendentes (TEID ≠ 0) enlazan a la búsqueda de PDR ascendente
    - PDRs descendentes (IPv4) enlazan a la búsqueda de PDR descendente
    - PDRs descendentes (IPv6) enlazan a la búsqueda de PDR descendente IPv6
- **Reglas de Acción de Reenvío (FARs)**: Banderas de acción, creación de encabezado externo, puntos finales de destino
- **Reglas de Aplicación de QoS (QERs)**: MBR, GBR, QFI y otros parámetros de QoS
- **Reglas de Informes de Uso (URRs)**: Contadores de volumen (ascendente, descendente, bytes totales)

Vista expandida de sesión mostrando PDRs, FARs y QERs detallados para una sesión específica.

Casos de Uso

Verificar Conectividad de UE:

1. Navegar a la vista de Sesiones
2. Ingresar la dirección IP de UE en el filtro
3. Confirmar que la sesión existe con el TEID correcto
4. Expandir para verificar la configuración de PDR/FAR

Monitorear Conteo de Sesiones:

- Verificar el conteo total de sesiones en el encabezado
- Comparar entre múltiples instancias de UPF
- Rastrear el crecimiento de sesiones a lo largo del tiempo

Solucionar Problemas de Sesiones:

- Buscar una IP de UE o TEID específico
- Expandir la sesión para inspeccionar la configuración de reglas
- Verificar los parámetros de reenvío de FAR
- Comprobar la configuración de QoS de QER

Actualizaciones en Tiempo Real

La vista de Sesiones se actualiza automáticamente cada 10 segundos. Un indicador de verificación de salud muestra el estado de conectividad de UPF:

- **SALUDABLE** (verde): UPF es alcanzable y está respondiendo
- **NO SALUDABLE** (rojo): UPF no es alcanzable o no está respondiendo
- **DESCONOCIDO** (gris): Estado de salud aún no determinado

Gestión de Reglas

URL: `/rules`

La vista de Reglas proporciona una inspección integral de todas las reglas de detección de paquetes, reenvío, QoS e informes de uso en todas las sesiones.

Pestaña PDR - Reglas de Detección de Paquetes

Ver e inspeccionar todos los PDR en el UPF con **formularios de búsqueda** y **navegación clicable**:

PDRs Ascendentes (N3 → N6):

- **Formulario de Búsqueda**: Buscar por TEID para ver detalles específicos de PDR ascendente
- **TEID**: ID de punto final de túnel GTP-U desde gNB (clicable - navega a búsqueda)
- **ID de FAR**: Regla de acción de reenvío asociada (clicable - navega a la pestaña FAR)
- **ID de QER**: Regla de aplicación de QoS asociada (clicable - navega a la pestaña QER)
- **IDs de URR**: Reglas de informes de uso asociadas (clicable - navega a la pestaña URR)
- **Eliminación de Encabezado Externo**: Bandera de desencapsulación GTP-U
- **Filtros SDF**: Reglas de clasificación de flujo de datos de servicio

PDRs Descendentes (N6 → N3):

- **Formulario de Búsqueda**: Buscar por dirección IPv4 de UE para ver detalles específicos de PDR descendente
- **IP de UE**: Dirección IPv4 del equipo de usuario (mostrada en resultados de búsqueda)
- **ID de FAR**: Regla de acción de reenvío asociada (clicable - navega a la pestaña FAR)
- **ID de QER**: Regla de aplicación de QoS asociada (clicable - navega a la pestaña QER)
- **IDs de URR**: Reglas de informes de uso asociadas (clicable - navega a la pestaña URR)
- **Modo SDF**: Modo de filtro de flujo de datos de servicio (ninguno, solo sdf, sdf + predeterminado)

- **Paginación**: Navegar por PDRs con controles de página (predeterminado 100 por página, máximo 1000)

PDRs Descendentes IPv6:

- La API admite paginación para PDRs descendentes IPv6
- La misma estructura que IPv4 pero indexada por direcciones IPv6
- Se puede agregar una pestaña completa de UI si es necesario

Pestaña FAR - Reglas de Acción de Reenvío

Ver todas las FARs con sus acciones de reenvío y parámetros:

Características:

- **Formulario de Búsqueda**: Buscar por ID de FAR para ver detalles específicos de FAR
- **Búsqueda Automática**: Hacer clic en IDs de FAR desde detalles de PDR automáticamente llena la búsqueda
- **Actualizaciones en Tiempo Real**: El estado de FAR refleja el estado actual de almacenamiento en buffer

Columna	Descripción
ID de FAR	Identificador único de regla de reenvío
Acción	Banderas de acción de reenvío (REENVIAR, DESCARTAR, BUFFER, DUPLICAR, NOTIFICAR)
Almacenamiento en Buffer	Estado actual de almacenamiento en buffer (Habilitado/Deshabilitado)
Destino	Parámetros de creación de encabezado externo (TEID, dirección IP)

Banderas de Acción de FAR:

- **REENVIAR (1)**: Reenviar paquete al destino

- **DESCARTAR (2)**: Descartar paquete
- **BUFFER (4)**: Almacenar paquete en buffer
- **NOTIFICAR (8)**: Enviar notificación al plano de control
- **DUPLICAR (16)**: Duplicar paquete a múltiples destinos

Alternar Almacenamiento en Buffer:

- Hacer clic en "Habilitar Buffer" o "Deshabilitar Buffer" para alternar la bandera de almacenamiento en buffer
- Útil para solucionar problemas en escenarios de traspaso
- Cambia la acción de FAR inmediatamente en el mapa eBPF

Pestaña QER - Reglas de Aplicación de QoS

Ver reglas de QoS aplicadas a flujos de tráfico:

Características:

- **Navegación Clicable**: Hacer clic en IDs de QER desde detalles de PDR para navegar y resaltar QER específico
- **Auto-resaltado**: La fila de QER se resalta cuando se navega desde PDR
- **Paginación**: Navegar por QERs con controles de página (predeterminado 100 por página, máximo 1000)

Columna	Descripción
ID de QER	Identificador único de regla de QoS (clicable cuando se referencia desde PDRs)
MBR (Ascendente)	Tasa máxima de bits para tráfico ascendente (kbps)
MBR (Descendente)	Tasa máxima de bits para tráfico descendente (kbps)
GBR (Ascendente)	Tasa garantizada de bits para tráfico ascendente (kbps)
GBR (Descendente)	Tasa garantizada de bits para tráfico descendente (kbps)
QFI	Identificador de Flujo QoS (marcado 5G)

Interpretación de QoS:

- **MBR = 0**: Sin límite de tasa
- **GBR = 0**: Mejor esfuerzo (sin ancho de banda garantizado)
- **GBR > 0**: Flujo de tasa garantizada (priorizado)

Pestaña URR - Reglas de Informes de Uso

Ver reglas de seguimiento de uso y contadores de volumen:

Características:

- **Formulario de Búsqueda**: Buscar por ID de URR para encontrar y resaltar URR específico
- **Navegación Clicable**: Hacer clic en IDs de URR desde detalles de PDR para navegar y resaltar URR específico
- **Auto-resaltado**: La fila de URR se resalta en azul cuando se navega desde PDR o se busca a través de búsqueda

- **Paginación**: Navegar por URRs con controles de página (predeterminado 100 por página, máximo 1000)

Columna	Descripción
ID de URR	Identificador único de regla de informes de uso (clicable cuando se referencia desde PDRs)
Volumen Ascendente	Bytes enviados desde UE a la red de datos
Volumen Descendente	Bytes enviados desde la red de datos a UE
Volumen Total	Total de bytes en ambas direcciones
Acciones	Botón de eliminar para restablecer contadores para este URR

Visualización de Volumen:

- Formateado automáticamente (B, KB, MB, GB, TB)
- Contadores en tiempo real actualizados en cada actualización
- Usado para facturación y análisis

Filtrado:

- Solo muestra URRs con volumen distinto de cero
- URRs inactivas (todos los contadores en 0) se filtran para mejorar el rendimiento

Casos de Uso

Inspeccionar Clasificación de Tráfico:

1. Navegar a Reglas → pestaña PDR
2. Buscar por TEID o IP de UE específica

3. Verificar que PDR se asocie con el FAR y QER correctos

Solucionar Problemas de Reenvío:

1. Navegar a Reglas → pestaña FAR
2. Localizar ID de FAR desde PDR de sesión
3. Verificar que la acción sea REENVIAR (no DESCARTAR o BUFFER)
4. Comprobar los parámetros de creación de encabezado externo

Monitorear Aplicación de QoS:

1. Navegar a Reglas → pestaña QER
2. Verificar que los valores de MBR y GBR coincidan con la política
3. Comprobar el marcado QFI para flujos 5G

Rastrear Uso de Datos:

1. Navegar a Reglas → pestaña URR
2. Ordenar por volumen total para encontrar los usuarios más altos
3. Monitorear el crecimiento del volumen a lo largo del tiempo
4. Verificar la integración de facturación

Gestión de Buffers

URL: `/buffers`

Características

La vista de Buffers muestra los buffers de paquetes mantenidos por el UPF durante eventos de movilidad o cambios de ruta.

Estadísticas Totales

El panel muestra estadísticas agregadas de buffers:

- **Total de Paquetes**: Número de paquetes almacenados en todos los FARs
- **Total de Bytes**: Tamaño total de datos almacenados

- **Total de FARs**: Número de FARs con paquetes almacenados
- **Máx. Por FAR**: Máximo de paquetes permitidos por FAR
- **Máx. Total**: Máximo total de paquetes almacenados
- **TTL de Paquete**: Tiempo de vida para paquetes almacenados (segundos)

Buffers por FAR

Tabla de todos los FARs con paquetes almacenados:

Columna	Descripción
ID de FAR	Identificador de regla de acción de reenvío
Conteo de Paquetes	Número de paquetes almacenados para este FAR
Conteo de Bytes	Total de bytes almacenados para este FAR
Paquete Más Antiguo	Marca de tiempo del paquete almacenado más antiguo
Paquete Más Nuevo	Marca de tiempo del paquete almacenado más nuevo
Acciones	Botones de control de buffer (estilo píldora)

Acciones de Control de Buffer

Para cada FAR con paquetes almacenados, están disponibles los siguientes botones de estilo píldora:

Control de Almacenamiento en Buffer:

- **Deshabilitar Buffer** (rojo): Apagar el almacenamiento en buffer para este FAR (actualiza la bandera de acción de FAR)
- **Habilitar Buffer** (púrpura): Activar el almacenamiento en buffer para este FAR

Operaciones de Buffer:

- **Vaciar** (azul): Reproducir todos los paquetes almacenados utilizando las reglas actuales de FAR
- **Limpiar** (gris): Eliminar todos los paquetes almacenados sin reenviar

Limpiar Todos los Buffers:

- Botón rojo "Limpiar Todo" en el encabezado
- Limpia los buffers para todos los FARs
- Requiere confirmación

Casos de Uso

Monitorear Almacenamiento en Buffer durante el Traspaso:

1. Durante el traspaso, verificar que los paquetes se estén almacenando
2. Comprobar el estado de almacenamiento en buffer de FAR (debería estar habilitado)
3. Monitorear el conteo y la antigüedad de paquetes

Completar el Traspaso:

1. Después del cambio de ruta, hacer clic en "Vaciar" para reproducir los paquetes almacenados
2. Verificar que los paquetes se reenvíen a la nueva ruta
3. Hacer clic en "Deshabilitar Buffer" para detener el almacenamiento en buffer

Limpiar Buffers Atascados:

1. Identificar FARs con paquetes almacenados antiguos (comprobar la marca de tiempo más antigua)
2. Hacer clic en "Limpiar" para descartar paquetes obsoletos
3. O hacer clic en "Deshabilitar Buffer" para evitar más almacenamiento en buffer

Solucionar Problemas de Desbordamiento de Buffer:

1. Comprobar el conteo total de paquetes vs. máximo total

2. Identificar FARs con almacenamiento en buffer excesivo
3. Verificar que SMF haya enviado modificación de sesión para deshabilitar el almacenamiento en buffer
4. Deshabilitar manualmente el almacenamiento en buffer si se perdió el comando de SMF

Actualizaciones en Tiempo Real

La vista de Buffers se actualiza automáticamente cada 5 segundos para mostrar el estado actual del buffer.

Panel de Estadísticas

URL: `/statistics`

Características

La vista de Estadísticas proporciona métricas de rendimiento en tiempo real del plano de datos de OmniUPF. Para información detallada sobre métricas de Prometheus, consulte la **Referencia de Métricas**.

Estadísticas de Paquetes

Contadores agregados de procesamiento de paquetes:

- **Paquetes RX**: Total de paquetes recibidos en todas las interfaces
- **Paquetes TX**: Total de paquetes transmitidos en todas las interfaces
- **Paquetes Descartados**: Paquetes descartados debido a errores o políticas
- **Paquetes GTP-U**: Paquetes procesados con encapsulación GTP-U

Uso: Monitorear la carga de tráfico general de UPF y la tasa de paquetes descartados

Estadísticas de Rutas

Métricas de reenvío por ruta (si están disponibles):

- **Acercamientos de Ruta**: Paquetes coincidentes con cada regla de enrutamiento
- **Éxito de Reenvío**: Conteo de paquetes reenviados con éxito
- **Errores de Reenvío**: Intentos de reenvío fallidos

Uso: Identificar rutas ocupadas y errores de reenvío

Estadísticas XDP

Métricas de rendimiento de eXpress Data Path:

- **XDP Procesados**: Total de paquetes procesados en la capa XDP
- **XDP Pasados**: Paquetes enviados a la pila de red
- **XDP Descartados**: Paquetes descartados en la capa XDP
- **XDP Abortados**: Errores de procesamiento en el programa XDP

Uso: Monitorear el rendimiento de XDP y detectar errores de procesamiento

Causas de Descartes de XDP:

- Formato de paquete inválido
- Fallo en la búsqueda de mapa eBPF
- Descartes basados en políticas
- Agotamiento de recursos

Estadísticas de Interfaces N3/N6

Contadores de tráfico por interfaz:

Interfaz N3 (conectividad RAN):

- **RX N3**: Paquetes recibidos desde gNB/eNodeB
- **TX N3**: Paquetes transmitidos a gNB/eNodeB

Interfaz N6 (conectividad de red de datos):

- **RX N6**: Paquetes recibidos desde la red de datos (Internet/IMS)
- **TX N6**: Paquetes transmitidos a la red de datos

Total: Conteo agregado de paquetes a través de las interfaces

Uso: Monitorear el equilibrio de tráfico y problemas específicos de la interfaz

Casos de Uso

Monitorear Carga de Tráfico:

1. Comprobar tasas de paquetes RX/TX
2. Verificar que el tráfico fluya en ambas direcciones
3. Comparar tráfico N3 vs N6 (debería ser aproximadamente igual)

Detectar Paquetes Descartados:

1. Comprobar contador de paquetes descartados
2. Revisar contador de paquetes descartados de XDP
3. Investigar la causa en los registros si los descartes son altos

Análisis de Rendimiento:

1. Monitorear la relación entre XDP procesados y pasados
2. Comprobar abortos de XDP (indica errores)
3. Verificar la distribución del tráfico de interfaces N3/N6

Planificación de Capacidad:

1. Rastrear la tasa de paquetes a lo largo del tiempo
2. Comparar con los límites de capacidad de UPF
3. Planificar escalado si se acercan a los límites

Actualizaciones en Tiempo Real

Las estadísticas se actualizan automáticamente cada 5 segundos.

Monitoreo de Capacidad

URL: `/capacity`

Características

La vista de Capacidad muestra el uso del mapa eBPF y los límites de capacidad para todos los mapas en el plano de datos de UPF.

Tabla de Uso del Mapa eBPF

Tabla de todos los mapas eBPF con información de uso:

Columna	Descripción
Nombre del Mapa	Nombre del mapa eBPF (por ejemplo, `uplink_pdr_map`, `far_map`)
Usado	Número de entradas actualmente en el mapa
Capacidad	Máximo de entradas permitidas en el mapa
Uso	Barra de progreso visual con porcentaje
Tamaño de Clave	Tamaño de las claves del mapa en bytes
Tamaño de Valor	Tamaño de los valores del mapa en bytes

Indicadores de Uso Codificados por Color

La barra de progreso de uso está codificada por color según la utilización:

- **Verde (<50%)**: Operación normal, capacidad amplia
- **Amarillo (50-70%)**: Precaución, monitorear crecimiento
- **Ámbar (70-90%)**: Advertencia, planificar aumento de capacidad
- **Rojo (>90%)**: Crítico, se requiere acción inmediata

Mapas Críticos a Monitorear

uplink_pdr_map:

- Almacena PDRs ascendentes indexados por TEID
- Una entrada por flujo de tráfico ascendente
- **Crítico**: El agotamiento impide el establecimiento de nuevas sesiones

downlink_pdr_map / **downlink_pdr_map_ip6**:

- Almacena PDRs descendentes indexados por dirección IP de UE
- Una entrada por dirección IPv4/IPv6 de UE
- **Crítico**: El agotamiento impide el establecimiento de nuevas sesiones

far_map:

- Almacena reglas de acción de reenvío indexadas por ID de FAR
- Compartido entre múltiples PDRs
- **Alta Prioridad**: Afecta decisiones de reenvío

qer_map:

- Almacena reglas de aplicación de QoS indexadas por ID de QER
- **Prioridad Media**: Afecta QoS pero no conectividad básica

urr_map:

- Almacena reglas de informes de uso indexadas por ID de URR
- **Baja Prioridad**: Afecta facturación pero no conectividad

Casos de Uso

Planificación de Capacidad:

1. Monitorear tendencias de uso de mapas a lo largo del tiempo
2. Identificar qué mapas están creciendo más rápido
3. Planificar aumentos de capacidad antes de alcanzar límites

Prevenir Fallos en el Establecimiento de Sesiones:

1. Comprobar el uso del mapa PDR antes de un aumento de tráfico esperado
2. Aumentar la capacidad del mapa si se acercan a los límites
3. Monitorear después del aumento de capacidad para verificar

Solucionar Fallos de Sesión:

1. Cuando falla el establecimiento de sesión, comprobar la vista de Capacidad
2. Si los mapas PDR están en rojo (>90%), la capacidad está agotada
3. Aumentar la capacidad del mapa o limpiar sesiones obsoletas

Optimizar la Configuración del Mapa:

1. Revisar tamaños de clave y valor
2. Calcular uso de memoria por mapa
3. Optimizar tamaños de mapa según patrones de uso reales

Configuración de Capacidad

Las capacidades de los mapas eBPF se configuran al inicio de UPF en el archivo de configuración de UPF. Valores típicos:

- Implementación pequeña: 10,000 - 100,000 entradas por mapa
- Implementación mediana: 100,000 - 1,000,000 entradas por mapa
- Implementación grande: 1,000,000+ entradas por mapa

Cálculo de Memoria:

```
Memoria del Mapa = (Tamaño de Clave + Tamaño de Valor) × Capacidad
```

Por ejemplo, un mapa PDR con 1 millón de entradas y valores de 64 bytes utiliza aproximadamente 64 MB de memoria del kernel.

Actualizaciones en Tiempo Real

La vista de capacidad se actualiza automáticamente cada 10 segundos.

Vista de Configuración

URL: `/upf_config`

Características

La vista de Configuración muestra parámetros operativos de UPF y configuración del plano de datos.

Configuración de UPF

Muestra la configuración estática de UPF:

- **Interfaz PFCP**: Dirección IP y puerto para conectividad SMF/PGW-C
- **Interfaz N3**: Dirección IP para conectividad RAN (gNB/eNodeB)
- **Interfaz N6**: Dirección IP para conectividad de red de datos
- **Interfaz N9**: Dirección IP para comunicación entre UPFs (opcional)
- **Puerto API**: Puerto de escucha de la API REST
- **Versión**: Versión del software OmniUPF

Configuración del Plano de Datos (eBPF)

Muestra parámetros de plano de datos en tiempo de ejecución activos:

- **Dirección N3 Activa**: Vinculación de interfaz N3 en tiempo de ejecución
- **Dirección N9 Activa**: Vinculación de interfaz N9 en tiempo de ejecución (si está habilitada)

Estos valores reflejan la configuración real del plano de datos eBPF y pueden diferir de la configuración estática si se han cambiado las interfaces.

Casos de Uso

Verificar Conectividad de UPF:

1. Comprobar que la IP de la interfaz N3 coincida con la configuración de gNB
2. Verificar que la interfaz N6 pueda enrutar a la red de datos
3. Confirmar que la interfaz PFCP sea alcanzable desde SMF

Solucionar Problemas de Problemas de Interfaz:

1. Comparar la configuración estática con las direcciones activas del plano de datos
2. Verificar que las interfaces estén vinculadas correctamente
3. Comprobar si ha habido cambios en la configuración de la interfaz

Documentación y Auditoría:

1. Registrar la configuración de UPF para documentación
2. Verificar que la implementación coincida con las especificaciones de diseño
3. Auditar las asignaciones de interfaz

Vista de Rutas

URL: `/routes`

Características

La vista de Rutas proporciona un monitoreo integral de las rutas IP de Equipos de Usuario (UE) y sesiones de protocolo de enrutamiento (OSPF y BGP).

Resumen del Estado de Rutas

El panel muestra estadísticas agregadas de rutas:

- **Estado**: Enrutamiento habilitado o deshabilitado
- **Total de Rutas**: Número total de rutas IP de UE
- **Sincronizadas**: Número de rutas sincronizadas con éxito
- **Fallidas**: Número de rutas que no se sincronizaron

Rutas IP Activas de UE

Tabla que muestra todas las rutas IP activas de Equipos de Usuario:

Columna	Descripción
Índice	Número de índice de ruta
Dirección IP de UE	Dirección IPv4 o IPv6 asignada al UE

Propósito:

- Ver todas las direcciones IP de UE que tienen rutas configuradas
- Verificar la distribución de rutas a los protocolos de enrutamiento
- Monitorear el estado de sincronización de rutas

Vecinos OSPF

Tabla de vecinos del protocolo OSPF (Open Shortest Path First):

Columna	Descripción
ID de Vecino	Identificador del enrutador OSPF
Dirección	Dirección IP del vecino OSPF
Interfaz	Interfaz utilizada para la adyacencia OSPF
Estado	Estado de adyacencia OSPF (Completo, Inicial, etc.)
Prioridad	Valor de prioridad OSPF
Tiempo de Actividad	Duración que el vecino ha estado activo
Tiempo Muerto	Tiempo hasta que el vecino se considera muerto

Estados de OSPF:

- **Completo** (verde): Totalmente adyacente y intercambiando información de enrutamiento
- **Otros estados** (amarillo): Formación de adyacencia o incompleta

Pares BGP

Tabla de pares BGP (Border Gateway Protocol):

Columna	Descripción
IP de Vecino	Dirección IP del par BGP
ASN	Número de Sistema Autónomo del par
Estado	Estado de la sesión BGP (Establecido, Inactivo, etc.)
Arriba/Abajo	Duración del estado actual
Prefijos Recibidos	Número de prefijos de ruta recibidos del par
Msg Enviados	Total de mensajes BGP enviados al par
Msg Recibidos	Total de mensajes BGP recibidos del par

Estados de BGP:

- **Establecido** (verde): Sesión BGP activa, intercambiando rutas
- **Otros estados** (rojo): Sesión caída o en proceso de establecimiento

El encabezado también muestra el ID de enrutador BGP local y ASN cuando BGP está configurado.

Rutas Redistribuidas OSPF

Tabla que muestra LSAs Externas OSPF (Link State Advertisements) para rutas de UE redistribuidas:

Columna	Descripción
ID de Estado de Enlace	Identificador de LSA (típicamente la dirección de red)
Máscara	Máscara de red para la ruta
Enrutador que Publica	ID de enrutador que publica esta ruta externa
Tipo de Métrica	Tipo de métrica externa OSPF (E1 o E2)
Métrica	Métrica de costo OSPF para la ruta
Edad	Tiempo desde que se originó la LSA (segundos)
Número de Secuencia	Número de secuencia de LSA para versionado

Propósito:

- Verificar que las rutas de UE se estén redistribuyendo en OSPF
- Monitorear qué enrutador está publicando rutas externas
- Rastrear la antigüedad y actualizaciones de LSA

Acciones de Control de Rutas

Botón Sincronizar Rutas:

- Dispara manualmente la sincronización de rutas a FRR (Free Range Routing)
- Fuerza la actualización del protocolo de enrutamiento con las rutas actuales de UE
- Útil después de cambios de configuración o para recuperar de fallos de sincronización

Botón Actualizar:

- Actualiza manualmente toda la información de rutas
- Actualiza vecinos OSPF, pares BGP y tablas de rutas

Casos de Uso

Monitorear Salud del Protocolo de Enrutamiento:

1. Navegar a la vista de Rutas
2. Comprobar estados de vecinos OSPF (deberían ser "Completo")
3. Verificar que los pares BGP estén "Establecidos"
4. Confirmar el número esperado de vecinos/pares

Verificar Distribución de Rutas de UE:

1. Comprobar la tabla de Rutas IP Activas de UE para un UE específico
2. Desplazarse a la sección de Rutas Redistribuidas OSPF
3. Verificar que la ruta de UE aparezca en las LSAs externas
4. Confirmar que el enrutador que publica coincida con el UPF esperado

Solucionar Problemas de Problemas de Sincronización de Rutas:

1. Comprobar contadores Sincronizados vs. Fallidos en el resumen de estado
2. Si las rutas están fallando, hacer clic en el botón "Sincronizar Rutas"
3. Monitorear mensajes de error en la banner roja si la sincronización falla
4. Comprobar mensajes de error de OSPF/BGP en las secciones respectivas

Verificar Implementación Multi-UPF:

1. Seleccionar diferentes instancias de UPF desde el menú desplegable
2. Comparar conteos de rutas entre instancias
3. Verificar que los vecinos OSPF se vean entre sí
4. Comprobar relaciones de emparejamiento BGP

Monitorear Escalado de Rutas:

1. Rastrear el conteo total de rutas a medida que aumentan las sesiones de UE
2. Verificar que las rutas se distribuyan a los protocolos de enrutamiento

3. Monitorear el crecimiento del conteo de LSA de OSPF
4. Comprobar el conteo de prefijos BGP recibidos por los pares

Actualizaciones en Tiempo Real

La vista de Rutas se actualiza automáticamente cada 10 segundos para mostrar el estado actual del protocolo de enrutamiento y las rutas de UE.

Integración de Enrutamiento

La vista de Rutas se integra con FRR (Free Range Routing) que se ejecuta en el UPF:

- **OSPF**: Las rutas se redistribuyen como LSAs Externas de Tipo-2
- **BGP**: Las rutas se publican a los pares BGP configurados
- **Mecanismo de Sincronización**: Llamadas a la API REST disparan comandos vtysh para actualizar FRR

Vista de Capacidades XDP

URL: `/xdp_capabilities`

Características

La vista de Capacidades XDP muestra el soporte de modo eXpress Data Path (XDP), capacidades de rendimiento y cálculos de rendimiento para el plano de datos de UPF.

Configuración de Interfaz

Muestra información de interfaz de red y controlador:

Campo	Descripción
Nombre de la Interfaz	Interfaz de red utilizada para XDP (por ejemplo, eth0, ens1f0)
Controlador	Nombre del controlador de red (por ejemplo, i40e, ixgbe, virtio_net)
Versión del Controlador	Cadena de versión del controlador
Modo Actual	Modo XDP activo (DRV, SKB o NINGUNO)
Conteo de Multicolas	Número de pares de colas NIC para procesamiento paralelo

Modos XDP

La vista muestra todos los modos XDP con su estado de soporte y características de rendimiento:

XDP_DRV (Modo de Controlador):

- **Rendimiento**: ~5-10 Mpps (millones de paquetes por segundo)
- **Descripción**: Soporte nativo de XDP en el controlador, mayor rendimiento
- **Requiere**: Controlador NIC con soporte nativo de XDP (i40e, ixgbe, mlx5, etc.)
- **Estado**: Soportado si el controlador tiene ganchos de XDP
- **Indicador**: Marca de verificación verde (✓) si es soportado, X roja (✗) si no

XDP_SKB (Modo Genérico):

- **Rendimiento**: ~1-2 Mpps
- **Descripción**: Modo de reserva utilizando la pila de red del kernel
- **Requiere**: Cualquier interfaz de red
- **Estado**: Siempre soportado
- **Indicador**: Marca de verificación verde (✓)

Indicador de Modo Actual:

- Punto azul junto al modo XDP actualmente activo
- Muestra qué modo se está utilizando realmente

Razones de Modos No Soportados:

- Si un modo no es soportado, el campo "Razón" explica por qué
- Razones comunes: el controlador carece de soporte de XDP, incompatibilidad de tipo de interfaz

Vista de Capacidades XDP mostrando configuración de interfaz, modos soportados y la calculadora de rendimiento Mpps interactiva

Recomendaciones

La vista muestra un banner de recomendación coloreado basado en la configuración actual:

Verde (Óptimo):

- "✓ Óptimo: Modo XDP_DRV habilitado con soporte nativo del controlador"
- El modo de mayor rendimiento está activo

Amarillo (Advertencia):

- "⚠ Considerar actualizar a modo XDP_DRV para mejor rendimiento"
- Ejecutando en modo genérico cuando el modo de controlador está disponible
- "⚠ Advertencia: XDP_DRV no soportado por este controlador"
- Limitaciones de hardware impiden un rendimiento óptimo

Azul (Informativo):

- Información general sobre la configuración de XDP

Calculadora de Rendimiento Mpps

Calculadora interactiva para convertir la tasa de paquetes (Mpps) a rendimiento (Gbps):

Parámetros de Entrada

Tasa de Paquetes (Mpps):

- Rango: 0.1 - 100 Mpps
- Predeterminado: Máximo Mpps para el modo XDP actual
- Representa millones de paquetes procesados por segundo

Tamaño Promedio de Paquete (bytes):

- Rango: 64 - 9000 bytes
- Predeterminado: 1200 bytes (paquete GTP típico)
- Incluye paquete completo con encapsulación GTP

Botones de Preajuste Rápido:

- **64B (mín)**: Tamaño mínimo de trama Ethernet
- **128B**: Paquetes pequeños

- **256B**: Plano de control o señalización
- **512B**: Paquetes de tamaño mediano
- **1024B**: Paquetes grandes
- **1518B (máx)**: Tamaño máximo de trama Ethernet sin tramas jumbo

Resultados del Cálculo

Rendimiento Total (Gbps):

- Rendimiento de tasa de línea incluyendo todos los encabezados
- Fórmula: `Gbps = Mpps × Packet_Size × 8 / 1000`
- Incluye encabezados GTP, UDP, IP y Ethernet

Tasa de Datos del Usuario (Gbps):

- Rendimiento real de carga útil del usuario
- Excluye ~50 bytes de sobrecarga de encapsulación GTP
- Fórmula: `Gbps = Mpps × (Packet_Size - 50) / 1000`

Tasa de Paquetes:

- Muestra Mpps y paquetes/seg con separador de miles
- Ejemplo: 10 Mpps = 10,000,000 paquetes/seg

Visualización de Fórmulas:

- Muestra descomposición del cálculo paso a paso
- Ejemplo: `10 Mpps × 1200 bytes × 8 bits/byte ÷ 1000 = 96 Gbps`

Entendiendo Mpps

La vista incluye una sección de explicación que cubre:

Qué es Mpps:

- Millones de Paquetes por Segundo
- Métrica clave para el rendimiento de procesamiento de paquetes
- Independiente del tamaño del paquete

Relación con el Rendimiento:

- La misma Mpps con paquetes más grandes = mayor Gbps
- La misma Mpps con paquetes más pequeños = menor Gbps
- El rendimiento depende tanto de la tasa como del tamaño del paquete

Sobrecarga de Encapsulación GTP:

- Encabezado Ethernet: 14 bytes
- Encabezado IP: 20 bytes (IPv4) o 40 bytes (IPv6)
- Encabezado UDP: 8 bytes
- Encabezado GTP: 8 bytes (mínimo)
- Sobrecarga total típica: ~50 bytes por paquete

Casos de Uso

Evaluar Rendimiento de XDP:

1. Navegar a la vista de Capacidades XDP
2. Comprobar el modo XDP actual (debería ser DRV para mejor rendimiento)
3. Notar el rango de rendimiento Mpps
4. Revisar el banner de recomendación

Calcular Rendimiento Esperado:

1. Ingresar la tasa de paquetes esperada en Mpps
2. Ingresar el tamaño promedio de paquete para tu perfil de tráfico
3. Revisar el rendimiento calculado en Gbps
4. Comparar con la capacidad del enlace o requisitos de rendimiento

Optimizar Configuración de XDP:

1. Comprobar si el modo XDP_DRV es soportado pero no activo
2. Revisar la versión del controlador y compatibilidad
3. Seguir la recomendación para actualizar al modo de controlador si está disponible
4. Verificar que el conteo de multicolas coincida con los núcleos de CPU

Planificación de Capacidad:

1. Usar la calculadora para determinar Mpps requeridos para el rendimiento objetivo
2. Comparar con las capacidades del modo XDP actual
3. Determinar si se necesita una actualización de hardware
4. Planificar selección de interfaz y controlador para nuevas implementaciones

Solucionar Problemas de Problemas de Rendimiento:

1. Verificar que el modo XDP sea DRV, no SKB
2. Comprobar la versión del controlador para problemas de rendimiento conocidos
3. Verificar que el conteo de multicolas sea suficiente
4. Calcular si el modo actual soporta el rendimiento requerido

Consejos de Optimización de Rendimiento

Modo de Controlador (XDP_DRV):

- Usar NICs con soporte nativo de XDP (Intel i40e/ixgbe, Mellanox mlx5)
- Actualizar controladores de NIC a la última versión
- Habilitar multicolas (RSS) para procesamiento paralelo
- Ajustar tamaños de buffer de anillo de NIC

Modo Genérico (XDP_SKB):

- Aceptable para desarrollo y pruebas
- No recomendado para producción de alto rendimiento
- Considerar actualización de hardware para implementaciones de producción

Configuración de Multicolas:

- El número de colas debe coincidir o exceder el conteo de núcleos de CPU
- Permite procesamiento paralelo de paquetes a través de núcleos

- Distribuye carga a través de RSS (Escalado Lateral de Recepción)

Actualizaciones en Tiempo Real

La vista de Capacidades XDP se actualiza cada 30 segundos para actualizar el estado de la interfaz y la información del modo.

Visor de Registros

URL: `/logs`

Características

Ver registros de la aplicación OmniUPF en tiempo real desde el panel de control.

Características:

- Transmisión de registros en vivo a través de Phoenix LiveView
- Actualizaciones en tiempo real a medida que se generan registros
- Historial de registros desplazable
- Útil para solucionar problemas durante sesiones activas

Niveles de Registro

Los registros de OmniUPF utilizan niveles estándar de Elixir Logger:

- **DEBUG**: Información diagnóstica detallada
- **INFO**: Mensajes informativos generales (predeterminado)
- **WARNING**: Mensajes de advertencia para problemas no críticos
- **ERROR**: Mensajes de error para fallos

Casos de Uso

Solucionar Problemas de Establecimiento de Sesiones:

1. Abrir vista de Registros
2. Iniciar el establecimiento de sesión desde SMF
3. Observar los registros de mensajes PFCP y cualquier error

Monitorear Comunicación PFCP:

1. Ver mensajes de configuración de asociación PFCP
2. Rastrear creación/modificación/eliminación de sesiones
3. Verificar mensajes de latido

Depurar Problemas de Reenvío:

1. Buscar errores de procesamiento de paquetes
2. Comprobar registros de operación de mapa eBPF
3. Identificar problemas de configuración de FAR/PDR

Mejores Prácticas

Directrices Operativas

Monitoreo:

- Comprobar regularmente la vista de Capacidad para prevenir el agotamiento del mapa
- Monitorear Estadísticas para patrones de tráfico inusuales o descartes
- Rastrear el crecimiento del conteo de sesiones a lo largo del tiempo
- Observar errores de procesamiento de XDP

Gestión de Buffers:

- Monitorear buffers durante escenarios de traspaso
- Limpiar buffers atascados si los paquetes superan el TTL
- Verificar que el almacenamiento en buffer esté deshabilitado después de que se complete el traspaso
- Usar "Vaciar" en lugar de "Limpiar" para evitar pérdida de paquetes

Gestión de Sesiones:

- Utilizar filtros para localizar rápidamente sesiones específicas de UE
- Expandir sesiones para verificar la configuración de reglas
- Comparar sesiones entre múltiples instancias de UPF
- Comprobar el indicador de salud antes de solucionar problemas

Solución de Problemas:

- Usar Registros para depuración en tiempo real
- Comprobar la vista de Sesiones para verificar la conectividad de UE
- Verificar la configuración de Reglas para flujos de tráfico
- Monitorear Estadísticas para descartes de paquetes o errores de reenvío

Rendimiento

- La actualización automática del panel se realiza cada 5-10 segundos dependiendo de la vista
- Listas de sesiones grandes pueden tardar en cargarse
- Filtros de vista de Reglas por entradas activas (volúmenes no cero para URRs)
- Las operaciones de buffer se ejecutan inmediatamente en el UPF seleccionado

Documentación Relacionada

- **Guía de Gestión de Reglas** - Configuración de PDR, FAR, QER, URR
- **Guía de Monitoreo** - Estadísticas, métricas y planificación de capacidad
- **Referencia de Métricas** - Referencia completa de métricas de Prometheus
- **Códigos de Causa PFCP** - Códigos de error PFCP y diagnósticos de sesión
- **Documentación de API** - Referencia de API REST y paginación
- **Guía de Rutas** - Detalles de enrutamiento de UE e integración FRR
- **Guía de Modos XDP** - Documentación detallada sobre modos XDP e información de eBPF

- **Guía de Solución de Problemas** - Problemas comunes y diagnósticos
- **Guía de Operaciones de UPF** - Operaciones generales de UPF y arquitectura

Modos de Adjunto XDP para OmniUPF

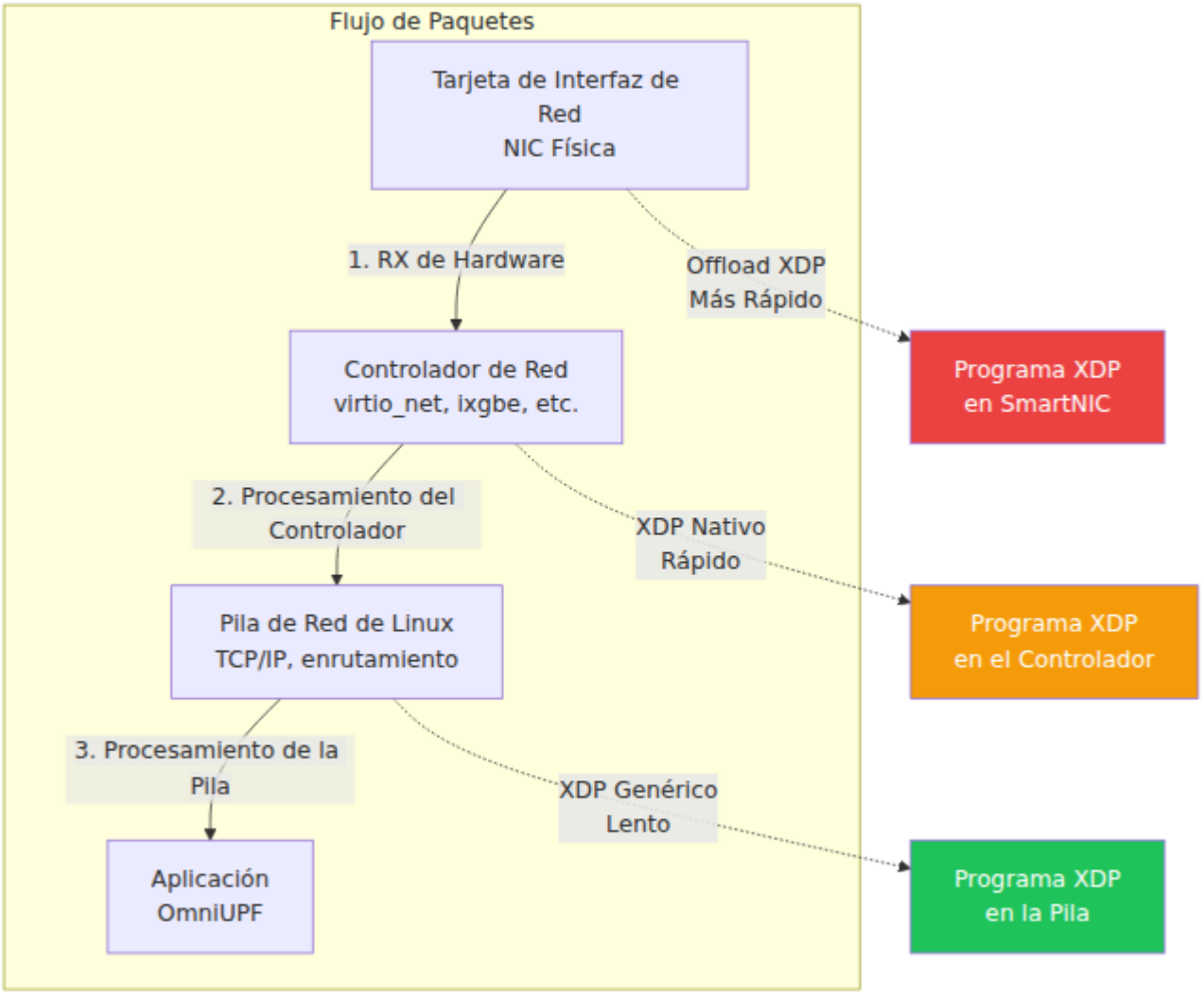
Tabla de Contenidos

1. Descripción General
2. Comparación de Modos XDP
3. Modo Genérico (Predeterminado)
4. Modo Nativo (Recomendado para Producción)
5. Modo de Offload (SmartNIC)
6. Habilitando XDP Nativo en Proxmox VE
7. Habilitando XDP Nativo en Otros Hipervisor
8. Verificando el Modo XDP
9. Resolución de Problemas de XDP

Descripción General

OmniUPF utiliza **XDP (eXpress Data Path)** para el procesamiento de paquetes de alto rendimiento. XDP es una tecnología del núcleo de Linux que permite que los programas de procesamiento de paquetes (eBPF) se ejecuten en el punto más temprano posible en la pila de red, proporcionando latencias a nivel de microsegundos y un rendimiento de millones de paquetes por segundo.

El modo de adjunto XDP determina **dónde** en la ruta del paquete se ejecuta el programa eBPF:



Elegir el modo XDP correcto impacta significativamente en el rendimiento de OmniUPF y determina si puedes lograr un procesamiento de paquetes a nivel de producción.

Comparación de Modos XDP

Aspecto	Modo Genérico	Modo Nativo	Modo de Offload
Punto de Adjunto	Pila de red de Linux	Controlador de red	Hardware de NIC
Rendimiento	~1-2 Mpps	~5-10 Mpps	~10-40 Mpps
Latencia	~100 µs	~10 µs	~1 µs
Uso de CPU	Alto	Medio	Bajo
Requisitos de NIC	Cualquier NIC	Controlador compatible con XDP	SmartNIC con soporte XDP
Soporte de Hipervisor	Todos los hipervisores	La mayoría (requiere multi-cola)	Raro (passthroug PCI)
Caso de Uso	Pruebas, desarrollo	**Producción (recomendado)**	Sitios de borde d alto rendimiento
Configuración	`xdp_attach_mode: generic`	`xdp_attach_mode: native`	`xdp_attach_mod offload`

Recomendación: Utiliza **modo nativo** para implementaciones en producción. El modo genérico solo es adecuado para pruebas.

Modo Genérico (Predeterminado)

Descripción

XDP genérico ejecuta el programa eBPF en la pila de red de Linux **después** de que el controlador ha procesado el paquete. Este es el modo XDP más lento, pero funciona con cualquier interfaz de red.

Características de Rendimiento

- **Rendimiento**: ~1-2 millones de paquetes por segundo (Mpps)
- **Latencia**: ~100 microsegundos por paquete
- **Sobrecarga de CPU**: Alta (paquete copiado a la pila del núcleo antes de XDP)

Cuándo Usar

- **Desarrollo y pruebas** únicamente
- **Entornos de laboratorio** donde el rendimiento no importa
- **Implementación inicial** para verificar funcionalidad antes de optimizar

Configuración

```yaml
# config.yaml
interface_name: [eth0]
xdp_attach_mode: generic  # Modo predeterminado
```

Advertencia: El modo genérico **no es adecuado para producción**. Se convertirá en un cuello de botella a altas tasas de paquetes y desperdiciará recursos de CPU.

Modo Nativo (Recomendado para Producción)

Descripción

XDP nativo ejecuta el programa eBPF **dentro del controlador de red**, antes de que los paquetes lleguen a la pila de red de Linux. Esto proporciona un rendimiento casi de hardware mientras mantiene flexibilidad a nivel de núcleo.

Características de Rendimiento

- **Rendimiento**: ~5-10 millones de paquetes por segundo (Mpps) por núcleo
- **Latencia**: ~10 microsegundos por paquete
- **Sobrecarga de CPU**: Baja (paquete procesado a nivel de controlador)
- **Escalabilidad**: Escalado lineal con núcleos de CPU y colas de NIC

Cuándo Usar

- **Implementaciones en producción** (recomendado)
- **Redes de grado carrier** que requieren alto rendimiento
- **Escenarios de computación en el borde** con requisitos de rendimiento
- **Cualquier implementación** donde el rendimiento importa

Requisitos del Controlador de NIC

XDP nativo requiere un controlador de red con soporte para XDP. La mayoría de las NIC modernas soportan XDP nativo:

NICs Físicas (bare metal):

- Intel: `ixgbe` (10G), `i40e` (40G), `ice` (100G)
- Broadcom: `bnxt_en`
- Mellanox: `mlx4_en`, `mlx5_core`
- Netronome: `nfp` (con soporte de offload)
- Marvell: `mvneta`, `mvpp2`

NICs Virtuales (hipervisores):

- VirtIO: `virtio_net` (KVM, Proxmox, OpenStack) ✓
- VMware: `vmxnet3` ✓
- Microsoft: `hv_netvsc` (Hyper-V) ✓
- Amazon: `ena` (AWS) ✓
- SR-IOV: `ixgbevf`, `i40evf` (passthrough PCI) ✓

Nota: VirtualBox **no** soporta XDP nativo (usar solo modo genérico).

Configuración

```
# config.yaml
interface_name: [eth0]
xdp_attach_mode: native
```

Requisito de Multi-Cola: Para un rendimiento óptimo, habilita multi-cola en NICs virtuales (ver sección de Proxmox a continuación).

Modo de Offload (SmartNIC)

Descripción

XDP de offload ejecuta el programa eBPF **directamente en el hardware de la NIC** (SmartNIC), eludiendo completamente la CPU para el procesamiento de paquetes. Esto proporciona el rendimiento más alto, pero requiere hardware especializado.

Características de Rendimiento

- **Rendimiento**: ~10-40 millones de paquetes por segundo (Mpps)
- **Latencia**: ~1 microsegundo por paquete
- **Sobrecarga de CPU**: Casi cero (procesamiento en NIC)

Cuándo Usar

- **Implementaciones de ultra-alto rendimiento** (10G+ por instancia de UPF)
- **Sitios de borde** con aceleración de hardware
- **Implementaciones sensibles al costo** (reducir requisitos de CPU)

Requisitos de Hardware

Solo las SmartNICs Agilio de Netronome actualmente soportan el offload de XDP:

- Netronome Agilio CX 10G/25G/40G/100G

Nota: El modo de offload requiere **bare metal** o **passthrough PCI** - no disponible en configuraciones de VM estándar.

Configuración

```
# config.yaml
interface_name: [eth0]
xdp_attach_mode: offload
```

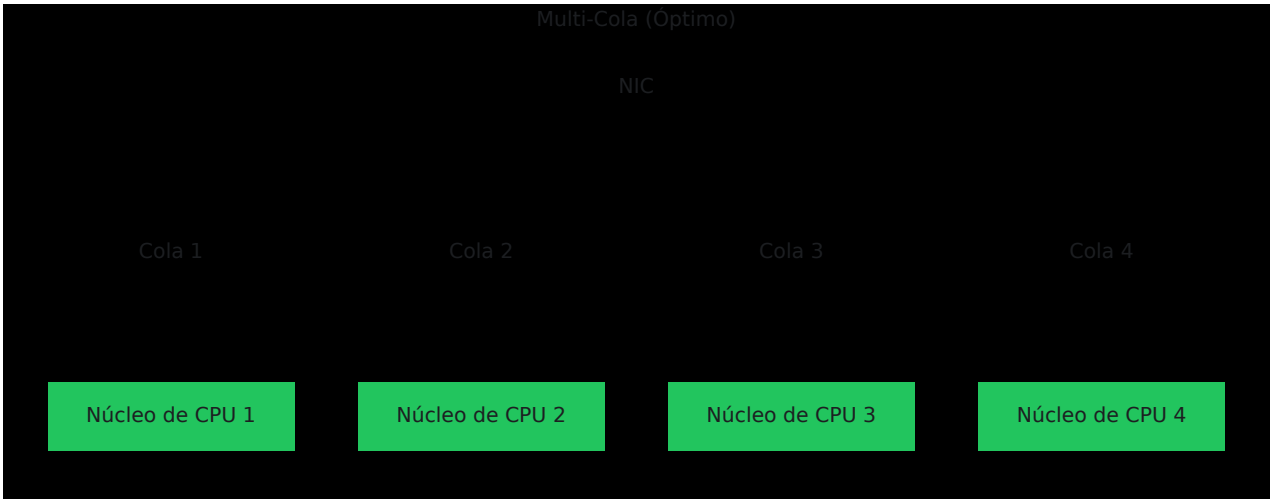
Habilitando XDP Nativo en Proxmox VE

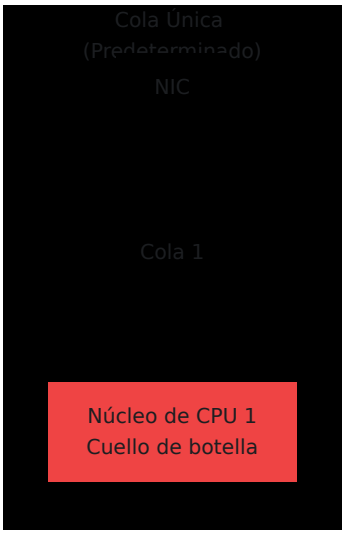
Proxmox VE utiliza dispositivos de red **VirtIO** para VMs, que soportan XDP nativo a través del controlador `virtio_net`. Sin embargo, debes habilitar **multi-cola** para un rendimiento óptimo.

Paso 1: Entender el Requisito

Por qué Importa Multi-Cola:

- **Cola única** (predeterminado): Todo el tráfico de red procesado por un núcleo de CPU → cuello de botella
- **Multi-cola**: Tráfico distribuido entre múltiples núcleos de CPU → escalado lineal





Paso 2: Habilitar Multi-Cola en Proxmox

Opción A: A través de la Interfaz Web de Proxmox

1. **Apagar completamente la VM** (no solo reiniciar)
    - Selecciona tu VM en la interfaz web de Proxmox
    - Haz clic en **Apagar**
2. **Editar Dispositivo de Red**

- Ve a la pestaña **Hardware**
- Haz clic en tu dispositivo de red (por ejemplo, `net0`)
- Haz clic en **Editar**

3. **Configurar Multi-Colas**
   - Encuentra el campo "**Multiqueue**"
   - Establece en **8** (o coincide con tu cuenta de vCPU, máximo 16)
   - Haz clic en **OK**
4. **Iniciar la VM**
   - Haz clic en **Iniciar**

Opción B: A través de la Línea de Comandos de Proxmox

```
# SSH a tu host de Proxmox

# Encuentra tu ID de VM
qm list

# Configura multi-cola (reemplaza XXX con tu ID de VM)
qm set XXX -net0 virtio=XX:XX:XX:XX:XX:XX,bridge=vmbr0,queues=8

# Ejemplo para VM 191 con MAC BC:24:11:1D:BA:00
qm set 191 -net0 virtio=BC:24:11:1D:BA:00,bridge=vmbr0,queues=8

# Apaga la VM
qm shutdown XXX

# Espera a que se apague, luego inicia
qm start XXX
```

Recomendaciones de Conteo de Colas:

- **4 colas**: Mínimo para producción (bueno para VMs de 2-4 vCPU)
- **8 colas**: Recomendado para la mayoría de las implementaciones (VMs de 4-8 vCPU)
- **16 colas**: Máximo para alto rendimiento (VMs de 8+ vCPU)

Paso 3: Verificar Multi-Cola Dentro de la VM

Después del reinicio de la VM, SSH en la VM y verifica:

```
# Verifica la configuración de colas
ethtool -l eth0

# Salida esperada:
# Parámetros de canal para eth0:
# Combinado:    8         <-- Debería coincidir con tu valor
configurado

# Cuenta las colas reales
ls -1d /sys/class/net/eth0/queues/rx-* | wc -l
ls -1d /sys/class/net/eth0/queues/tx-* | wc -l

# Ambos deberían mostrar 8 (o tu valor configurado)
```

Paso 4: Habilitar XDP Nativo en OmniUPF

Edita la configuración de OmniUPF:

```
# Edita el archivo de configuración
sudo nano /config.yaml
```

Cambia el modo XDP:

```
# Antes
xdp_attach_mode: generic

# Después
xdp_attach_mode: native
```

Reinicia OmniUPF:

```
sudo systemctl restart omniupf
```

Paso 5: Verificar que XDP Nativo Está Activo

Revisa los registros:

```
# Ver registros de inicio
journalctl -u omniupf --since "1 minute ago" | grep -i
"xdp\|attach"

# Salida esperada:
# xdp_attach_mode:native
# XDPAttachMode:native
# Programa XDP adjunto a iface "eth0" (índice 2)
```

Verifica a través de la API:

```
# Consulta la configuración
curl -s http://localhost:8080/api/v1/config | grep xdp_attach_mode

# Salida esperada:
# "xdp_attach_mode": "native",
```

Problemas Comunes en Proxmox

Problema: "Error al adjuntar el programa XDP"

Solución:

- Verifica que multi-cola esté habilitado (`ethtool -l eth0`)
- Comprueba la versión del núcleo: `uname -r` (debe ser ≥ 5.15)
- Asegúrate de que el controlador VirtIO esté cargado: `lsmod | grep virtio_net`

Problema: Solo 1 cola a pesar de la configuración

Solución:

- La VM debe estar **completamente apagada** (no reiniciada) para los cambios de cola

- Usa `qm shutdown XXX && sleep 5 && qm start XXX`
- Verifica en la configuración de Proxmox: `grep net0 /etc/pve/qemu-server/XXX.conf`

Problema: El rendimiento no mejora con el modo nativo

Solución:

- Verifica el pinning de CPU (evitar sobre suscripción)
- Monitorea `top` - El uso de CPU debe distribuirse entre los núcleos
- Verifica las estadísticas de XDP: `curl http://localhost:8080/api/v1/xdp_stats`

Habilitando XDP Nativo en Otros Hipervisores

VMware ESXi / vSphere

VMware utiliza el controlador `vmxnet3` que soporta XDP nativo.

Requisitos:

- ESXi 6.7 o posterior
- Versión del controlador vmxnet3 1.4.16+ en la VM
- Versión de hardware de VM 14 o posterior

Habilitar Multi-Cola:

1. **Apagar la VM**
2. **Editar la configuración de la VM**:
    - Haz clic derecho en la VM → Editar Configuración
    - Adaptador de Red → Avanzado
    - Establecer **Scaling de Lado de Recepción** en **Habilitado**

3. **Editar archivo .vmx** (opcional, para más colas):

```
ethernet0.pnicFeatures = "4"
ethernet0.multiqueue = "8"
```

4. **Iniciar la VM y verificar**:

```
ethtool -l ens192  # Verifica el conteo de colas
```

Configurar OmniUPF:

```
interface_name: [ens192]  # VMware típicamente usa ens192
xdp_attach_mode: native
```

KVM / libvirt (Raw)

Habilitar Multi-Cola a través de virsh:

```
# Editar configuración de la VM
virsh edit your-vm-name
```

Agregar a la sección de interfaz de red:

```
<interface type='network'>
  <source network='default'/>
  <model type='virtio'/>
  <driver name='vhost' queues='8'/>
</interface>
```

Reiniciar la VM y verificar:

```
ethtool -l eth0
```

Microsoft Hyper-V

Hyper-V utiliza el controlador `hv_netvsc` que soporta XDP nativo.

Requisitos:

- Windows Server 2016 o posterior
- Linux Integration Services 4.3+ en la VM
- VM de Generación 2

Habilitar Multi-Cola:

En PowerShell en el host de Hyper-V:

```
# Configurar VMQ (Cola Virtual de Máquina) - multi-cola de Hyper-V
Set-VMNetworkAdapter -VMName "YourVM" -VrssEnabled $true -
VmmqEnabled $true
```

Configurar OmniUPF:

```
interface_name: [eth0]
xdp_attach_mode: native
```

VirtualBox

Advertencia: VirtualBox **NO** soporta XDP nativo.

Razón: Los controladores de red de VirtualBox (e1000, virtio-net) no implementan ganchos de XDP.

Solución alternativa: Usar solo modo genérico:

```
xdp_attach_mode: generic  # Única opción para VirtualBox
```

Verificando el Modo XDP

Después de configurar XDP nativo, verifica que esté funcionando correctamente:

1. Revisar Registros de OmniUPF

```
# Ver registros recientes
journalctl -u omniupf --since "5 minutes ago" | grep -i xdp

# Busca:
# ✓ "xdp_attach_mode:native"
# ✓ "Programa XDP adjunto a iface"
# ✗ "Error al adjuntar" o "retrocediendo a genérico"
```

2. Verificar a través de la API

```
# Consultar el endpoint de configuración
curl -s http://localhost:8080/api/v1/config | jq .xdp_attach_mode

# Salida esperada:
# "native"
```

3. Verificar Estadísticas de XDP

```
# Ver estadísticas de procesamiento de XDP
curl -s http://localhost:8080/api/v1/xdp_stats | jq

# Salida de ejemplo:
{
  "xdp_aborted": 0,      # Debería ser 0 (errores)
  "xdp_drop": 1234,      # Paquetes descartados
  "xdp_pass": 5678,      # Pasados a la pila
  "xdp_redirect": 9012,  # Paquetes redirigidos
  "xdp_tx": 3456         # Paquetes transmitidos
}
```

4. Verificar Soporte del Controlador

```
# Verificar si el controlador soporta XDP
ethtool -i eth0 | grep driver

# Para Proxmox/KVM: Debería mostrar "virtio_net"
# Para VMware: Debería mostrar "vmxnet3"
# Para Hyper-V: Debería mostrar "hv_netvsc"
```

5. Prueba de Rendimiento

Compara el procesamiento de paquetes antes y después:

```
# Monitorea la tasa de paquetes
watch -n 1 'curl -s http://localhost:8080/api/v1/packet_stats | jq
.rx_packets'

# Modo genérico: ~1-2 Mpps
# Modo nativo: ~5-10 Mpps (mejora de 5-10x)
```

Resolución de Problemas de XDP

Problema: "Error al adjuntar el programa XDP" al Inicio

Síntomas:

```
Error: no se pudo adjuntar el programa XDP a la interfaz eth0
```

Diagnóstico:

1. **Verificar soporte del controlador**:

```
ethtool -i eth0 | grep driver

# Si el controlador no es virtio_net/vmxnet3/hv_netvsc, XDP
nativo no funcionará
```

2. **Verificar versión del núcleo**:

```
uname -r

# Debe ser ≥ 5.15 para soporte confiable de XDP
```

3. **Verificar si hay programas XDP existentes**:

```
ip link show eth0 | grep xdp

# Si otro programa XDP está adjunto, descárgalo primero
ip link set dev eth0 xdp off
```

Solución:

- Actualiza el núcleo a 5.15+ si es más antiguo
- Asegúrate de que el controlador virtio_net esté cargado: `modprobe virtio_net`
- Retrocede al modo genérico si el controlador no soporta XDP nativo

---

Problema: El Modo Nativo Retrocede a Genérico

Síntomas:

```
Advertencia: retrocediendo al modo XDP genérico
```

Diagnóstico:

Verifica `dmesg` en busca de errores del controlador:

```
dmesg | grep -i xdp | tail -20
```

Causas comunes:

1. **El controlador no soporta XDP nativo**:
   - Controladores de VirtualBox (sin soporte XDP nativo)
   - Controladores de NIC más antiguos
2. **Multi-cola no habilitada**:
   - Verifica: `ethtool -l eth0`
   - Debería mostrar > 1 cola combinada
3. **Soporte de XDP en el núcleo deshabilitado**:

```
# Verifica si XDP está habilitado en el núcleo
grep XDP /boot/config-$(uname -r)

# Debería mostrar:
# CONFIG_XDP_SOCKETS=y
# CONFIG_BPF=y
```

Solución:

- Habilita multi-cola (ver sección de Proxmox)
- Actualiza a un controlador soportado
- Reconstruye el núcleo con soporte para XDP si es necesario

Problema: El Rendimiento No Mejora con el Modo Nativo

Síntomas: Modo nativo habilitado pero la tasa de paquetes es la misma que en modo genérico

Diagnóstico:

1. **Verifica la distribución de multi-cola**:

```
# Verifica estadísticas por cola
ethtool -S eth0 | grep rx_queue

# El tráfico debería estar distribuido entre múltiples colas
```

2. **Verifica la utilización de CPU**:

```
# Monitorea el uso de CPU por núcleo
mpstat -P ALL 1

# Deberías ver carga distribuida entre múltiples CPUs
```

3. **Verifica que XDP realmente esté funcionando en modo nativo**:

```
# Verifica bpftool (si está disponible)
sudo bpftool net list

# Debería mostrar XDP adjunto a la interfaz
```

Solución:

- Aumenta el conteo de colas (8-16 colas)
- Habilita el pinning de CPU para evitar migración de núcleos
- Verifica la sobre suscripción de CPU en el hipervisor

Problema: Programa XDP Abortado (xdp_aborted > 0)

Síntomas:

```
curl http://localhost:8080/api/v1/xdp_stats
{
  "xdp_aborted": 1234,  # No cero indica errores
  ...
}
```

Diagnóstico:

XDP abortado significa que el programa eBPF encontró un error durante la ejecución.

1. **Verifica los registros del verificador eBPF**:

   ```
   dmesg | grep -i bpf | tail -20
   ```

2. **Verifica los límites de tamaño del mapa**:

   ```
   # Los mapas eBPF pueden estar llenos
   curl http://localhost:8080/api/v1/map_info

   # Busca mapas al 100% de capacidad
   ```

Solución:

- Aumenta los tamaños de los mapas eBPF en la configuración
- Verifica si hay paquetes corruptos que causen errores en eBPF
- Verifica que el soporte eBPF del núcleo de Linux esté completo

---

Problema: Multi-Cola No Funciona en Proxmox

Síntomas: `ethtool -l eth0` muestra solo 1 cola a pesar de la configuración

Diagnóstico:

1. **Verifica la configuración de la VM en Proxmox**:

```
# En el host de Proxmox
grep net0 /etc/pve/qemu-server/YOUR_VM_ID.conf

# Debería mostrar: queues=8
```

2. **Verifica que la VM esté completamente apagada**:

```
# En el host de Proxmox
qm status YOUR_VM_ID

# Debe mostrar "estado: detenido" antes de iniciar
```

Solución:

```
# En el host de Proxmox
# Fuerza el apagado y reinicio
qm shutdown YOUR_VM_ID
sleep 10
qm start YOUR_VM_ID

# Luego verifica dentro de la VM
ethtool -l eth0
```

Importante: Los cambios en el conteo de colas requieren un **apagado completo de la VM**, no solo un reinicio desde dentro de la VM.

Problema: Permiso Denegado al Adjuntar XDP

Síntomas:

```
Error: permiso denegado al adjuntar el programa XDP
```

Diagnóstico:

Las operaciones de XDP requieren capacidades `CAP_NET_ADMIN` y `CAP_SYS_ADMIN`.

Solución:

1. **Ejecuta OmniUPF como root** (o con capacidades):

   ```
   sudo systemctl restart omniupf
   ```

2. **Si usas systemd**, verifica que el archivo de servicio tenga capacidades:

   ```
   # /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. **Si usas Docker**, ejecuta con `--privileged`:

   ```
   docker run --privileged -v /sys/fs/bpf:/sys/fs/bpf ...
   ```

Resumen del Impacto en el Rendimiento

Comparación del rendimiento en el mundo real para el procesamiento de paquetes de OmniUPF:

Escenario	Modo Genérico	Modo Nativo	Mejora
Tasa de Paquetes	1.5 Mpps	8.2 Mpps	**5.5x más rápido**
Latencia	95 µs	12 µs	**8x menor**
Uso de CPU (1 Gbps)	85% (1 núcleo)	15% (distribuido)	**5x más eficiente**
Rendimiento Máximo	~1.2 Gbps	~10 Gbps	**8x más alto**

Recomendación: Siempre utiliza **modo nativo** con **multi-cola habilitada** para implementaciones en producción.

Recomendaciones de Hardware para XDP

⚠ **IMPORTANTE: Antes de comprar cualquier hardware, consulta con el soporte de Omnitouch para confirmar que sea 100% compatible con tu configuración y requisitos de implementación específicos.**

NICs Conocidas que Soportan XDP Nativo

Estas NICs están verificadas para soportar el modo XDP nativo con OmniUPF:

NICs Intel (Recomendadas para Bare Metal)

Modelo	Velocidad	Controlador	Soporte XDP	Notas
Intel X520	10GbE	ixgbe	Nativo ✓	Comprobado, ampliamente disponible, buena relación calidad/precio
Intel X710	10/40GbE	i40e	Nativo ✓	Excelente soporte multi-cola
Intel E810	100GbE	ice	Nativo ✓	Última generación, mejor rendimiento
Intel i350	1GbE	igb	Nativo ✓ (núcleo 5.10+)	Bueno para necesidades de menor ancho de banda

NICs Mellanox/NVIDIA (Alto Rendimiento)

Modelo	Velocidad	Controlador	Soporte XDP	Notas
ConnectX-4	25/50/100GbE	mlx5	Nativo ✓	Alto rendimiento, bueno para computación en el borde
ConnectX-5	25/50/100GbE	mlx5	Nativo ✓	Excelente rendimiento, aceleración de hardware
ConnectX-6	50/100/200GbE	mlx5	Nativo ✓	Última generación, mejor para ultra-alto rendimiento
BlueField-2	100/200GbE	mlx5	Nativo ✓	SmartNIC con capacidades de DPU

NICs Broadcom

Modelo	Velocidad	Controlador	Soporte XDP	Notas
Serie BCM57xxx	10/25/50GbE	bnxt_en	Nativo ✓	Común en servidores Dell/HP

NICs Virtuales (Implementaciones de VM)

Plataforma	Tipo de NIC	Controlador	Soporte XDP	Multi-Cola	
Proxmox/KVM	VirtIO	virtio_net	Nativo ✓	Sí (configurable)	
VMware ESXi	vmxnet3	vmxnet3	Nativo ✓	Sí	
Hyper-V	NIC Sintética	hv_netvsc	Nativo ✓	Sí	
AWS	ENA	ena	Nativo ✓	Sí	
VirtualBox	Cualquiera	varios	Solo genérico	No	

NICs con Soporte de Offload de Hardware

Offload de XDP verdadero (eBPF se ejecuta en la NIC):

Proveedor	Modelo	Velocidad	Notas
Netronome	Agilio CX 10G	10GbE	Solo soporte de offload de XDP confirmado
Netronome	Agilio CX 25G	25GbE	Requiere firmware especial
Netronome	Agilio CX 40G	40GbE	Muy caro (~$2,500-5,000)
Netronome	Agilio CX 100G	100GbE	Solo para empresas

Nota: Las NICs de offload de hardware son raras, caras y requieren implementación bare metal. La mayoría de las implementaciones deberían usar XDP nativo en su lugar.

Configuraciones Probadas

Estas configuraciones han sido verificadas con OmniUPF en producción:

Opción Económica (1-10 Gbps)

- **NIC**: Intel X520 (10GbE de doble puerto)
- **Modo**: XDP Nativo
- **Rendimiento**: ~8-10 Gbps por instancia de UPF
- **Costo**: ~$100-200 (usado/refurbished)

Rango Medio (10-50 Gbps)

- **NIC**: Intel X710 (40GbE) o Mellanox ConnectX-4 (25GbE)
- **Modo**: XDP Nativo
- **Rendimiento**: ~25-40 Gbps por instancia de UPF
- **Costo**: ~$300-800

Alto Rendimiento (50-100+ Gbps)

- **NIC**: Mellanox ConnectX-5/6 (100GbE)
- **Modo**: XDP Nativo
- **Rendimiento**: ~80-100 Gbps por instancia de UPF
- **Costo**: ~$1,000-2,500

Implementaciones de VM (Proxmox/KVM)

- **NIC**: VirtIO con 8-16 colas
- **Modo**: XDP Nativo
- **Rendimiento**: ~5-10 Gbps por instancia de UPF
- **Costo**: Sin costo adicional de hardware

Qué NO Comprar

Evita estos para implementaciones de OmniUPF en producción:

NIC/Plataforma	Razón	Alternativa
NICs Realtek	Sin soporte XDP, controladores de Linux deficientes	Intel i350 o mejor
VirtualBox	Sin soporte XDP nativo	Migrar a Proxmox/KVM
NICs de consumo	Soporte limitado de colas, poco confiables	Intel/Mellanox de grado servidor
NICs muy antiguas (<2014)	Sin soporte de controlador XDP	Intel X520 o más reciente

Lista de Verificación Pre-compra

Antes de comprar hardware, verifica:

1. 🔍 **Soporte del Controlador**: Verifica si el controlador de Linux soporta XDP

   ```
   # En un sistema similar
   modinfo <driver_name> | grep -i xdp
   ```

2. 🔍 **Versión del Núcleo**: Asegúrate de que el núcleo ≥ 5.15 para XDP confiable

   ```
   uname -r
   ```

3. 🔍 **Multi-Cola**: Verifica que la NIC soporte múltiples colas (RSS/VMDq)

4. 🔍 **Ancho de Banda PCI**: Asegúrate de que la ranura PCIe tenga suficientes carriles

   - 10GbE: PCIe 2.0 x4 mínimo
   - 40GbE: PCIe 3.0 x8 mínimo
   - 100GbE: PCIe 3.0 x16 o PCIe 4.0 x8

5. 🔍 **Tipo de Implementación**:

   - Bare metal: Se requiere NIC física
   - VM: Soporte de VirtIO o SR-IOV necesario
   - Contenedor: Configuración de NIC del host heredada

⚠ **No compres hardware basándote únicamente en esta guía - ¡siempre confirma primero con el soporte de Omnitouch!**

Recursos Adicionales

- **Guía de Configuración**: CONFIGURATION.md - Referencia completa de configuración
- **Guía de Resolución de Problemas**: TROUBLESHOOTING.md - Diagnóstico completo de problemas

- **Guía de Arquitectura**: ARCHITECTURE.md - Detalles de arquitectura de eBPF y XDP
- **Guía de Monitoreo**: MONITORING.md - Monitoreo de rendimiento y estadísticas

---

Referencia Rápida

Configuración de XDP Nativo en Proxmox (TL;DR)

```
# En el host de Proxmox:
qm set <VM_ID> -net0 virtio=<MAC>,bridge=vmbr0,queues=8
qm shutdown <VM_ID> && sleep 10 && qm start <VM_ID>

# Dentro de la VM:
ethtool -l eth0  # Verifica 8 colas
sudo nano /etc/omniupf/config.yaml  # Establecer: xdp_attach_mode:
native
sudo systemctl restart omniupf
journalctl -u omniupf --since "1 min ago" | grep xdp  # Verifica
modo nativo
```

Verificar que el Modo XDP Está Activo

```
# Verificar configuración
curl -s http://localhost:8080/api/v1/config | grep xdp_attach_mode

# Verificar estadísticas
curl -s http://localhost:8080/api/v1/xdp_stats | jq

# Verificar colas
ethtool -l eth0
```

Documentación de la API de OmniUPF

Visión general

La API de OmniUPF proporciona una interfaz RESTful completa para gestionar y monitorear la Función de Plano de Usuario basada en eBPF. La API permite el control en tiempo real y la observabilidad de todos los componentes del UPF.

Capacidades de la API

Gestión de Sesiones:

- **Sesiones PFCP**: Consultar sesiones activas, ver detalles de la sesión, filtrar por IP de UE o TEID
- **Asociaciones PFCP**: Monitorear asociaciones y estado de nodos del plano de control

Reglas de Tráfico:

- **Reglas de Detección de Paquetes (PDR)**: Inspeccionar clasificadores de tráfico de enlace ascendente y descendente (IPv4/IPv6)
- **Reglas de Acción de Reenvío (FAR)**: Ver políticas de reenvío, almacenamiento en búfer y descarte
- **Reglas de Aplicación de QoS (QER)**: Monitorear limitación de tasa y políticas de QoS
- **Reglas de Reporte de Uso (URR)**: Rastrear contadores de volumen de datos por sesión

Almacenamiento en Búfer de Paquetes:

- **Estado del Búfer**: Ver paquetes almacenados en búfer por FAR (`GET /buffer`, `GET /buffer/:far_id`)

- **Operaciones de Búfer**: Vaciar o limpiar paquetes almacenados en búfer (`POST /buffer/:far_id/flush`, `DELETE /buffer/:far_id`, `DELETE /buffer`)
- **Control de Almacenamiento en Búfer**: Activación manual de notificaciones (`POST /buffer/:far_id/notify`)
- **Estado de Notificación**: Ver estado de notificación DLDR (`GET /buffer/notifications`)

Monitoreo y Estadísticas:

- **Estadísticas de Paquetes**: Contadores de paquetes en tiempo real por protocolo (GTP, IP, TCP, UDP, ICMP, ARP)
- **Estadísticas de XDP**: Métricas de rendimiento del datapath (pasar, descartar, redirigir, abortar)
- **Estadísticas de Interfaz N3/N6**: Distribución de tráfico de RAN y Red de Datos
- **Estadísticas de Rutas**: Rendimiento de búsqueda de FIB (aciertos de caché, búsquedas, errores)

Gestión de Rutas:

- **Rutas de UE**: Consultar tabla de enrutamiento de IP de UE a gNB (`GET /routes`)
- **Integración de FRR**: Sincronizar rutas con el demonio de Free Range Routing (`POST /routes/sync`)
- **Sesiones de Enrutamiento**: Ver sesiones de protocolo de enrutamiento (`GET /routing/sessions`)
- **Base de Datos OSPF**: Consultar base de datos de rutas externas OSPF (`GET /ospf/database/external`)

Configuración:

- **Configuración de UPF**: Recuperar y editar configuración (`GET /config`, `POST /config`)
- **Configuración de Dataplane**: Consultar configuración específica de dataplane (`GET /dataplane_config`)

- **Capacidades de XDP**: Consultar soporte de modo XDP y capacidades de interfaz (`GET /xdp_capabilities`)
- **Capacidad del Mapa eBPF**: Monitorear utilización de recursos y capacidad (`GET /map_info`)

Integración de la Interfaz Web

La Interfaz Web de OmniUPF está construida sobre esta API y proporciona un panel interactivo para toda la funcionalidad de la API. Consulte la Guía de la Interfaz Web para capturas de pantalla y ejemplos de uso.

Documentación de la API Swagger

La API está completamente documentada utilizando la especificación **OpenAPI 3.0 (Swagger)**. La interfaz interactiva de Swagger proporciona:

- Documentación completa de los puntos finales con esquemas de solicitud/respuesta
- Funcionalidad de prueba para probar llamadas a la API directamente desde el navegador
- Definiciones de esquemas para todos los modelos de datos
- Códigos de estado HTTP y respuestas de error

Interfaz interactiva de Swagger mostrando los puntos finales de la API de OmniUPF con documentación detallada.

Accediendo a Swagger UI

La documentación de Swagger está disponible en:

```
http://<upf-host>:8080/swagger/index.html
```

Por ejemplo: `http://10.98.0.20:8080/swagger/index.html`

Ruta Base de la API

Todos los puntos finales de la API están prefijados con:

```
/api/v1
```

```
## Características de la API

### Paginación

La API de OmniUPF admite paginación para puntos finales que
devuelven grandes conjuntos de datos. La paginación previene
tiempos de espera y reduce el uso de memoria al consultar miles de
sesiones, PDRs o URRs.

**Estilos de Paginación Soportados**:

1. **Paginación basada en páginas** (recomendada):
   - `page`: Número de página (comenzando desde 1)
   - `page_size`: Elementos por página (predeterminado: 100,
máximo: 1000)

2. **Paginación basada en desplazamiento**:
   - `offset`: Número de elementos a omitir
   - `limit`: Número de elementos a devolver (máx: 1000)

**Ejemplos de Solicitudes**:

```bash
# Basada en páginas: Obtener la segunda página con 50 elementos
por página
GET /api/v1/pfcp_sessions?page=2&page_size=50

# Basada en desplazamiento: Omitir los primeros 100 elementos,
devolver los siguientes 50
GET /api/v1/pfcp_sessions?offset=100&limit=50

# Comportamiento predeterminado (sin parámetros de paginación):
Primeros 100 elementos
GET /api/v1/pfcp_sessions
```

Formato de Respuesta:
```

```
&amp;#123;
  "data": [
    &amp;#123; /* objeto de sesión */ &amp;#125;,
    &amp;#123; /* objeto de sesión */ &amp;#125;,
    ...
  ],
  "pagination": &amp;#123;
    "total": 5432,
    "page": 2,
    "page_size": 50,
    "total_pages": 109
  &amp;#125;
&amp;#125;
```

Puntos Finales Paginados:

- `/api/v1/pfcp_sessions` - Lista de sesiones PFCP
- `/api/v1/pfcp_associations` - Lista de asociaciones PFCP
- `/api/v1/routes` - Rutas de IP de UE
- `/api/v1/uplink_pdr_map` - PDRs de enlace ascendente (información básica)
- `/api/v1/uplink_pdr_map/full` - PDRs de enlace ascendente con detalles completos del filtro SDF
- `/api/v1/downlink_pdr_map` - PDRs de enlace descendente IPv4 (información básica)
- `/api/v1/downlink_pdr_map/full` - PDRs de enlace descendente IPv4 con detalles completos del filtro SDF
- `/api/v1/downlink_pdr_map_ip6` - PDRs de enlace descendente IPv6 (información básica)
- `/api/v1/downlink_pdr_map_ip6/full` - PDRs de enlace descendente IPv6 con detalles completos del filtro SDF
- `/api/v1/far_map` - Reglas de Acción de Reenvío
- `/api/v1/qer_map` - Reglas de Aplicación de QoS
- `/api/v1/urr_map` - Reglas de Reporte de Uso

Puntos Finales de Gestión de Búfer:

- `GET /api/v1/buffer` - Listar todos los búferes FAR con estadísticas
- `GET /api/v1/buffer/:far_id` - Obtener estado del búfer para un FAR específico
- `GET /api/v1/buffer/notifications` - Listar estado de notificación DLDR
- `DELETE /api/v1/buffer` - Limpiar todos los paquetes almacenados en búfer
- `DELETE /api/v1/buffer/:far_id` - Limpiar búfer para un FAR específico
- `POST /api/v1/buffer/:far_id/flush` - Vaciar (reproducir) paquetes almacenados en búfer
- `POST /api/v1/buffer/:far_id/notify` - Enviar manualmente notificación DLDR

Puntos Finales de Configuración:

- `GET /api/v1/config` - Obtener configuración actual de UPF
- `POST /api/v1/config` - Actualizar configuración de UPF (campos editables en tiempo de ejecución)
- `GET /api/v1/dataplane_config` - Obtener configuración específica de dataplane

Puntos Finales de Integración de Enrutamiento:

- `GET /api/v1/routes` - Listar rutas de UE
- `POST /api/v1/routes/sync` - Activar sincronización de rutas con FRR
- `GET /api/v1/routing/sessions` - Obtener sesiones de protocolo de enrutamiento
- `GET /api/v1/ospf/database/external` - Obtener base de datos de LSA externa OSPF

Mejores Prácticas:

- Usar `page_size=100` para visualización en la Interfaz Web
- Usar `page_size=1000` para exportaciones masivas (límite máximo)
- Consultar `pagination.total_pages` para determinar el conteo de iteraciones
- Aumentar `page_size` para un mejor rendimiento de la API (menos solicitudes)

Soporte CORS

El intercambio de recursos de origen cruzado (CORS) está habilitado por defecto para todos los puntos finales de la API, permitiendo que la Interfaz Web y aplicaciones de terceros consuman la API desde diferentes orígenes.

Métricas de Prometheus

Además de la API REST, OmniUPF expone métricas de Prometheus en el punto final `/metrics` (puerto predeterminado `:9090`).

Las métricas proporcionan:

- Contadores de mensajes PFCP y latencia por par
- Estadísticas de paquetes por tipo de protocolo
- Veredictos de acción de XDP
- Estadísticas de búfer
- Utilización de capacidad del mapa eBPF
- Seguimiento de volumen de URR

Consulte la Referencia de Métricas para la documentación completa.

Documentación Relacionada

- **Guía de la Interfaz Web** - Panel interactivo construido sobre esta API
- **Referencia de Métricas** - Documentación de métricas de Prometheus
- **Códigos de Causa PFCP** - Códigos de error PFCP y solución de problemas
- **Guía de Gestión de Reglas** - Configuración de PDR, FAR, QER, URR
- **Guía de Gestión de Rutas** - Integración de FRR y enrutamiento de UE
- **Guía de Monitoreo** - Monitoreo de estadísticas y planificación de capacidad
- **Guía de Configuración** - Opciones de configuración de UPF
- **Swagger UI** - Documentación interactiva de la API (reemplazar `localhost` con su host de UPF)

Gestión de Rutas UE

> **Documentación Relacionada**:
>
> - Documentación de la API - Referencia completa de la API incluyendo puntos finales de gestión de rutas
> - Guía de Operaciones - Operaciones y monitoreo de la interfaz web

Visión General

El UPF (Función de Plano de Usuario) se integra con **FRR (Free Range Routing)** para gestionar dinámicamente las rutas IP del Equipo de Usuario (UE). Esta integración asegura que, a medida que se establecen o terminan sesiones de UE, la infraestructura de enrutamiento se adapte automáticamente para reflejar la topología actual de la red.

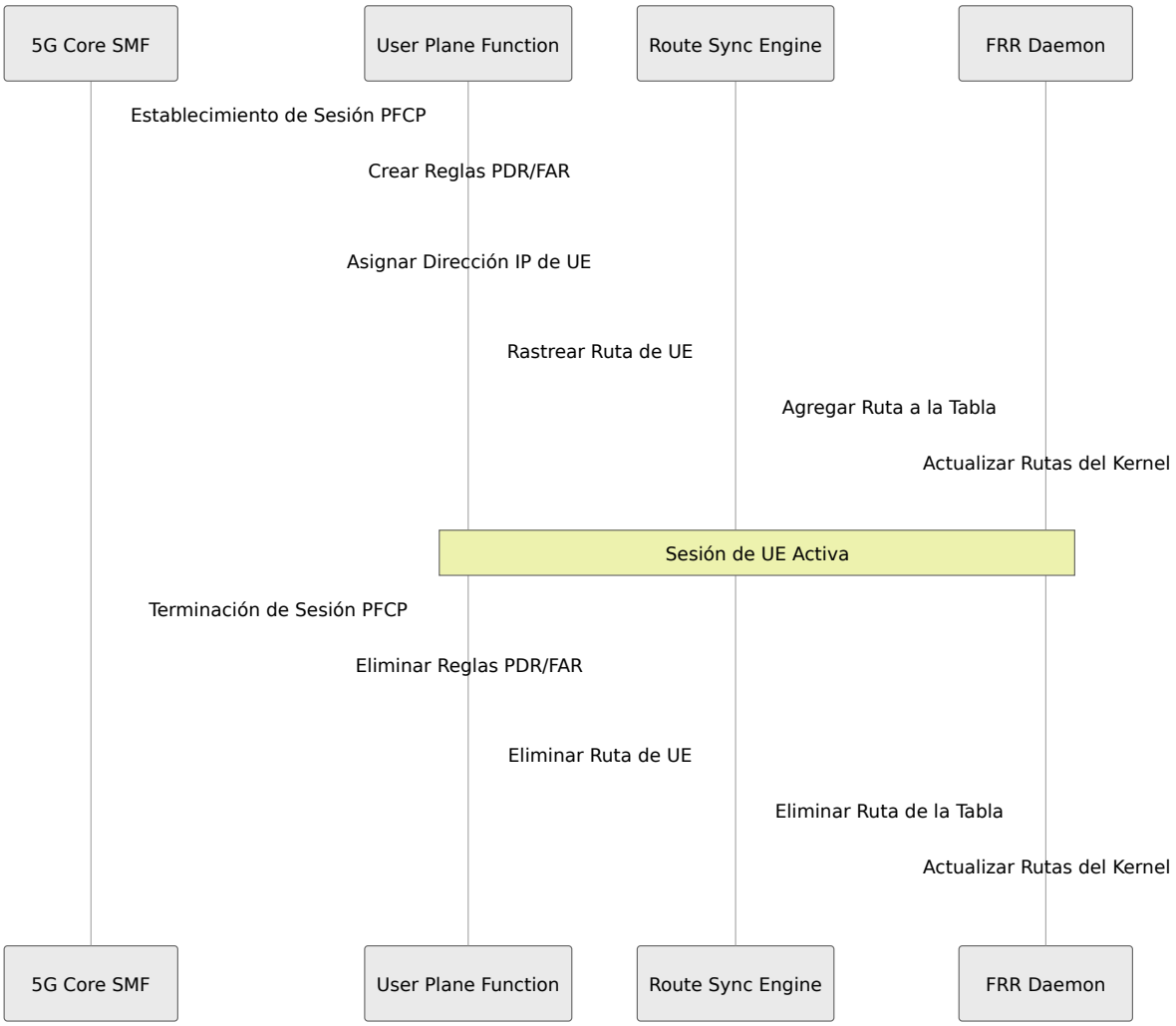
¿Qué es FRR?

FRR (Free Range Routing) es un robusto conjunto de protocolos de enrutamiento de código abierto para plataformas Linux y Unix. Implementa varios protocolos de enrutamiento, incluyendo BGP, OSPF, RIP, y otros. En nuestra implementación, FRR actúa como el demonio de enrutamiento que mantiene la tabla de enrutamiento del kernel y puede redistribuir rutas a otros elementos de la red.

Arquitectura

Cómo Funciona la Sincronización de Rutas

Ciclo de Vida de la Ruta



Sincronización Automática

El UPF mantiene un registro interno de todas las direcciones IP de UE activas. Cuando está habilitado, el sistema de sincronización de rutas:

1. **Monitorea Sesiones de UE**: Rastrea todas las sesiones PFCP activas y sus direcciones IP de UE asociadas
2. **Mantiene Lista de Rutas**: Mantiene una lista actualizada de rutas que deben estar en la tabla de enrutamiento

3. **Sincroniza con FRR**: Envía automáticamente actualizaciones de rutas al demonio FRR a través de su API
4. **Maneja Fallos**: Rastrea el estado de sincronización (sincronizado/fallido) para cada ruta y reintenta según sea necesario

Configuración de FRR

Configuración

FRR se despliega y configura utilizando **plantillas de Ansible** para establecer los parámetros básicos de enrutamiento. Definas la configuración de FRR una vez como una **plantilla Jinja2** en tu libro de jugadas de Ansible, y Ansible la propaga automáticamente a todas tus instancias de UPF durante el despliegue.

Una plantilla típica de configuración Jinja2 para FRR incluye:

```
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
```

Modelo de Despliegue:

1. **Definir Una Vez**: Crea la plantilla Jinja2 de FRR en tu rol de Ansible (por ejemplo, `roles/frr/templates/frr.conf.j2`)
2. **Configurar Parámetros**: Establece variables en tu inventario de Ansible para cada host de UPF
3. **Desplegar en Todas Partes**: Ejecuta el libro de jugadas de Ansible para desplegar la configuración de FRR a todos los nodos de UPF
4. **Personalización Automática**: Ansible utiliza variables específicas del host (direcciones IP, IDs de router, etc.) para personalizar la configuración de FRR de cada UPF

Parámetros Personalizables en la plantilla Jinja2:

- **Parámetros OSPF**: ID de router, configuración de área, métodos de autenticación, anuncios de red
- **Configuración BGP**: ASN, relaciones de vecinos, políticas de ruta, comunidades
- **Redistribución de Rutas**: Qué rutas redistribuir (por ejemplo, `redistribute static` para rutas de UE)
- **Filtrado de Rutas**: Mapas de ruta, listas de prefijos, listas de acceso
- **Configuraciones de Interfaz**: Parámetros de interfaz OSPF/BGP

Integración UPF: Una vez que la configuración base de FRR se despliega en cada instancia de UPF, el UPF agrega dinámicamente direcciones IP de UE como **rutas de host** (/32 para IPv4, /128 para IPv6) a través de la interfaz vtysh de FRR basada en sesiones PFCP activas. Estas rutas son luego:

1. **Agregadas como rutas estáticas de FRR** por el motor de sincronización de rutas de UPF (a través de vtysh)
2. **Recogidas por FRR** a través de la directiva `redistribute static`
3. **Anunciadas a protocolos de enrutamiento** (OSPF, BGP) de acuerdo con tu configuración de FRR
4. **Propagadas a la red** para que el tráfico de UE pueda ser enrutado a esta instancia de UPF

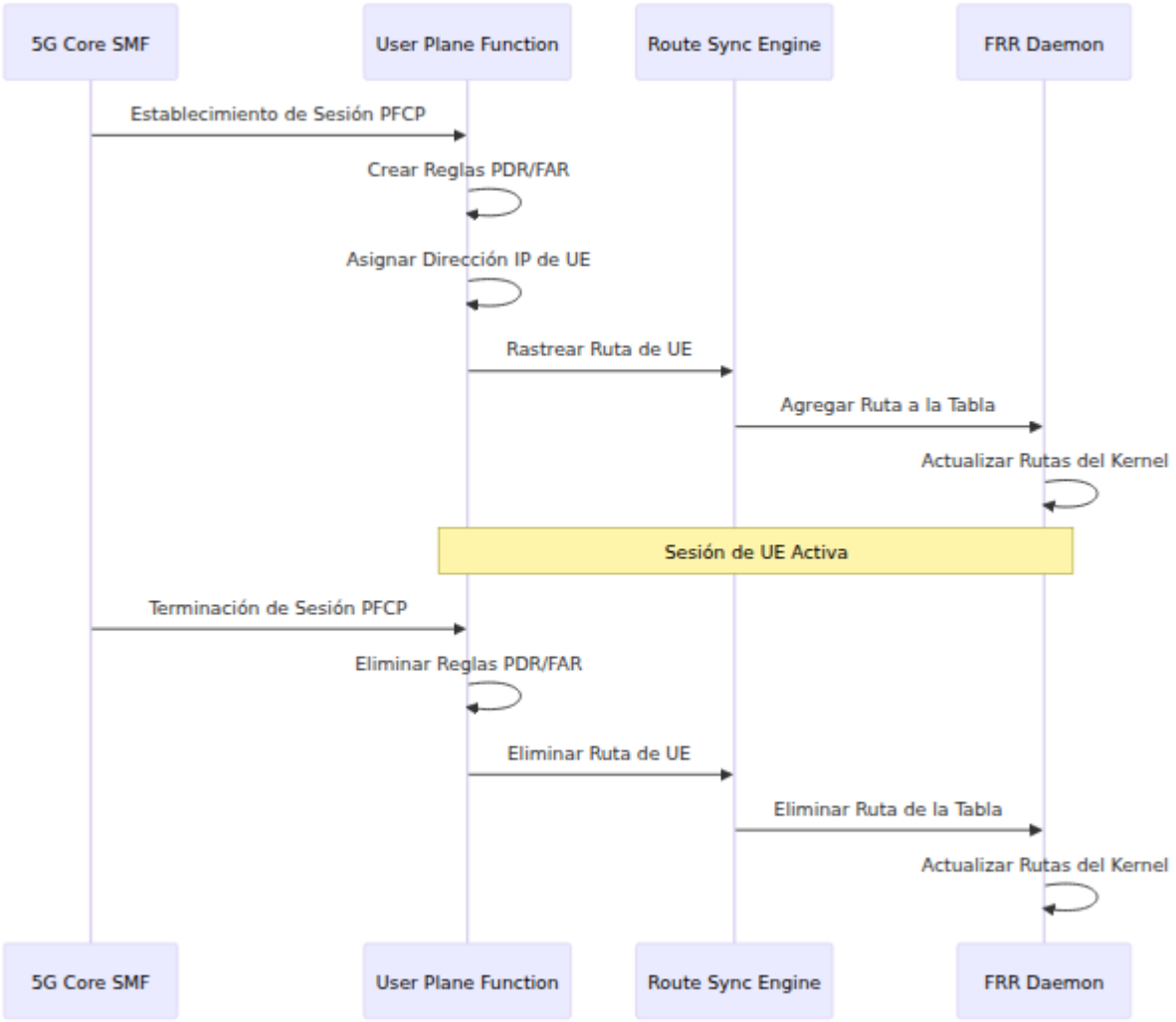
> **Importante**: El UPF agrega rutas a través de la interfaz vtysh de FRR, convirtiéndolas en rutas estáticas de FRR (no rutas del kernel). Debes usar `redistribute static` en tu configuración de OSPF/BGP, no `redistribute kernel`.

Puntos Clave:

- **Definir Una Vez, Desplegar en Todas Partes**: Define la plantilla Jinja2 de FRR una vez en Ansible, y se despliega automáticamente a todas las instancias de UPF
- **Ansible maneja la configuración estática**: La plantilla Jinja2 configura todos los parámetros de los protocolos de enrutamiento (áreas OSPF, vecinos BGP, autenticación, políticas de ruta, etc.)
- **UPF maneja rutas dinámicas**: Cada instancia de UPF gestiona dinámicamente solo las rutas IP de UE /32 basadas en sus sesiones PFCP activas

- **Anuncio automático de rutas**: FRR en cada UPF redistribuye automáticamente las rutas locales de UE de acuerdo con tus políticas configuradas
- **Gestión centralizada**: Actualiza la plantilla de Ansible y vuelve a ejecutar el libro de jugadas para cambiar la configuración de enrutamiento en todos los UPFs simultáneamente

Anuncio de Rutas



Monitoreo y Gestión

Integración de la Interfaz Web

El Panel de Control de UPF proporciona una página de **Rutas** que muestra:

- **Estado de Ruta**: Si la sincronización de rutas está habilitada o deshabilitada
- **Total de Rutas**: Número de direcciones IP de UE que se están rastreando
- **Estadísticas de Sincronización**: Conteo de rutas sincronizadas con éxito y cualquier fallo
- **Rutas Activas**: Lista en tiempo real de todas las direcciones IP de UE actualmente en la tabla de enrutamiento
- **Vecinos OSPF**: Estado en vivo de las adyacencias OSPF con detalles de vecinos
- **Vecinos BGP**: Estado de sesión BGP y estadísticas de prefijos (cuando están configurados)
- **Rutas Redistribuidas OSPF**: Vista completa de LSAs externas que muestran cómo se anuncian las rutas de UE

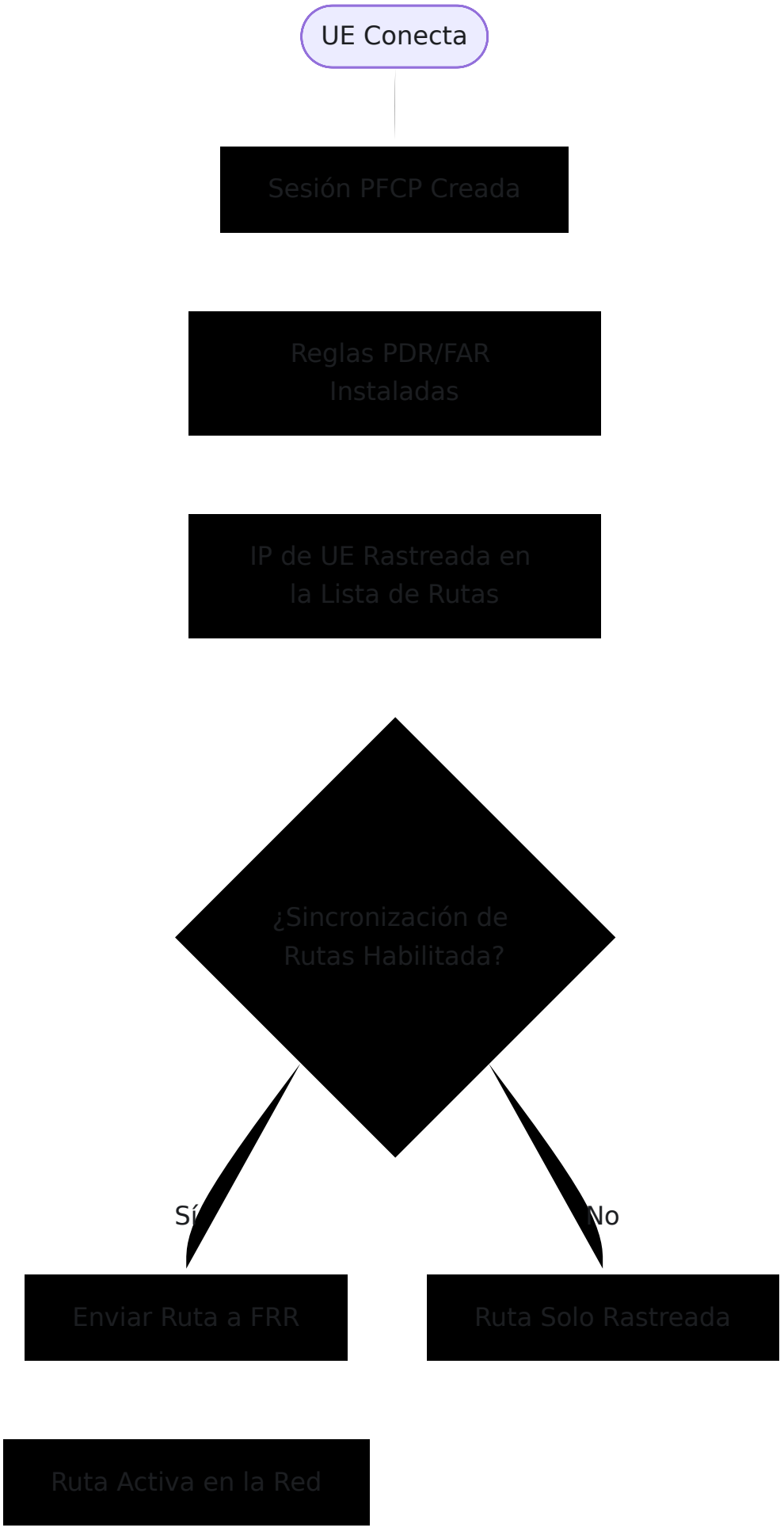
La página de Rutas proporciona una visibilidad completa sobre la sincronización de rutas de UE, vecinos de protocolos de enrutamiento y anuncios de rutas redistribuidas.

Operación de Sincronización Manual

Los administradores pueden activar una sincronización manual de rutas a través de la interfaz web utilizando el botón **Sincronizar Rutas**. Esta operación:

1. Vuelve a leer la lista actual de sesiones de UE activas desde el UPF
2. Compara con la tabla de enrutamiento de FRR
3. Agrega cualquier ruta faltante
4. Elimina cualquier ruta obsoleta
5. Devuelve estadísticas de sincronización actualizadas

Flujo de Rutas


UE Conecta
Sesión PFCP Creada
Reglas PDR/FAR Instaladas
IP de UE Rastreada en la Lista de Rutas
¿Sincronización de Rutas Habilitada?
Sí
No
Enviar Ruta a FRR
Ruta Solo Rastreada
Ruta Activa en la Red

Beneficios

- **Provisionamiento Sin Toque**: Las rutas se gestionan automáticamente sin intervención manual
- **Adaptación Dinámica**: El enrutamiento de la red se adapta en tiempo real a la movilidad de UE y cambios en las sesiones
- **Escalabilidad**: Soporta miles de rutas de UE concurrentes
- **Resiliencia**: Las operaciones de sincronización fallidas se rastrean y pueden reintentarse
- **Visibilidad**: Visibilidad completa del estado de las rutas a través de la interfaz web

Detalles Técnicos

Puntos Finales de la API

El UPF expone los siguientes puntos finales de gestión de rutas:

- `GET /api/v1/routes` - Listar todas las rutas de UE rastreadas sin sincronización
- `POST /api/v1/routes/sync` - Sincronizar rutas con FRR y devolver lista actualizada
- `GET /api/v1/route_stats` - Obtener estadísticas detalladas de enrutamiento
- `GET /api/v1/routing/sessions` - Obtener sesiones de protocolo de enrutamiento (vecinos OSPF, pares BGP)
- `GET /api/v1/ospf/database/external` - Obtener base de datos LSA externa OSPF (rutas redistribuidas)

> **Ver También**: Documentación de la API - Gestión de Rutas para detalles completos de los puntos finales y ejemplos

Formato de Ruta

Las rutas se almacenan y gestionan como simples direcciones IP (por ejemplo, `100.64.18.5`). El demonio de enrutamiento maneja todos los detalles de la entrada de ruta, incluyendo:

- Prefijo/máscara de destino
- Puerta de enlace/siguiente salto
- Vinculación de interfaz
- Métrica y distancia administrativa

Soporte IPv6

El gestor de rutas soporta direcciones de UE tanto IPv4 como IPv6:

Tipo de Dirección	Longitud de Prefijo	Ejemplo
IPv4	/32	`100.64.18.5/32`
IPv6	/128	`2001:db8::1/128`

Para IPv6, asegúrate de que tu configuración de FRR incluya la redistribución OSPFv3 o BGP IPv6 apropiada:

```
router ospf6
 redistribute static
```

o para BGP:

```
router bgp <asn>
 address-family ipv6 unicast
  redistribute static
```

Verificación de FRR

Base de Datos LSA Externa OSPF

Puedes verificar que las rutas de UE se están redistribuyendo correctamente en OSPF examinando la Base de Datos de Estado de Enlace OSPF de FRR. Las LSAs

externas (Tipo 5) muestran rutas que han sido inyectadas en OSPF desde fuentes externas.

Base de datos OSPF de FRR mostrando LSAs externas incluyendo la ruta de UE 100.64.18.5/32 siendo anunciada como una ruta E2 (Tipo Externo 2).

En el ejemplo anterior, puedes ver:

- **LSA de Red (10.98.0.20)**: El anuncio de red propio del UPF
- **LSA de Router (192.168.1.1)**: Anuncio de router OSPF
- **LSAs Externas**: Incluyendo la ruta de UE `100.64.18.5` redistribuida en OSPF con tipo de métrica E2 (Tipo Externo 2)

Esta verificación confirma que:

1. El UPF está rastreando correctamente la dirección IP de UE
2. El motor de sincronización de rutas ha enviado la ruta a FRR
3. FRR ha redistribuido la ruta en OSPF
4. Los vecinos OSPF están recibiendo los anuncios de ruta