OmniUPF Guía de Operaciones

Tabla de Contenidos

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 grado 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
- **Informes de uso** para facturación y análisis
- **Almacenamiento en búfer de paquetes** para escenarios de movilidad y gestión de sesiones
- Soporte para **intercepción legal** para cumplimiento regulatorio

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 totalmente conforme a 3GPP
- Ruta de datos basada en eBPF para un 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 Informes 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 diagnósticos
- 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 multicores
- Capaz de descarga para aceleración de hardware
- Optimizado para implementaciones nativas en la nube

Para un uso detallado del panel de control, consulte Operaciones de la Interfaz Web.

Entendiendo la Arquitectura del Plano de Usuario

OmniUPF es una solución unificada de plano de usuario que proporciona reenvío de paquetes de grado de operador para redes 5G Autónomas (SA), 5G NSA y 4G

LTE/EPC. **OmniUPF es un único 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 Datos del PDN)** - puerta de enlace EPC 4G a redes externas (controlado por OmniPGW-C a través de Sxc/PFCP)
- **SGW-U (Puerta de Enlace de Servicio del 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 consistente ya sea funcionando 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 tanto como SGW-U como PGW-U, controlado por funciones de plano de control separadas.

Modo Separado SGW-U y PGW-U (Roaming/Múltiples Sitios)

En implementaciones de roaming o múltiples sitios, 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 SGWU como 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 de CPU del 40-50%** - 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:

```
# OmniUPF config.yml
interface_name: [eth0]
n3_address: "10.0.1.10"      # IP de la interfaz S1-U
n9_address: "10.0.1.10"      # La misma IP habilita el bucle N9
pfcp_address: ":8805"         # Tanto SGWU-C como PGWU-C se conectan aquí
```

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:

- Redundancia geográfica requerida (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 interfaz S5/S8 (PGW-U) desde eNodeB con encapsulación GTP-U
    - El plano de usuario coincide los paquetes con los PDR 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 coincide los paquetes con los PDR 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 en búfer durante el cambio de ruta
    - Transición sin interrupciones 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 5G y 4G a través de interfaces estándar 3GPP:

Interfaces 5G

Interfaz	De → A	Propósito	Especificación 3GPP
N4	OmniSMF ↔ OmniUPF	Establecimiento, modificación, eliminación de sesiones 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 a 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 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 a 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.

Para la gestión de sesiones PFCP, consulte Operaciones PFCP.

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**: Coincidencia de paquetes contra 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 del núcleo
- **Soporte XDP**: Adjuntar a nivel del controlador de red para latencia sub-microsegundo
- **Multicore**: Escala a través de núcleos de CPU con soporte de mapa por

CPU
- **Capacidad**: Millones de PDRs/FARs en mapas eBPF (limitado 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
- **Informes 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)

Para operaciones PFCP detalladas, consulte **Guía de Operaciones PFCP**.

Servidor API REST

La **API REST** proporciona acceso programático al estado y operaciones de 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 Búfer**: Ver y controlar búferes de paquetes
- **Información de Mapas**: Monitorear uso y capacidad de mapas eBPF

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

Categoría	Puntos de Acceso	Descripción
Salud	`/health`	Verificación de salud y estado
Configuración	`/config`	Configuración de UPF
Sesiones	`/pfcp_sessions`, `/pfcp_associations`	Datos de sesión/ asociación 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 informes de uso
Búferes	`/buffer`	Estado y control del búfer de paquetes
Estadísticas	`/packet_stats`, `/route_stats`, `/xdp_stats`, `/n3n6_stats`	Métricas de rendimiento
Capacidad	`/map_info`	Capacidad y uso de mapas eBPF
Ruta de Datos	`/dataplane_config`	Direcciones de interfaz N3/N9

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

Panel de Control Web

El **Panel de Control Web** proporciona un tablero en tiempo real para el monitoreo y gestión de 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 Búferes**: 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 de UPF y direcciones de ruta de datos
- **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.

Creación de Sesiones 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**: Coincidencia en TEID N3, reenvío a través de FAR a N6
- **PDR de Bajada**: Coincidencia en dirección IP de UE, reenvío 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**: Informes de volumen para facturación (opcional)

Para el monitoreo de sesiones, consulte Operaciones PFCP.

Modificación de Sesiones 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 el nuevo punto final de túnel gNB (F-TEID)
    - Opcionalmente almacenar paquetes en búfer 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 Sesiones 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
- Búferes de paquetes limpiados
- Informe final de uso enviado a 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 locales y remotos (Identificadores de Punto de Fin 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 la sesión** incluyendo configuraciones completas de PDR/FAR/QER/URR
- **Monitorear conteos de sesiones** por asociación PFCP

Para procedimientos detallados de sesión, consulte Vista de Sesiones.

Gestión de Reglas

Reglas de Detección de Paquetes (PDR):

Los 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):

Los 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):

Los 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 activos** por sesión
- **Inspeccionar marcas QFI** para flujos de QoS 5G

Reglas de Informes de Uso (URR):

Los 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 activos** 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 conexiones caídas, descargas interrumpidas y fallos en comunicaciones en tiempo real cada vez que se mueven entre torres de celdas 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 de celda 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 bloquean** o reinician (navegación web, descargas interrumpidas)
- **Videollamadas que se congelan** o 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 caen por completo (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 de celdas:

Línea de Tiempo:

- **T+0ms**: Ruta antigua aún activa
- **T+10ms**: SMF le dice a UPF que almacene en búfer (ruta antigua cerrándose, 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 empresarial entra en el campus corporativo (cambios en la dirección del 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)
- Cambios en 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 según las banderas de acción de 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**: Búferes en memoria por FAR con metadatos (marca de tiempo, dirección, tamaño de 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 son descartados
- **Limpieza**: Proceso en segundo plano elimina paquetes expirados cada 60 segundos

Ciclo de Vida del Búfer:

1. **Almacenamiento en Búfer Activado**: SMF establece la acción FAR BUFF=1 (bit 2) a través de la 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 búferes almacena paquetes con ID FAR, marca de tiempo, dirección
4. **Almacenamiento en Búfer Desactivado**: SMF establece la 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 caer	Sin interrupciones, sin problemas
Descarga de Archivo en	La descarga falla, debe	La descarga continúa

Escenario	Sin Almacenamiento en Búfer	Con Almacenamiento en Búfer
el Límite de la Celda	reiniciarse	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	Cristalina, sin caídas
Streaming de Video en el Tren	El video se almacena en búfer, la calidad cae	Reproducción fluida
Hotspot Móvil para Laptop	La sesión SSH se cae, la videollamada falla	Todas las conexiones mantenidas

Beneficios para el Operador de Red:

- **Tasa de Caída de Llamadas Reducida (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 de 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 de 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 de búfer por FAR o global

Resolución de Problemas:

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

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

Configuración del Búfer

Configure el comportamiento del almacenamiento en búfer en `config.yml`:

```yaml
# Configuración del búfer
buffer_port: 22152                # Puerto UDP para paquetes almacenados en búfer (por defecto)
buffer_max_packets: 10000         # Máximo de paquetes por FAR (evitar agotamiento de memoria)
buffer_max_total: 100000          # Máximo total de paquetes en todos los FARs
buffer_packet_ttl: 30             # TTL en segundos (descartar paquetes antiguos)
buffer_cleanup_interval: 60       # Intervalo de limpieza en segundos
```

Recomendaciones:

- **Redes de alta movilidad** (autopistas, trenes): Aumentar `buffer_max_packets` a 20,000+
- **Áreas urbanas densas** (transferencias frecuentes): Disminuir `buffer_packet_ttl` a 15s
- **Aplicaciones de baja latencia**: Establecer `buffer_packet_ttl` a 10s para evitar datos obsoletos
- **Redes IoT**: Disminuir 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 desde 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 Ruta:

Métricas de reenvío por ruta:

- **Coincidencias de ruta**: Paquetes coincidentes por cada ruta
- **Conteos de reenvío**: Éxito/fallo por destino
- **Contadores de errores**: TEIDs inválidos, IPs de UE desconocidas

Estadísticas de XDP:

Métricas de rendimiento del 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 de UPF (límite de memoria del núcleo)
- Exceder la capacidad causa fallos en el establecimiento de sesiones

Para el monitoreo de capacidad, consulte Gestión de Capacidad.

Gestión de Configuración

Configuración de UPF:

Ver y verificar parámetros operativos de 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 la API REST
- **Punto de Métricas**: Puerto de métricas de Prometheus

Configuración del Plano de Datos:

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**
    - Verificar que el SMF pueda alcanzar la interfaz PFCP del UPF (puerto 8805)
    - Comprobar el estado de la asociación PFCP en la vista de Sesiones
    - Verificar que la configuración del ID de Nodo coincida entre SMF y UPF
2. **Capacidad del Mapa eBPF Agotada**
    - Comprobar la vista de Capacidad para el uso de mapas en rojo (>90%)
    - Aumentar los tamaños de mapa eBPF en la configuración de UPF
    - Eliminar sesiones obsoletas si el mapa está lleno
3. **Configuración Inválida de PDR/FAR**
    - Verificar que la dirección IP de UE sea única y válida
    - Comprobar que la asignación de TEID no tenga conflictos
    - Asegurarse de que el FAR haga referencia a instancias de red válidas
4. **Problemas de Configuración de Interfaz**
    - Verificar que la IP de la interfaz N3 sea accesible desde gNB
    - Comprobar tablas de enrutamiento para conectividad N6 a la red de datos
    - Confirmar que el tráfico GTP-U no esté bloqueado por un firewall

Para la resolución de problemas detallada, consulte Guía de Resolución de Problemas.

Pérdida de Paquetes o Problemas de Reenvío

Síntomas: El UE tiene conectividad pero experimenta pérdida de paquetes o no hay flujo de tráfico

Causas Raíz Comunes:

1. **Configuración Incorrecta de PDR**
    - Verificar que el PDR de subida TEID coincida con el TEID asignado por gNB
    - Comprobar que el PDR de bajada IP de UE coincida con la IP asignada
    - Inspeccionar filtros SDF para reglas demasiado restrictivas

2. **Problemas de Acción de FAR**
    - Verificar que la acción FAR sea REENVIAR (no DESCARTAR o ALMACENAR EN BÚFER)
    - Comprobar parámetros de creación de encabezado externo para GTP-U
    - Asegurarse de que el punto final de destino sea correcto
3. **Límites de QoS Excedidos**
    - Comprobar configuraciones de QER MBR (Tasa de Bit Máxima)
    - Verificar la asignación de GBR (Tasa de Bit Garantizada)
    - Monitorear pérdidas de paquetes debido a limitación de tasa
4. **Problemas de MTU de Interfaz**
    - Verificar que la sobrecarga de GTP-U (40-50 bytes) no cause fragmentación
    - Comprobar la configuración de MTU de las interfaces N3/N6
    - Monitorear mensajes ICMP de fragmentación necesaria

Problemas Relacionados con el Búfer

Síntomas: Paquetes almacenados en búfer indefinidamente, desbordamiento de búfer

Causas Raíz Comunes:

1. **Almacenamiento en Búfer No Desactivado Después de la Transferencia**
    - Comprobar la bandera de almacenamiento en búfer de FAR (bit 2)
    - Verificar que SMF envió la Modificación de Sesión para desactivar el almacenamiento en búfer
    - Desactivar manualmente el almacenamiento en búfer a través del panel de control si está atascado
2. **Expiración de TTL del Búfer**
    - Comprobar la edad del paquete en la vista de búfer
    - Verificar la configuración de TTL del búfer (el valor por defecto puede ser demasiado largo)
    - Limpiar búferes expirados manualmente
3. **Capacidad del Búfer Agotada**
    - Monitorear el uso total del búfer y límites por FAR
    - Comprobar reglas mal configuradas que causan almacenamiento en

búfer excesivo
    - Ajustar límites de max_per_far y max_total de búfer

Para la resolución de problemas de búfer, consulte Operaciones de Búfer.

Anomalías en las Estadísticas

Síntomas: Contadores de paquetes inesperados, estadísticas faltantes

Causas Raíz Comunes:

1. **Desbordamiento de Contadores**
    - Los mapas eBPF utilizan contadores de 64 bits (no deberían desbordarse)
    - Comprobar eventos de reinicio de contadores en los registros
    - Verificar que el informe de URR esté funcionando
2. **Estadísticas de Ruta No Actualizándose**
    - Verificar que el programa eBPF esté adjunto a las interfaces
    - Comprobar que la versión del núcleo soporte las características eBPF requeridas
    - Revisar estadísticas de XDP para errores de procesamiento
3. **Desajuste de Estadísticas de Interfaz**
    - Comparar estadísticas de N3/N6 con contadores de interfaz del núcleo
    - Comprobar tráfico que elude eBPF (por ejemplo, enrutamiento local)
    - Verificar que todo el tráfico fluya a través de ganchos XDP

Degradación del Rendimiento

Síntomas: Alta latencia, bajo rendimiento, saturación de CPU

Diagnóstico:

1. **Monitorear Estadísticas de XDP**: Comprobar si hay descartes o abortos de XDP
2. **Comprobar Tiempo de Acceso a Mapas eBPF**: Las búsquedas hash deben ser sub-microsegundo
3. **Revisar Utilización de CPU**: eBPF debe distribuirse entre núcleos
4. **Analizar Interfaz de Red**: Verificar que el NIC soporte descarga de XDP

Consideraciones de Escalabilidad:

- **Rendimiento de XDP**: 10M+ paquetes por segundo por núcleo
- **Capacidad de PDR**: Millones de PDRs limitados solo por la memoria del núcleo
- **Conteo de Sesiones**: Miles de sesiones concurrentes por instancia de UPF
- **Rendimiento**: Rendimiento de múltiples gigabits con un NIC adecuado

Para la optimización del rendimiento, consulte Guía de Arquitectura.

Documentación Adicional

Guías de Operaciones Específicas de Componentes

Para operaciones y resolución de problemas detalladas para cada componente de UPF:

Guía de Configuración

Referencia completa de configuración que incluye:

- Parámetros de configuración (YAML, variables de entorno, CLI)
- Modos de operación (UPF/PGW-U/SGW-U)
- Visión general de modos de adjunto XDP
- Compatibilidad con hipervisor (Proxmox, VMware, KVM, Hyper-V, VirtualBox)
- Compatibilidad de NIC y soporte de controlador XDP
- Ejemplos de configuración para diferentes escenarios
- Dimensionamiento de mapas y planificación de capacidad

Guía de Modos XDP

Configuración y optimización detalladas de XDP que incluyen:

- Modos de adjunto XDP explicados (genérico/nativo/descarga)
- Comparación de rendimiento y benchmarks
- Configuración nativa de XDP en Proxmox VE paso a paso
- Configuración de múltiples colas para un rendimiento óptimo
- Configuración de XDP en VMware ESXi, KVM y Hyper-V
- Verificación y resolución de problemas de XDP
- Selección de hardware para rendimiento de XDP

Guía de Arquitectura

Profundización técnica que incluye:

- Fundamentos de la tecnología eBPF y ciclo de vida del programa
- Pipeline de procesamiento de paquetes XDP con llamadas de cola
- Implementación del protocolo PFCP

- Arquitectura de almacenamiento en búfer (encapsulación GTP-U al puerto 22152)
- Limitación de tasa de ventana deslizante de QoS (ventana de 5ms)
- Características de rendimiento (latencia de 3.5µs, 10 Mpps/núcleo)

Guía de Gestión de Reglas

Referencia de reglas PFCP que incluye:

- Reglas de Detección de Paquetes (PDR) - Clasificación de tráfico
- Reglas de Acción de Reenvío (FAR) - Decisiones de enrutamiento con banderas de acción
- Reglas de Aplicación de QoS (QER) - Gestión de ancho de banda (MBR/GBR)
- Reglas de Informes de Uso (URR) - Seguimiento y reporte de volumen
- Diagramas de flujo de paquetes de subida y bajada
- Lógica de procesamiento de reglas y precedencia

Guía de Monitoreo

Estadísticas y gestión de capacidad que incluyen:

- Estadísticas de interfaces N3/N6 y distribución de tráfico
- Estadísticas de procesamiento de XDP (pasar/descartar/redirigir/abortar)
- Monitoreo de capacidad de mapas eBPF con zonas codificadas por colores
- Métricas de rendimiento (tasa de paquetes, rendimiento, tasa de caída)
- Fórmulas de planificación de capacidad y estimación de sesiones
- Umbrales de alerta y mejores prácticas

Guía de Interfaz Web

Uso del panel de control que incluye:

- Visión general del tablero y navegación
- Monitoreo de sesiones (estados saludables/no saludables)
- Inspección de reglas (detalles de PDR, FAR, QER, URR)
- Monitoreo de búfer y estado de almacenamiento en búfer
- Tablero de estadísticas en tiempo real
- Visualización de capacidad de mapas eBPF
- Visualización de configuración

Documentación API

Referencia completa de la API REST que incluye:

- Documentación interactiva OpenAPI/Swagger
- Puntos de acceso de sesiones y asociaciones PFCP

- Reglas de Detección de Paquetes (PDR) - IPv4 e IPv6
- Reglas de Acción de Reenvío (FAR)
- Reglas de Aplicación de QoS (QER)
- Reglas de Informes de Uso (URR)
- Gestión de búferes de paquetes
- Estadísticas y puntos de monitoreo
- Gestión de rutas y integración FRR
- Información de mapas eBPF
- Gestión de configuración
- Directrices de autenticación y seguridad
- Flujos de trabajo y ejemplos comunes de API

Guía de Gestión de Rutas de UE

Integración de enrutamiento FRR que incluye:

- Visión general y arquitectura de FRR (Free Range Routing)
- Ciclo de vida de sincronización de rutas de UE
- Sincronización automática de rutas al demonio de enrutamiento
- Anuncio de rutas a través de OSPF y BGP
- Monitoreo de vecinos OSPF
- Verificación de base de datos LSA externa de OSPF
- Gestión de sesiones de pares BGP
- Interfaz de monitoreo de rutas en la Interfaz Web
- Operaciones de sincronización de rutas manuales
- Diagramas de Mermaid para flujo de rutas y arquitectura

Guía de Resolución de Problemas

Diagnóstico completo de problemas que incluye:

- Lista de verificación y herramientas de diagnóstico rápidas
- Problemas de instalación y configuración
- Fallos de asociación PFCP
- Problemas de procesamiento de paquetes
- Errores de XDP y eBPF
- Degradación del rendimiento
- Problemas específicos de hipervisor (Proxmox, VMware, VirtualBox)
- Problemas de NIC y controlador
- Procedimientos de resolución paso a paso

Documentación por Caso de Uso

Instalación y Configuración de OmniUPF

1. Comience con esta guía para la descripción general

2. Guía de Configuración para parámetros de configuración
3. Guía de Interfaz Web para acceder al panel de control

Despliegue SGWU+PGWU en Instancia Única (Bucle N9)

1. Guía de Operaciones de Bucle N9 - Guía completa para el despliegue combinado SGWU+PGWU
2. Bucle N9 - Configuración - Configuración de red y PFCP
3. Bucle N9 - Monitoreo - Verificar que el bucle esté activo
4. Bucle N9 - Resolución de Problemas - Problemas comunes y soluciones

Despliegue en Proxmox

1. Guía de Modos XDP - Configuración Nativa de XDP en Proxmox - **Comience aquí para rendimiento**
2. Guía de Configuración - Compatibilidad con Hipervisor
3. Guía de Configuración - Configuración de SR-IOV en Proxmox
4. Resolución de Problemas - Problemas en Proxmox

Optimización del Rendimiento

1. Guía de Modos XDP - **Habilitar XDP nativo para un aumento de rendimiento de 5-10x**
2. Guía de Arquitectura - Optimización del Rendimiento
3. Guía de Configuración - Modos XDP
4. Guía de Monitoreo - Métricas de Rendimiento
5. [Resolución de

Guía de Arquitectura de OmniUPF

Tabla de Contenidos

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. Al ejecutar la lógica de procesamiento de paquetes directamente en el núcleo de Linux, OmniUPF elimina la sobrecarga del procesamiento en espacio de usuario y logra un rendimiento de múltiples gigabits con latencia de microsegundos.

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 de SmartNIC
- Compatible con tarjetas de red que soportan XDP

- Reversa a modos nativos del 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 el espacio del 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 bloquear el núcleo
- **Rendimiento**: Se ejecuta a 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 de PDR por TEID o IP de UE
`BPF_MAP_TYPE_ARRAY`	Array indexado por entero	Búsqueda de QER, FAR, URR por ID
`BPF_MAP_TYPE_PERCPU_HASH`	Tabla hash por CPU (sin bloqueo)	Búsquedas de PDR de alto rendimiento
`BPF_MAP_TYPE_LRU_HASH`	Hash LRU (Menos Recientemente Usado)	Desalojo automático de entradas antiguas

Operaciones de Mapa:

- **Búsqueda**: O(1) búsqueda hash (sub-microsegundo)
- **Actualizar**: Actualizaciones atómicas desde espacio de usuario
- **Eliminar**: 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 Adjunto XDP

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

1. Modo de Descarga XDP

Ejecución de 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 más de 100 Gbps
- Requiere SmartNIC compatible (Netronome, Mellanox ConnectX-6)

Configuración:

```
xdp_attach_mode: offload
```

Limitaciones:

- Requiere hardware de SmartNIC costoso
- Complejidad limitada del programa eBPF
- No todas las características de 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 para XDP (la mayoría de los controladores modernos)

Configuración:

```
xdp_attach_mode: native
```

Ventajas:

- Rendimiento muy alto (millones de pps)
- Amplia compatibilidad de hardware
- Conjunto completo de características de 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 de XDP
- Funciona en cualquier interfaz de red
- Útil para pruebas y desarrollo

Configuración:

```
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 de 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	Paquetes inválidos, limitación de

Código de Retorno	Significado	Uso en OmniUPF
	inmediatamente	tasa, eliminación por políticas
`XDP_TX`	Transmitir paquete de vuelta por la misma interfaz	No se utiliza actualmente
`XDP_REDIRECT`	Enviar paquete a una interfaz diferente	Ruta de reenvío principal (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 de Mapas

Dimensionamiento de Mapas

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 traspaso encapsulando paquetes en GTP-U y enviándolos a un proceso de espacio de usuario a través de un socket UDP.

Arquitectura de Buffering

Detalles de Encapsulación de 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 el listener
   de espacio de usuario se vincula
   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 manejador de buffer de espacio de usuario recibe y almacena el paquete

Operación de Vaciamiento de Buffer

Cuando se completa el traspaso, el SMF actualiza el FAR para eliminar la bandera BUFFER. Los paquetes en buffer se reproducen:

Parámetros de Gestión de Buffer

Parámetro	Predeterminado	Descripción
Máx Por FAR	10,000 paquetes	Máximo de paquetes en buffer por FAR
Máx Total	100,000 paquetes	Máximo total de paquetes en buffer
TTL de Paquete	30 segundos	Tiempo antes de que los paquetes en buffer expiren
Puerto de Buffer	22152	Puerto UDP para entrega de buffer
Intervalo de Limpieza de Buffer	60 segundos	Con qué frecuencia 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.

Implementación de Ventana Deslizante

Algoritmo (desde `qer.h`):

```
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, restablecerla
    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**: Usa punteros volátiles para acceso concurrente
- **MBR = 0**: Tratado como ancho de banda ilimitado

Ejemplo de Cálculo de QoS

Escenario: MBR = 100 Mbps, Tamaño de 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 pronto)
   - 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
   Intervalo 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 de PDR (Hash)	0.3 µs	0.9 µs
Verificación de Tasa de QER	0.1 µs	1.0 µs
Procesamiento de FAR	0.5 µs	1.5 µs
Actualización de URR	0.2 µs	1.7 µs
Encapsulación/Decapsulación de 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 usa ~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 de 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:

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 de Lado de Recepción) para distribuir colas RX
3. Fijar programas eBPF a núcleos específicos

Ajuste de NIC:

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

Ajuste del Núcleo:

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

# Deshabilitar balanceo de IRQ para núcleos XDP
systemctl stop irqbalance

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

# Aumentar tamaños de buffer 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 Máximo × 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 UI Web** - Uso del panel de control
- **Guía de Solución de Problemas** - Problemas comunes y diagnósticos

Guía de Configuración de OmniUPF

Tabla de Contenidos

Descripción General

OmniUPF es una función de plano de usuario versátil que puede operar en múltiples modos para soportar tanto redes centrales 4G (EPC) como 5G. La configuración se gestiona a través de archivos de configuración YAML.

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 forma independiente

---

Modos de Adherencia XDP

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

Para obtener instrucciones detalladas sobre la configuración de XDP, especialmente para Proxmox y otros hipervisores, consulte la Guía de Modos XDP.

Comparación de Modos

Modo	Punto de Adherencia	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
Descarga	Hardware de NIC (SmartNIC)	~10-40 Mpps	Producción de alto rendimiento	SmartNIC con descarga XDP

Modo Genérico (Predeterminado)

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:

```
xdp_attach_mode: generic
```

Mejor para:

- Máquinas virtuales sin SR-IOV
- Entornos de prueba 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 menor que el modo genérico
- Funciona en bare metal y VMs con SR-IOV

Desventajas:

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

Configuración:

```
xdp_attach_mode: native
```

Mejor para:

- Despliegues de producción en bare metal
- VMs con paso 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 XDP habilitado

Modo de Descarga (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:

```
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 de descarga XDP (Netronome Agilio CX, Mellanox BlueField)
- Firmware y controladores especializados

Parámetros de Configuración

Interfaces de Red

Parámetro	Descripción	Tipo	Predeterminado
`interface_name`	Interfaces de red para tráfico N3/N6/N9 (puntos de adherencia XDP)	Lista	`[lo]`
`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 a UPF para ULCL)	IP	Igual que `n3_address`

Ejemplo:

```
interface_name: [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)	Host:Puerto	`:8805`
`pfcp_node_id`	ID de Nodo Local	IP	`127.0.0.1`

Parámetro	Descripción	Tipo	Predeterminado
	para el protocolo PFCP		
`pfcp_remote_node`	Pares PFCP remotos (SMF/PGW-C/SGW-C) a conectar	Lista	`[]`
`association_setup_timeout`	Tiempo de espera entre Solicitudes de Configuración de Asociación (segundos)	Entero	`5`
`heartbeat_retries`	Número de reintentos de latido antes de declarar el par como muerto	Entero	`3`
`heartbeat_interval`	Intervalo de latido PFCP (segundos)	Entero	`5`
`heartbeat_timeout`	Tiempo de espera del latido PFCP (segundos)	Entero	`5`

Ejemplo:

```yaml
pfcp_address: :8805
pfcp_node_id: 10.100.50.241
pfcp_remote_node:
  - 10.100.50.10  # OmniSMF
  - 10.100.60.20  # OmniPGW-C
heartbeat_interval: 10
heartbeat_retries: 5
```

API y Monitoreo

Parámetro	Descripción	Tipo	Predeterminado
`api_address`	Dirección local para el servidor API REST	Host:Puerto	`:8080`
`metrics_address`	Dirección local para el endpoint de métricas de Prometheus	Host:Puerto	`:9090`
`logging_level`	Nivel de registro (`trace`, `debug`, `info`, `warn`, `error`)	Cadena	`info`

Ejemplo:

```yaml
api_address: :8080
metrics_address: :9090
logging_level: debug
```

Gestión de Rutas GTP

Parámetro	Descripción	Tipo	Predeterminado
`gtp_peer`	Lista de pares GTP para latidos de Solicitud de Eco	Lista	`[]`
`gtp_echo_interval`	Intervalo entre Solicitudes de Eco GTP (segundos)	Entero	`10`

Ejemplo:

```yaml
gtp_peer:
  - 10.100.50.50:2152  # gNB
  - 10.100.50.60:2152  # Otro UPF para N9
gtp_echo_interval: 15
```

Capacidad del Mapa eBPF

Parámetro	Descripción	Tipo	Predeterminado	Auto-calculado
`max_sessions`	Número máximo de sesiones concurrentes	Entero	`65535`	Usado para calcular tamaños de mapas
`pdr_map_size`	Tamaño del mapa eBPF PDR	Entero	`0`	`max_sessions × 2`
`far_map_size`	Tamaño del mapa eBPF FAR	Entero	`0`	`max_sessions × 2`
`qer_map_size`	Tamaño del mapa eBPF QER	Entero	`0`	`max_sessions`
`urr_map_size`	Tamaño del mapa eBPF URR	Entero	`0`	`max_sessions × 2`

Nota: Configurar los tamaños de mapa a `0` (predeterminado) habilita el auto-cálculo basado en `max_sessions`. Sobrescriba con valores específicos si se necesita un dimensionamiento personalizado.

Ejemplo:

```yaml
max_sessions: 100000
# Los mapas se dimensionarán automáticamente:
# PDR: 200,000 entradas
# FAR: 200,000 entradas
# QER: 100,000 entradas
# URR: 200,000 entradas
```

Ejemplo de dimensionamiento personalizado:

```yaml
max_sessions: 50000
pdr_map_size: 131070  # Tamaño personalizado
far_map_size: 131070
qer_map_size: 65535
urr_map_size: 131070
```

Configuración de Buffers

Parámetro	Descripción	Tipo	Predeterminado
`buffer_port`	Puerto UDP para paquetes almacenados desde eBPF	Entero	`22152`
`buffer_max_packets`	Número máximo de paquetes a almacenar por FAR	Entero	`10000`
`buffer_max_total`	Número máximo total de paquetes almacenados (0=ilimitado)	Entero	`100000`
`buffer_packet_ttl`	TTL para paquetes almacenados en segundos (0=sin expiración)	Entero	`30`
`buffer_cleanup_interval`	Intervalo de limpieza de buffer en segundos (0=sin limpieza)	Entero	`60`

Ejemplo:

```yaml
buffer_port: 22152
buffer_max_packets: 20000
buffer_max_total: 200000
buffer_packet_ttl: 60
buffer_cleanup_interval: 30
```

Banderas 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 asignación de IP de UE (requiere `feature_ueip`)	CIDR	`10.60.0.0/24`
`feature_ftup`	Habilitar la asignación de F-TEID por OmniUPF	Booleano	`false`
`teid_pool`	Tamaño del pool de TEID para asignación de F-TEID (requiere `feature_ftup`)	Entero	`65535`

Ejemplo (asignación de IP de UE):

```yaml
feature_ueip: true
ueip_pool: 10.45.0.0/16  # Asignar IPs de UE desde este pool
```

Ejemplo (asignación de F-TEID):

```yaml
feature_ftup: true
teid_pool: 1000000  # Permitir hasta 1M de asignaciones de TEID
```

Métodos de Configuración

Archivo de Configuración YAML (Recomendado)

Archivo: `config.yml`

```yaml
# Configuración de Red
interface_name: [eth0]
n3_address: 10.100.50.233
n9_address: 10.100.50.233
xdp_attach_mode: native

# Configuración PFCP
pfcp_address: :8805
pfcp_node_id: 10.100.50.241
pfcp_remote_node:
  - 10.100.50.10

# API y Monitoreo
api_address: :8080
metrics_address: :9090
logging_level: info

# Capacidad
max_sessions: 100000

# Pares GTP
gtp_peer:
  - 10.100.50.50:2152
gtp_echo_interval: 10

# Características
feature_ueip: true
ueip_pool: 10.45.0.0/16
feature_ftup: false

# Almacenamiento
buffer_max_packets: 15000
buffer_packet_ttl: 45
```

Iniciando OmniUPF:

```
./eupf --config /path/to/config.yml
```

Compatibilidad con Hypervisores

Descripción General

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

Para obtener 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 Bridge (XDP Genérico)

Caso de uso: Redes estándar de VM

Configuración:

- Dispositivo de Red: VirtIO o E1000
- Modo XDP: `generic`
- Rendimiento: ~1-2 Mpps

Configuración de VM de Proxmox:

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

Configuración de OmniUPF:

```
interface_name: [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 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 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
```

Configuración 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:

```
interface_name: [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:

- NIC física completa pasada a la VM
- Modo XDP: `native` o `offload` (si es SmartNIC)
- Rendimiento: ~5-40 Mpps (depende de la NIC)

Configuración 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:

```
interface_name: [ens1f0]
xdp_attach_mode: native  # o 'offload' para SmartNIC
```

KVM/QEMU

Modo Bridge:

```
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

Switch 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/Bridge (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

Paquetes por segundo (Mpps) y 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 streaming de video.

Impacto del Tamaño del Paquete en el Rendimiento

En 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 × 1,000,000 pps × 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 × 1,000,000 pps × 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 × 1,000,000 pps × 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 × 1,000,000 pps × 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**
Streaming de video (N3)	1400-1450 bytes	1436-1486 bytes	**11.5-11.9 Gbps**	Fragmentos de video HD/4K
MTU máxima (N3)	1500 bytes	1536 bytes	**12.3 Gbps**	Descargas grandes de TCP

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**
Streaming de video	1400-1450 bytes	**11.2-11.6 Gbps**	Descargas de video
MTU máxima	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.

Por qué esto importa para las redes móviles:

El tráfico móvil es **altamente variable** en tamaño de paquete y la sobrecarga GTP-U (36 bytes) impacta significativamente en el rendimiento de paquetes pequeños:

Tamaño de paquete interno (datos de usuario reales):

- **VoIP (códec AMR-WB)**: 65-80 bytes → Con GTP-U: 101-116 bytes
- **Datos de sensores IoT**: 50-200 bytes → Con GTP-U: 86-236 bytes
- **Navegación web (HTTP/3)**: 400-800 bytes → Con GTP-U: 436-836 bytes
- **Streaming de video**: 1200-1450 bytes → Con GTP-U: 1236-1486 bytes
- **Descargas grandes**: 1500 bytes → Con GTP-U: 1536 bytes

Impacto de la sobrecarga GTP-U:

- Paquetes pequeños (< 200 bytes): **~35-70% de sobrecarga** - Mpps es el factor limitante
- Paquetes medianos (200-800 bytes): **~5-20% de sobrecarga** - Limitación mixta
- Paquetes grandes (> 1200 bytes): **~3% de sobrecarga** - La velocidad de enlace es el factor limitante

Planificación de Rendimiento:

Una NIC clasificada en **10 Mpps** logrará en la interfaz N3:

- **Tráfico pesado de VoIP** (paquetes internos de 100 bytes): ~1.0 Gbps (la sobrecarga GTP-U domina)
- **Mezcla móvil moderna** (paquetes internos promedio de 1200 bytes): ~9.9 Gbps
- **Tráfico pesado de video** (paquetes internos de 1400 bytes): ~11.5 Gbps
- **Rendimiento máximo** (paquetes internos de 1500 bytes): ~12.3 Gbps

En la interfaz N6 (sin sobrecarga GTP-U):

- **Mezcla móvil moderna** (paquetes de 1200 bytes): ~9.6 Gbps a 10 Mpps
- **Rendimiento máximo** (paquetes de 1500 bytes): ~12.0 Gbps a 10 Mpps

Regla General para UPF Móvil:

- **Tráfico de paquetes pequeños** (VoIP, IoT, señalización): Mpps es limitante - planifique para 1-2 Gbps por 10 Mpps
- **Tráfico móvil moderno** (promedio de 1200 bytes): Planifique para ~9-10 Gbps por 10 Mpps de capacidad
- **Tráfico pesado de video** (streaming, descargas): Planifique para ~10-12 Gbps por 10 Mpps de capacidad
- **Siempre considere tanto N3 como N6** - N3 tiene sobrecarga GTP-U, N6 no

Planificación de Capacidad Práctica:

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

Capacidad Mpps de NIC	Rendimiento N3 (GTP-U)	Rendimiento N6 (IP Nativa)	Escenario de 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	Metro grande, hub regional

Nota: Estas estimaciones suponen un tamaño de carga útil promedio de 1200 bytes, que es representativo del tráfico móvil moderno dominado por streaming de video, redes sociales y aplicaciones en la nube. El rendimiento real variará según la mezcla de tráfico.

Controladores de Red Compatibles con XDP

OmniUPF requiere controladores de red con soporte XDP para los modos **nativo** y **descarga**. 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 + Descarga	~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	Descarga	~30 Mpps
Amazon ENA	ena	Sí	Nativo	~5 Mpps
Solarflare SFC9xxx	sfc	Sí	Nativo	~8 Mpps
VirtIO	virtio_net	Limitado	Genérico	~2 Mpps

Comprobando Soporte XDP de NIC

Verifique si el controlador soporta XDP:

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

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

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

Verificar la adherencia del programa XDP:

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

# Ejemplo de salida (XDP adherido):
# 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	Rendimiento (N3)	Escenario de Despliegue
Pruebas/ Desarrollo	Cualquier NIC (VirtIO, E1000)	Genérico	1-2 Mpps	1-2 Gbps	Pruebas de laboratorio, PoC
Sitio de	Intel X710,	Nativo	5-10 Mpps	5-10 Gbps	Celda rural,

Caso de Uso	NIC Recomendada	Modo	Capacidad Mpps	Rendimiento (N3)	Escenario de Despliegue
Pequeña Celda	Mellanox CX-5				empresa
Celda Mediana/ Metro	Intel E810, Mellanox CX-6	Nativo	10-20 Mpps	10-20 Gbps	Celda urbana, pequeña ciudad
Metro Grande	Mellanox CX-6, Intel E810 (dual)	Nativo	20-40 Mpps	20-40 Gbps	Área metropolitana, ciudad mediana
Hub Regional	Mellanox BlueField, Netronome Agilio	Descarga	40+ Mpps	40+ Gbps	Agregación regional
VM de Proxmox (Bridge)	VirtIO	Genérico	1-2 Mpps	1-2 Gbps	Solo pruebas
VM de Proxmox (SR-IOV)	Intel X710/E810 VF, Mellanox CX-5 VF	Nativo	5-10 Mpps	5-10 Gbps	VM de producción

Estimaciones de Rendimiento:

- Basado en un tamaño de paquete promedio de 1200 bytes con encapsulación GTP-U (1236 bytes en N3)
- El rendimiento N6 es ligeramente inferior (~9.6 Gbps por 10 Mpps) debido a la ausencia de sobrecarga GTP-U
- El rendimiento real varía con la mezcla de tráfico - redes pesadas en VoIP verán un rendimiento inferior

---

Recursos Adicionales

Documentación Oficial de XDP:

- Proyecto XDP
- Documentación XDP del Núcleo

Listas de Compatibilidad de NIC:

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

Ejemplos de Configuración

Ejemplo 1: Entorno de Desarrollo (Modo Genérico)

Escenario: Probar OmniUPF en laptop o VM sin SR-IOV

```yaml
# Configuración de desarrollo
interface_name: [eth0]
xdp_attach_mode: generic
api_address: :8080
pfcp_address: :8805
pfcp_node_id: 127.0.0.1
n3_address: 127.0.0.1
metrics_address: :9090
logging_level: debug
max_sessions: 1000
```

Ejemplo 2: Producción Bare Metal (Modo Nativo)

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

```yaml
# Configuración bare metal de producción
interface_name: [ens1f0, ens1f1]  # N3 en ens1f0, N6 en ens1f1
xdp_attach_mode: native
api_address: :8080
pfcp_address: 10.100.50.241:8805
pfcp_node_id: 10.100.50.241
n3_address: 10.100.50.233
n9_address: 10.100.50.234
metrics_address: :9090
logging_level: info
max_sessions: 500000
gtp_peer:
  - 10.100.50.10:2152  # gNB 1
  - 10.100.50.11:2152  # gNB 2
gtp_echo_interval: 30
pfcp_remote_node:
  - 10.100.50.50  # OmniSMF
heartbeat_interval: 10
feature_ueip: true
ueip_pool: 10.45.0.0/16
buffer_max_packets: 50000
buffer_packet_ttl: 60
```

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

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

```yaml
# Configuración de Proxmox SR-IOV
interface_name: [ens1f0]  # VF SR-IOV
xdp_attach_mode: native
api_address: :8080
pfcp_address: 192.168.100.10:8805
pfcp_node_id: 192.168.100.10
n3_address: 192.168.100.10
metrics_address: :9090
logging_level: info
max_sessions: 100000
gtp_peer:
  - 192.168.100.50:2152
gtp_echo_interval: 15
pfcp_remote_node:
  - 192.168.100.20  # SMF
```

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

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

```yaml
# Configuración PGW-U
interface_name: [eth0]
xdp_attach_mode: native
api_address: :8080
pfcp_address: 10.200.1.10:8805
pfcp_node_id: 10.200.1.10
n3_address: 10.200.1.10  # Interfaz S5/S8 (GTP-U)
metrics_address: :9090
logging_level: info
max_sessions: 200000
gtp_peer:
  - 10.200.1.50:2152  # SGW-U
gtp_echo_interval: 20
pfcp_remote_node:
  - 10.200.2.10  # OmniPGW-C (interfaz Sxb)
heartbeat_interval: 5
```

Ejemplo 5: Modo Múltiple (UPF + PGW-U Simultáneamente)

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

```yaml
# Configuración de modo múltiple
```

```yaml
interface_name: [eth0, eth1]
xdp_attach_mode: native
api_address: :8080
pfcp_address: :8805
pfcp_node_id: 10.50.1.100
n3_address: 10.50.1.100
n9_address: 10.50.1.101
metrics_address: :9090
logging_level: info
max_sessions: 300000
gtp_peer:
  - 10.50.2.10:2152  # gNB 5G
  - 10.50.2.20:2152  # eNodeB 4G (a través de SGW-U)
gtp_echo_interval: 15
pfcp_remote_node:
  - 10.50.3.10  # OmniSMF (5G)
  - 10.50.3.20  # OmniPGW-C (4G)
heartbeat_interval: 10
feature_ueip: true
ueip_pool: 10.60.0.0/16
```

Ejemplo 6: Modo de Descarga SmartNIC

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

```yaml
# Configuración de SmartNIC de descarga
interface_name: [enp1s0np0]  # Interfaz SmartNIC
xdp_attach_mode: offload
api_address: :8080
pfcp_address: 10.10.1.50:8805
pfcp_node_id: 10.10.1.50
n3_address: 10.10.1.50
metrics_address: :9090
logging_level: warn  # Reducir sobrecarga
max_sessions: 1000000
pdr_map_size: 2000000
far_map_size: 2000000
qer_map_size: 1000000
gtp_peer:
  - 10.10.2.10:2152
  - 10.10.2.20:2152
  - 10.10.2.30:2152
gtp_echo_interval: 30
pfcp_remote_node:
  - 10.10.3.10
heartbeat_interval: 15
```

```yaml
buffer_max_packets: 100000
buffer_max_total: 1000000
```

Dimensionamiento de Mapas y Planificación de Capacidad

Auto-Dimensionamiento (Recomendado)

Establezca `max_sessions` y deje que OmniUPF calcule los tamaños de mapa automáticamente:

```yaml
max_sessions: 100000
# Tamaños auto-calculados:
# PDR: 200,000 entradas (2 × max_sessions)
# FAR: 200,000 entradas (2 × max_sessions)
# QER: 100,000 entradas (1 × max_sessions)
# URR: 200,000 entradas (2 × max_sessions)
```

Uso de memoria: ~91 MB para 100K sesiones

Dimensionamiento Manual

Sobrescriba el auto-cálculo para requisitos personalizados:

```yaml
max_sessions: 100000
pdr_map_size: 300000  # Soportar más PDRs por sesión
far_map_size: 200000
qer_map_size: 150000  # Más QERs que el predeterminado
urr_map_size: 200000
```

Estimación de Capacidad

Calcular sesiones máximas:

```
Sesiones Máximas = 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

Sesiones Máximas = min(100,000, 100,000, 100,000) = **100,000**

Requisitos de Memoria

Uso de memoria por sesión:

- PDR: 2 × 212 B = 424 B
- FAR: 2 × 20 B = 40 B
- QER: 1 × 36 B = 36 B
- URR: 2 × 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 2× 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 del rendimiento
- **Guía de Gestión de Reglas** - Configuración de PDR, FAR, QER, URR
- **Guía de Monitoreo** - Estadísticas, monitoreo de capacidad y alertas
- **Guía de Interfaz Web** - Operaciones del panel de control
- **Guía de Operaciones** - Descripción general de la arquitectura y despliegue de UPF

Guía de Monitoreo

Tabla de Contenidos

Descripción General

El monitoreo efectivo de OmniUPF es crítico para mantener la calidad del servicio, prevenir el agotamiento de capacidad y solucionar 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 protocolo
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 el agotamiento de recursos	Cada 10s	Porcentajes de uso del mapa, usado vs. capacidad
Estadísticas de Buffer	Rastrear el almacenamiento de paquetes durante la movilidad	Cada 5s	Paquetes en buffer, edad del buffer, recuento 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 la RAN (tráfico GTP-U de subida)
- **TX N3**: Paquetes transmitidos a la RAN (tráfico GTP-U de bajada)
- **RX N6**: Paquetes recibidos de la Red de Datos (IP nativa de bajada)
- **TX N6**: Paquetes transmitidos a la Red de Datos (IP nativa de subida)
- **Total**: Recuento agregado de paquetes a través de todas las interfaces

Comportamiento Esperado:

- **RX N3 ≈ TX N6**: Los paquetes de subida fluyen de la RAN a la Red de Datos
- **RX N6 ≈ TX N3**: Los paquetes de bajada fluyen de la Red de Datos a la RAN
- Un desequilibrio significativo puede indicar:
    - Tráfico asimétrico (descargas >> subidas)
    - 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:

- **Abortados**: El programa XDP encontró un error (debería ser siempre 0)
- **Descartados**: Paquetes descartados intencionalmente por el programa XDP
- **Pasados**: Paquetes pasados a la pila de red para un procesamiento adicional
- **Redirigidos**: Paquetes redirigidos directamente a la interfaz de salida
- **TX**: Paquetes transmitidos a través de XDP

Interpretación:

- **Abortados > 0**: Problema crítico con el programa eBPF o compatibilidad del kernel
- **Descartados > 0**: Descartes basados en políticas o paquetes inválidos
- **Pasos altos**: La mayoría de los paquetes procesados en la pila de red

(normal)
- **Redirecciones altas**: 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 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 recuento 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 incorrecta

---

Estadísticas de Rutas

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

Búsqueda de FIB IPv4:

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

Búsqueda de FIB IPv6:

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

- **OK**: Búsquedas de ruta IPv6 exitosas

Indicadores de Rendimiento:

- **Alta Tasa de Aciertos en Caché**: Indica un buen rendimiento de la caché de enrutamiento
- **Alto Recuento de OK**: Confirma que las tablas de enrutamiento están configuradas correctamente
- **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 de 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 (PDRs de Bajada - IPv4):

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

pdr_map_downlin_ip6 (PDRs de Bajada - IPv6):

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

pdr_map_teid_ip (PDRs de Subida):

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

qer_map (Reglas de Aplicación de QoS):

- **Capacidad**: 65,535 entradas
- **Tamaño de Clave**: 4 B (ID de 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 de URR)
- **Tamaño de Valor**: 16 B (contadores de volumen)
- **Uso de Memoria**: ~2.6 MB
- **Criticidad**: Baja - Afecta solo 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 las 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 OmniUPF
2. Actualizar el archivo de configuración del UPF con los nuevos tamaños de mapa
3. Reiniciar el servicio OmniUPF

4. Verificar la nueva capacidad en la vista de Capacidad
5. Monitorear 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

Tasa de Procesamiento de Paquetes

Cálculo:

```
Tasa de Paquetes (pps) = (Delta de Recuento 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:

- Contador de XDP abortados en aumento
- Alta utilización de CPU
- Aumento en las caídas de paquetes
- Aumento en la latencia

Cálculo de Rendimiento

Cálculo:

```
Rendimiento (Mbps) = (Delta de Recuento 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 de QER (MBR excedido)
- Fallos en la búsqueda de mapas 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%
- Contador de XDP abortados > 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 de buffer excedido (paquetes mayores a 30s)

Alertas Informativas (Monitorear tendencias):

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

Configuración de Alertas

Las alertas se pueden configurar a través de:

1. **Métricas de Prometheus**: Exportar métricas para monitoreo externo
2. **Monitoreo de Registros**: Analizar registros de OmniUPF para patrones de error
3. **Polling de API REST**: Consultar periódicamente los endpoints `/map_info`, `/packet_stats`
4. **Monitoreo de Interfaz 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,  # PDRs de bajada + subida por sesión
  Capacidad del Mapa FAR / 2,  # FARs de bajada + subida 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 un 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 recuento diario de sesiones pico
2. Calcular la tasa de crecimiento semanal
3. Extrapolar al límite de capacidad

Fórmula de Tasa de Crecimiento:

```
Semanas hasta la 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 una actualización de capacidad en 12 semanas (dejando un margen de 5 semanas).

Solución 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 de QER**: Verificar valores de MBR de QER vs. tráfico real
- **TEIDs Inválidos**: Verificar que el TEID de PDR de subida coincida con la asignación de gNB
- **IPs de UE Desconocidas**: Verificar que exista un PDR de bajada para la IP de 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: XDP abortado > 0

Diagnóstico:

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

Causas Comunes:

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

Resolución:

- Reiniciar el servicio OmniUPF
- Verificar que la versión del kernel cumpla con los requisitos mínimos (Linux 5.4+)
- Revisar los registros 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 un reinicio del 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 la 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 del UPF
4. Verificar la utilización de la interfaz N3/N6

Causas Comunes:

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

Resolución:

- Escalar el UPF horizontalmente (agregar más instancias)
- Mejorar la CPU o habilitar RSS (Escalado Lateral de Recepción)
- Mejorar las interfaces de red a velocidades más altas
- Ajustar la función hash del mapa eBPF

Documentación Relacionada

- **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 Interfaz Web** - Funciones de monitoreo del panel de control
- **Guía de Solución de Problemas** - Problemas comunes y diagnósticos
- **Guía de Arquitectura** - Datapath eBPF y optimización del 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 infraestructura y complejidad operativa
- **Computación en la nube** - 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 de 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 enlace ascendente, 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), descapsula, reenvía a Internet
5. **Total: 2 pasadas de 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 enlace ascendente
3. **Detección de bucle:** IP de destino = IP local (10.0.1.10)
4. **Procesamiento en el lugar:** Actualiza GTP TEID a 200 (sesión PGWU)
5. rol PGWU: Descapsula, reenvía a Internet
6. **Total: 1 pasada de XDP, cero saltos de red**

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

---

Detalles del Procesamiento de Paquetes

Flujo de Enlace Ascendente: eNodeB → SGWU → PGWU → Internet

Ruta del Código: `cmd/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 enlace ascendente 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 núcleo)
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 Enlace Descendente: Internet → PGWU → SGWU → eNodeB

Ruta del Código: `cmd/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 enlace descendente 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 puertos diferentes)

Configuración de OmniUPF

config.yml:

```yaml
# Interfaces de red
interface_name: [eth0]              # Una sola interfaz para S1-U y N9
xdp_attach_mode: native             # Usar nativo para mejor rendimiento

# Interfaz PFCP
pfcp_address: ":8805"               # Escuchar en todas las interfaces, puerto 8805
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: "10.0.1.10"             # IP de la interfaz N9 (MISMA que N3)

# APIs
api_address: ":8080"                # API REST
metrics_address: ":9090"            # Métricas de Prometheus

# Grupos de Recursos
ueip_pool: "10.60.0.0/16"           # Grupo de direcciones IP de UE
```

```
teid_pool: 65535                    # Grupo de asignación de TEID

# Capacidad
max_sessions: 100000                # Máximo de sesiones concurrentes
de UE
```

Configuración Clave:

- ⚠ **`n3_address` y `n9_address` DEBEN ser idénticos** para habilitar el bucle
- ⚠ Dirección 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 dirección n3 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 enlace ascendente:** Coincide TEID=100 (desde eNodeB) → FAR: Encapsular TEID=200, dst=10.0.1.10
- **PDR de enlace descendente:** Coincide TEID=200 (desde PGWU) → FAR: Actualizar túnel TEID=100, reenviar a eNodeB

Sesión PGWU (desde OmniPGW-C):

- **PDR de enlace ascendente:** Coincide TEID=200 (desde SGWU) → FAR: Descapsular, reenviar a Internet
- **PDR de enlace descendente:** Coincide IP UE=10.60.0.1 → FAR: Encapsular TEID=200, dst=10.0.1.10

Monitoreo y Verificación

Verificar que el Bucle N9 está Activo

Verificar Registros de 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] descapsulado, enrutando a N6

upf: [n6] usar mapeo 10.60.0.1 -> teid:200
upf: [n6-loopback] detección de auto-reenvío de enlace descendente,
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 al Bucle

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

Causa Raíz: `n3_address` ≠ `n9_address`

Solución:

```
# 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: "10.0.1.10"   # 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 el 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 un TEID coincidente que SGWU-C use para el reenvío N9

Uso Alto de CPU

Síntoma: Uso de CPU más alto de lo esperado

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

Diagnóstico:

```
# Verificar 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 de búsqueda)
- Verificar que la limitación de tasa de QER no esté causando pérdidas 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:

```
buffer_port: 22152
buffer_max_packets: 20000      # Aumentar para redes de alta movilidad
buffer_max_total: 100000
buffer_packet_ttl: 30          # Ajustar según el tiempo de transferencia
```

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 de XDP + pila de red	1× pasada de XDP	**Reducción del 40-50%**
Pérdida de Paquetes	Riesgo durante la congestión de la 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
- **Ahorro de Costos:** Reducción de hardware, energía, refrigeración, mantenimiento
- **Resolución de Problemas Más Fácil:** Un solo rastreo de paquetes, una sola salida de depuración eBPF

Casos de Uso

Ideal Para:

- 🔷 **Computación en la Nube:** Minimizar la latencia para ruptura local
- 🔷 **Despliegues Pequeños/Medianos:** < 100K suscriptores
- 🔷 **Laboratorio/Pruebas:** Plano de usuario EPC completo en una sola VM
- 🔷 **Limitaciones de Costos:** 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 obligatoria de SGW y PGW

Comparación con Otros Modos de Implementación

Instancia Única (Bucle N9) vs. Instancias Separadas

Característica	Bucle N9	Separadas	Contenedores
Latencia	⚡ < 1 µs	🔷 1-5 ms	🔷 5-20 ms
Rendimiento	⚡ 40+ Gbps	🔷 20+ Gbps	🔷 10+ Gbps
Infraestructura	🔷 1 servidor	🔷 2 servidores	⚠ 1 servidor, 2 VMs
Complejidad	🔷 Simple	🔷 Compleja	⚠ Moderada
Costo	🔷 Más Bajo	🔷 Más Alto	⚠ Medio
Escalabilidad	⚠ Solo vertical	🔷 Horizontal	🔷 Horizontal
Redundancia	🔷 Punto único de falla	🔷 Redundancia geográfica	⚠ Redundancia local

---

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-microsegundos** 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 las condiciones de bucle y procesa los paquetes en línea.

Para más información:

- **Configuración:** CONFIGURATION.md
- **Arquitectura:** ARCHITECTURE.md
- **Operaciones:** OPERATIONS.md
- **Solución de Problemas:** TROUBLESHOOTING.md

Guía de Gestión de Reglas

Tabla de Contenidos

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 facturación	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 de Terminación 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, 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 de la navegación web
- Aplicar diferentes QoS a VoIP frente a datos de mejor esfuerzo
- Enrutar aplicaciones específicas a través de diferentes caminos 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

Flags de Acción

Las acciones de FAR son flags bit a bit que pueden combinarse:

Flag	Bit	Valor	Descripción
REENVIAR	1	2	Reenviar paquete al destino
BUFFER	2	4	Almacenar paquete en el 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

El flag BUFFER (bit 2) controla el almacenamiento de paquetes durante eventos de movilidad.

Operaciones de Almacenamiento:

- **Habilitar Almacenamiento**: Establecer bit 2 de la acción FAR (Acción |= 4)
- **Deshabilitar Almacenamiento**: Limpiar bit 2 de la acción FAR (Acción &= ~4)
- **Vaciar Almacenamiento**: Reenviar todos los paquetes almacenados utilizando las reglas FAR actuales
- **Limpiar Almacenamiento**: Descartar todos los paquetes almacenados sin reenviar

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 los detalles de FAR

Información Mostrada:

- ID de FAR
- Acción (numérica + flags decodificados)
- Estado de almacenamiento (ENCENDIDO/APAGADO)
- 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 los flujos de tráfico, incluidos límites de ancho de banda y marcado de paquetes.

Estructura de QER

Parámetros de QoS

QFI (Identificador de Flujo de QoS):

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

Estado de la 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 es descartado

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 de 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

Reglas de Reporte de Uso (URR)

Propósito

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

Estructura de URR

Seguimiento de Volumen

Volumen Ascendente:

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

Volumen Descendente:

- Bytes transmitidos desde la Red de Datos a UE
- Medido antes de la encapsulación 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 disparar 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 por 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

Relaciones de Reglas

Cadena PDR → FAR → QER → URR

Cada PDR referencia un FAR, que puede referenciar un QER y uno 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
```

ID de FAR 1 (Descendente):

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

ID de FAR 2 (Ascendente):

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

ID de QER 1:

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

ID de URR 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. Usar 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 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 por 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 el 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 por 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 mayores usuarios
4. Actualizar periódicamente para observar el crecimiento del volumen

Casos de Uso:

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

Resolución 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 del 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. Vaciar o limpiar manualmente el buffer
5. Deshabilitar el almacenamiento en FAR

Desbordamiento del buffer:

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

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 el 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 configuración del Informe de Sesión PFCP
2. Verificar disparadores de reporte de URR (umbrales de volumen/tiempo)
3. Revisar registros para 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 PFCP** - Gestión de sesiones PFCP e instalación de reglas
- **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 Resolución de Problemas** - Problemas comunes y diagnósticos

Guía de Solución de Problemas de OmniUPF

Tabla de Contenidos

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. Verificar que OmniUPF esté en ejecución
systemctl status omniupf  # o ps aux | grep eupf

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

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

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

# 5. Verificar registros del kernel en busca de errores
```

```
dmesg | tail -50
journalctl -u omniupf -n 50
```

Herramientas de Diagnóstico

API REST de OmniUPF

Verificar estado de UPF:

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

Verificar asociaciones PFCP:

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

Verificar conteo de sesiones:

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

Verificar 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 GTP-U en N3:

```
tcpdump -i eth0 -n udp port 2152 -w /tmp/n3_traffic.pcap
```

Capturar tráfico PFCP en N4:

```
tcpdump -i eth0 -n udp port 8805 -w /tmp/pfcp_traffic.pcap
```

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: "Operación no permitida" al cargar eBPF

Síntomas:

```
ERRO[0000] failed to load eBPF program: operation not permitted
```

Causa: Capacidades insuficientes o límites de memoria bloqueada

Resolución:

```
# Verificar límite actual de memoria bloqueada
ulimit -l

# Establecer memoria bloqueada ilimitada (requerido para eBPF)
ulimit -l unlimited

# Hacer persistente (agregar a /etc/security/limits.conf)
echo "* soft memlock unlimited" | sudo tee -a /etc/security/
limits.conf
echo "* hard memlock unlimited" | sudo tee -a /etc/security/
limits.conf

# Ejecutar OmniUPF con capacidades requeridas
sudo setcap cap_sys_admin,cap_net_admin,cap_bpf+eip /usr/bin/eupf
```

```
# O ejecutar con sudo
sudo ./eupf
```

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 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:
# - Sangrías incorrectas (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: 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 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: ID de Nodo PFCP no es una dirección IPv4 válida

Resolución:

```
# 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 Asociaci��n PFCP

Problema: No se establecieron asociaciones PFCP

Síntomas:

- La interfaz web muestra "No hay asociaciones"
- Los registros de SMF muestran "Fallo en el Establecimiento de 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 a 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 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 heartbeat de PFCP

Síntomas:

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

Diagnóstico:

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

# Monitorear registros de heartbeat
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 heartbeat demasiado agresivo

Resolución:

```
# Aumentar intervalo de heartbeat
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. Capturar tráfico en la interfaz
tcpdump -i eth0 -n -c 10

# 4. Verificar estadísticas de paquetes
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 desconocido o IP de UE)

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 del 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

Resolución:

```
# Verificar estadísticas de QER
curl http://localhost:8080/api/v1/sessions | jq '.[].qers'

# Si MBR (Tasa Máxima de Bits) es demasiado baja, solicitar a SMF que
actualice QER
# O verificar si el tráfico excede las tasas configuradas
```

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 la interfaz
```

```
curl http://localhost:8080/api/v1/packet_stats | jq
'.interface_stats'

# Capturar tráfico en ambas interfaces
tcpdump -i eth0 -n udp port 2152 &  # N3
tcpdump -i eth1 -n not udp port 2152 &  # N6
```

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 a 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 encapsulación GTP faltante

Resolución:

```
# Verificar que exista PDR de downlink para IP de UE
curl http://localhost:8080/api/v1/sessions | jq '.[].pdrs[] |
select(.pdi.ue_ip_address)'

# Verificar que FAR tenga CREACIÓN_DE_HEADER_EXTERNO
curl http://localhost:8080/api/v1/sessions | jq '.[].fars[] |
.outer_header_creation'

# Verificar alcanzabilidad de gNB
ping <GNB_N3_IP>
```

Problemas de XDP y eBPF

Para configuración detallada de XDP, selección de modo y solución de problemas, consulte la Guía de Modos XDP.

Problema: Programa XDP falló al 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:

```
# Reconstruir 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 nivel de registro del verificador eBPF para depuración
sudo sysctl kernel.bpf_stats_enabled=1
```

Problema: Contador de abortos de XDP en aumento

Síntomas:

- 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 de eBPF (requiere
reconstrucción):
# Construir 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 **elimina 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'

# Monitorear uso de CPU
top -bn1 | grep eupf

# 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 Escalado Lateral) 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 buffer

Resolución:

```
# Reducir límites de buffer 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 buffer
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 exceso

Resolución:

```
# Verificar por qué los paquetes están en buffer
curl http://localhost:8080/api/v1/upf_buffer_info | jq '.buffers[] |
{far_id, packet_count, direction}'

# Limpiar buffers 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 construcción)
# Construir 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 buffer de anillo
ethtool -g eth0

# Monitorear caídas en tiempo real
watch -n 1 'ethtool -S eth0 | grep -E "drop|miss"'
```

Resolución:

```
# Aumentar tamaño del buffer RX
ethtool -G eth0 rx 4096

# Aumentar tamaño del buffer TX
ethtool -G eth0 tx 4096

# Verificar nuevas configuraciones
ethtool -g eth0
```

Problemas Específicos de 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 SR-IOV passthrough

```
# En el host de 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 UI 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 modo promiscuo "Permitir Todo"
# - 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 de NIC no soporta XDP

Síntomas:

```
ERRO[0000] failed to attach XDP program: operation not supported
```

Diagnóstico:

```
# Verificar controlador de 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 de NIC

```
# Verificar actualizaciones de controlador (Ubuntu)
sudo apt update
sudo apt install linux-modules-extra-$(uname -r)

# O instalar controlador específico del vendedor
# 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. Reportar error al vendedor de NIC o al equipo del kernel de
Linux
```

Fallos en el Establecimiento de Sesiones

Problema: Fallos en el establecimiento de sesiones

Síntomas:

- SMF informa fallo en el establecimiento de sesión
- UE no puede establecer sesión PDU

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 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 Buffering

Problema: Paquetes atascados en el buffer

Síntomas:

- Conteo de paquetes en buffer en aumento
- Paquetes no entregados después de la transferencia

Diagnóstico:

```
# Verificar estadísticas de buffer
curl http://localhost:8080/api/v1/upf_buffer_info

# Verificar buffers individuales de FAR
curl http://localhost:8080/api/v1/upf_buffer_info | jq '.buffers[] |
{far_id, packet_count, oldest_packet_ms}'

# Monitorear tamaño del buffer
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 Modificación de Sesión PFCP 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 Solicitud de Modificación de Sesión PFCP
# - Actualice FAR con acción FORW
```

TTL de buffer expirado

Causa: Paquetes expiraron antes de la actualización de FAR

Resolución:

```
# Aumentar TTL de buffer
buffer_packet_ttl: 60  # Aumentar de 30 a 60 segundos
```

Desbordamiento de buffer

Causa: Demasiados paquetes almacenados por FAR

Resolución:

```
# Aumentar límites de buffer
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 Programa eBPF

```
# Trazar ejecución del programa 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 a Nivel XDP

```
# Capturar paquetes antes de XDP (tcpdump)
tcpdump -i eth0 -w /tmp/before_xdp.pcap

# Capturar paquetes después de XDP (requiere XDP_PASS)
tcpdump -i any -w /tmp/after_xdp.pcap

# Comparar conteos de paquetes para identificar caídas
```

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 registros relevantes
    - Pasos para reproducir
3. **Soporte comunitario**:
    - Problemas de GitHub: https://github.com/edgecomllc/eupf/issues
    - Documentación: Ver guías relacionadas a continuación

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
- **Guía de Gestión de Reglas** - Conceptos de PDR, FAR, QER, URR
- **Guía de Operaciones** - Arquitectura y descripción general de UPF

Guía de Operaciones de la Interfaz Web

Tabla de Contenidos

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 de mapas eBPF
- **Visualización de registros en vivo** para la resolución 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 de mapas eBPF

Acceso al Panel de Control

Acceso Predeterminado

El panel de control es accesible a través de HTTPS en el servidor de gestión de OmniUPF:

```
https://<upf-server>:443/
```

Puerto Predeterminado: 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 de mapas eBPF
- **Config** - `/upf_config` - Configuración de UPF y direcciones de dataplane
- **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
ID de Siguiente Sesión	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 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 expansión para ver información detallada de la regla

Características:

- **Filtrar por IP**: Encontrar sesiones para una dirección IP de UE específica
- **Filtrar por TEID**: Encontrar sesiones por ID de punto final de túnel
- **Expandir sesión**: Ver detalles completos en JSON de PDR/FAR/QER/URR
- **Auto-refrescar**: Actualizaciones 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

- **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, total de bytes)

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 el 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

Resolver 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 accesible y está respondiendo
- **NO SALUDABLE** (rojo): UPF no es accesible 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 y informes de uso en todas las sesiones.

Pestaña PDR - Reglas de Detección de Paquetes

Ver e inspeccionar todos los PDR en el UPF:

PDRs de Ascendente (N3 → N6):

- **TEID**: ID de punto final de túnel GTP-U desde gNB
- **ID de FAR**: Regla de acción de reenvío asociada
- **ID de QER**: Regla de aplicación de QoS asociada (si la hay)
- **Eliminación de Encabezado Externo**: Bandera de desencapsulación GTP-U

PDRs de Descendente (N6 → N3):

- **IP de UE**: Dirección IPv4 o IPv6 del equipo de usuario
- **ID de FAR**: Regla de acción de reenvío asociada
- **ID de QER**: Regla de aplicación de QoS asociada (si la hay)
- **Modo SDF**: Modo de filtro de flujo de datos de servicio (ninguno, solo sdf, sdf + predeterminado)

PDRs de IPv6:

- Tablas separadas para PDRs de ascendente y descendente de IPv6
- Misma estructura que IPv4 pero indexada por direcciones IPv6

Pestaña FAR - Reglas de Acción de Reenvío

Ver todas las FARs con sus acciones de reenvío y parámetros:

Columna	Descripción
ID de FAR	Identificador único de regla de reenvío
Acción	Banderas de acción de reenvío (REENVIAR, DESCARTAR, BUFFER, DUPLICAR, NOTIFICAR)
Buffering	Estado actual de buffering (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 el buffer
- **NOTIFICAR (8)**: Enviar notificación al plano de control
- **DUPLICAR (16)**: Duplicar paquete a múltiples destinos

Alternar Buffering:

- Hacer clic en "Habilitar Buffer" o "Deshabilitar Buffer" para alternar la

bandera de buffering
- Útil para resolver problemas de 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:

Columna	Descripción
ID de QER	Identificador único de regla de QoS
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 de 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:

Columna	Descripción
ID de URR	Identificador único de regla de informes de uso
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
Método	Método de informes (volumen, tiempo, evento)

Visualización de Volumen:

- Formato automático (B, KB, MB, GB, TB)
- Contadores en tiempo real actualizados en cada refresco
- Utilizado para facturación y análisis

Filtrado:

- Solo muestra URRs con volumen distinto de cero
- Limitado a 1000 URRs más activos por rendimiento

Casos de Uso

Inspeccionar Clasificación de Tráfico:

1. Navegar a Reglas → pestaña PDR
2. Buscar TEID o IP de UE específica
3. Verificar que PDR se asocie con el FAR y QER correctos

Resolver 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 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 Buffering:

- **Deshabilitar Buffer** (rojo): Desactivar el buffering para este FAR (actualiza la bandera de acción de FAR)
- **Habilitar Buffer** (púrpura): Activar el buffering 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 Buffering Durante el Traspaso:

1. Durante el traspaso, verificar que los paquetes se estén almacenando
2. Comprobar el estado de buffering de FAR (debería estar habilitado)
3. Monitorear el conteo de paquetes y su antigüedad

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 buffering

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

Resolver Problemas de Desbordamiento de Buffer:

1. Comprobar el conteo total de paquetes vs. máximo total
2. Identificar FARs con buffering excesivo
3. Verificar que SMF haya enviado una modificación de sesión para desactivar el buffering
4. Desactivar manualmente el buffering si se omitió 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 datapath de OmniUPF.

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
- **Éxitos de Reenvío**: Conteo de paquetes reenviados con éxito
- **Errores de Reenvío**: Intentos de reenvío fallidos

Uso: Identificar rutas ocupadas y errores de reenvío

Estadísticas de 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 de XDP procesados vs. 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 10 segundos.

Monitoreo de Capacidad

URL: `/capacity`

Características

La vista de Capacidad muestra el uso de mapas eBPF y los límites de capacidad para todos los mapas en el datapath de UPF.

Tabla de Uso de Mapas 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 de ascendente 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 de descendente 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 la facturación pero no la 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

Resolver Problemas de Fallos en Sesiones:

1. Cuando falla el establecimiento de sesiones, 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 el 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 la configuración del dataplane.

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 inter-UPF (opcional)
- **Puerto API**: Puerto de escucha de la API REST
- **Versión**: Versión del software OmniUPF

Configuración del Dataplane (eBPF)

Muestra parámetros de runtime del dataplane activos:

- **Dirección N3 Activa**: Vinculación de interfaz N3 en runtime
- **Dirección N9 Activa**: Vinculación de interfaz N9 en runtime (si está habilitada)

Estos valores reflejan la configuración real del datapath 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 accesible desde SMF

Resolver Problemas de Interfaces:

1. Comparar la configuración estática con las direcciones activas del dataplane
2. Verificar que las interfaces estén vinculadas correctamente
3. Comprobar si ha habido cambios en la configuración de interfaces

Documentación y Auditoría:

1. Registrar la configuración de UPF para documentación
2. Verificar que la implementación coincida con las especificaciones de diseño
3. Auditar asignaciones de interfaces

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 fallaron en la sincronización

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 a la UE

Propósito:

- Ver todas las direcciones IP de UE que tienen rutas configuradas
- Verificar la distribución de rutas a 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 de router 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 e 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 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): Sesi��n caída o estableciendo

El encabezado también muestra el ID de Router 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
Router Anunciante	ID de router que anuncia 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 router está anunciando rutas externas
- Rastrear la antigüedad y actualizaciones de LSA

Acciones de Control de Rutas

Botón Sincronizar Rutas:

- Inicia 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 Refrescar:

- Refresca manualmente toda la información de rutas
- Actualiza vecinos OSPF, pares BGP y tablas de rutas

Casos de Uso

Monitorear la 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 la Distribución de Rutas de UE:

1. Comprobar la tabla de Rutas IP Activas de UE para una UE específica
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 router anunciante coincida con el UPF esperado

Resolver 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 barra roja si la sincronización falla
4. Comprobar mensajes de error de OSPF/BGP en las secciones respectivas

Verificar Implementación de 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 Tipo-2
- **BGP**: Las rutas se anuncian a los pares BGP configurados
- **Mecanismo de Sincronización**: Las llamadas a la API REST activan 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 dataplane de UPF.

Configuración de la Interfaz

Muestra información de la interfaz de red y del 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 Multi-Colas	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 XDP
- **Indicador**: Marca de verificación verde (✓) si está soportado, X rojo (✗) si no

XDP_SKB (Modo Genérico):

- **Rendimiento**: ~1-2 Mpps
- **Descripción**: Modo de respaldo 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 está realmente en uso

Razones de Modos No Soportados:

- Si un modo no es soportado, el campo "Razón" explica por qué
- Razones comunes: el controlador carece de soporte XDP, incompatibilidad de tipo de interfaz

Vista de Capacidades XDP mostrando la configuración de la interfaz, modos soportados y el calculador interactivo de rendimiento Mpps

Recomendaciones

La vista muestra un banner de recomendación codificado por colores 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 de Cálculo

Rendimiento Total (Gbps):

- Rendimiento de tasa de línea incluyendo todos los encabezados
- Fórmula: `Gbps = Mpps × Tamaño_Paquete × 8 / 1000`
- Incluye encabezados GTP, UDP, IP y Ethernet

Tasa de Datos de Usuario (Gbps):

- Rendimiento real de carga útil de usuario
- Excluye ~50 bytes de sobrecarga de encapsulación GTP
- Fórmula: `Gbps = Mpps × (Tamaño_Paquete - 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 el desglose 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 de la tasa y el 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 el 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 el 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 la 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 multi-colas 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 actuales del modo XDP
3. Determinar si se necesita una actualización de hardware
4. Planificar la selección de interfaz y controlador para nuevas implementaciones

Resolver 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 multi-colas 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 los controladores de NIC a la última versión
- Habilitar multi-colas (RSS) para procesamiento paralelo
- Ajustar los tamaños de buffer de anillo de NIC

Modo Genérico (XDP_SKB):

- Aceptable para desarrollo y pruebas
- No recomendado para producción de alto rendimiento
- Considerar una actualización de hardware para implementaciones de producción

Configuración de Multi-Colas:

- El número de colas debe coincidir o exceder el conteo de núcleos de CPU
- Permite el procesamiento paralelo de paquetes a través de núcleos
- Distribuye la carga a través de RSS (Recibir Escalado Lateral)

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 la resolución de 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

Resolver Problemas de Establecimiento de Sesiones:

1. Abrir la vista de Registros
2. Iniciar el establecimiento de sesión desde SMF
3. Observar los registros de mensajes PFCP y cualquier error

Monitorear la Comunicación PFCP:

1. Ver mensajes de configuración de asociación PFCP
2. Rastrear la 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 mapas eBPF

3. Identificar problemas de configuración de FAR/PDR

Mejores Prácticas

Directrices Operativas

Monitoreo:

- Comprobar regularmente la vista de Capacidad para prevenir el agotamiento de mapas
- Monitorear Estadísticas para patrones de tráfico inusuales o descartes
- Rastrear el crecimiento del conteo de sesiones a lo largo del tiempo
- Estar atento a 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 buffering esté desactivado después de completar el traspaso
- Usar "Vaciar" en lugar de "Limpiar" para evitar la pérdida de paquetes

Gestión de Sesiones:

- Usar 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 resolver problemas

Resolución 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 auto-actualización del panel de control es de 5-10 segundos dependiendo de la vista
- Listas de sesiones grandes pueden tardar en cargarse
- Los filtros de la vista de Reglas se aplican a entradas activas (volúmenes no cero para URRs)
- Las operaciones de buffer se ejecutan inmediatamente en el UPF seleccionado

Documentación Relacionada

- **Guía de Operaciones PFCP** - Gestión de sesiones PFCP y detalles del protocolo
- **Guía de Gestión de Reglas** - Configuración de PDR, FAR, QER, URR
- **Guía de Monitoreo** - Estadísticas, métricas y planificación de capacidad
- **Guía de Rutas** - Detalles de enrutamiento de UE e integración de FRR
- **Guía de Modos XDP** - Documentación detallada de modos XDP e información de eBPF
- **Guía de Resolución de Problemas** - Problemas comunes y diagnósticos
- **Guía de Operaciones UPF** - Operaciones generales de UPF y arquitectura

Modos de Adjunto XDP para OmniUPF

Tabla de Contenidos

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 de calidad 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 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	Cualquier NIC	Controlador	SmartNIC con soporte

Aspecto	Modo Genérico	Modo Nativo	Modo de Offload
NIC		compatible con XDP	XDP
Soporte de Hipervisor	Todos los hipervisores	La mayoría (requiere multi-cola)	Raro (passthrough PCI)
Caso de Uso	Pruebas, desarrollo	**Producción (recomendado)**	Sitios de borde de alto rendimiento
Configuración	`xdp_attach_mode: generic`	`xdp_attach_mode: native`	`xdp_attach_mode: 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 kernel antes de XDP)

Cuándo Usar

- **Desarrollo y pruebas** solamente
- **Entornos de laboratorio** donde el rendimiento no importa
- **Implementación inicial** para verificar la funcionalidad antes de optimizar

Configuración

```
# 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 kernel.

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 de operador** 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

```yaml
# 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

Offload XDP 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 Netronome Agilio actualmente soportan offload XDP:

- Netronome Agilio CX 10G/25G/40G/100G

Nota: El modo de offload requiere **bare metal** o **passthrough PCI** - no disponible en configuraciones estándar de VM.

Configuración

```yaml
# 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 **"Multi-Colas"**
    - 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
```

```
# Establece 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 /etc/eupf/config.yaml
```

Cambia el modo XDP:

```
# Antes
xdp_attach_mode: generic
```

```
# Después
xdp_attach_mode: native
```

Reinicia OmniUPF:

```
sudo systemctl restart eupf
```

Paso 5: Verificar que XDP Nativo esté Activo

Revisa los registros:

```
# Ver registros de inicio
journalctl -u eupf --since "1 minuto atrás" | 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`)
- Verifica la versión del kernel: `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 (evita la sobre suscripción)
- Monitorea `top` - el uso de CPU debería distribuirse entre núcleos
- Verifica 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
    - Establece **Receive Side Scaling** en **Habilitado**
3. **Editar el 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 la configuración de la VM
virsh edit tu-nombre-vm
```

Agrega a la sección de interfaz de red:

```
<interface type='network'>
  <source network='default'/>
  <model type='virtio'/>
  <driver name='vhost' queues='8'/>
</interface>
```

Reinicia la VM y verifica:

```
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:

PowerShell en el host de Hyper-V:

```
# Establecer VMQ (Virtual Machine Queue) - multi-cola de Hyper-V
Set-VMNetworkAdapter -VMName "TuVM" -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: Usa 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. Verificar Registros de OmniUPF

```
# Ver registros recientes
journalctl -u eupf --since "5 minutos atrás" | 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)
```

Solucionando Problemas de XDP

Problema: "Error al adjuntar el programa XDP" al Inicio

Síntomas:

```
Error: fallo al adjuntar el programa XDP a la interfaz eth0
```

Diagnóstico:

1. **Verifica el soporte del controlador**:

   ```
   ethtool -i eth0 | grep driver

   # Si el controlador no es virtio_net/vmxnet3/hv_netvsc, XDP
   nativo no funcionará
   ```

2. **Verifica la versión del kernel**:

   ```
   uname -r

   # Debe ser ≥ 5.15 para un soporte XDP confiable
   ```

3. **Verifica 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 kernel 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` para 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 habilitado**:
    - Verifica: `ethtool -l eth0`
    - Debería mostrar > 1 cola combinada
3. **Soporte XDP en el kernel deshabilitado**:

```
# Verifica si XDP está habilitado en el kernel
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 kernel con soporte 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 distribuirse entre múltiples colas
   ```

2. **Verifica la utilización de CPU**:

   ```
   # Monitorea el uso de CPU por núcleo
   mpstat -P ALL 1

   # Deberías ver la carga distribuida entre múltiples CPUs
   ```

3. **Verifica que XDP esté realmente 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 la 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 mapa eBPF en la configuración
- Verifica si hay paquetes corruptos que causen errores en eBPF
- Verifica que el soporte eBPF del kernel 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 "status: stopped" antes de iniciar
```

Solución:

```
# En el host de Proxmox
# Fuerza el apagado y reinicia
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 las capacidades `CAP_NET_ADMIN` y `CAP_SYS_ADMIN`.

Solución:

1. **Ejecuta OmniUPF como root** (o con capacidades):

   ```
   sudo systemctl restart eupf
   ```

2. **Si usas systemd**, verifica que el archivo de servicio tenga capacidades:

   ```
   # /lib/systemd/system/eupf.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 Conocidos 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 ✓	Comprobada, 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 ✓ (kernel 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 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	Notas
Proxmox/ KVM	VirtIO	virtio_net	Nativo ✓	Sí (configurable)	**Mejor para VMs**
VMware ESXi	vmxnet3	vmxnet3	Nativo ✓	Sí	Requiere ESXi 6.7+
Hyper-V	NIC Sintética	hv_netvsc	Nativo ✓	Sí	Windows Server 2016+
AWS	ENA	ena	Nativo ✓	Sí	Instancias metal EC2
VirtualBox	Cualquiera	varios	Solo genérico ⍰	No	No recomendado para producción

NICs con Soporte de Offload de Hardware

Offload XDP verdadero (eBPF se ejecuta en la NIC):

Proveedor	Modelo	Velocidad	Notas
Netronome	Agilio CX 10G	10GbE	Solo soporte de offload 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 producción de OmniUPF:

NIC/Plataforma	Razón	Alternativa
NICs Realtek	Sin soporte XDP, malos controladores de Linux	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 nuevo

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 <nombre_del_controlador> | grep -i xdp
   ```

2. 🔰 **Versión del Kernel**: Asegúrate de que el kernel sea ≥ 5.15 para un soporte 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: NIC física requerida
    - VM: Soporte 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 Solución 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/eupf/config.yaml  # Establece: xdp_attach_mode: native
sudo systemctl restart eupf
journalctl -u eupf --since "1 min ago" | grep xdp  # Verifica modo
nativo
```

Verifica que el Modo XDP esté Activo

```
# Verifica la configuración
curl -s http://localhost:8080/api/v1/config | grep xdp_attach_mode
```

```
# Verifica estadísticas
curl -s http://localhost:8080/api/v1/xdp_stats | jq

# Verifica colas
ethtool -l eth0
```

Documentación de la API de OmniUPF

Descripción general

La API de OmniUPF proporciona una interfaz RESTful para gestionar y monitorear la Función de Plano de Usuario basada en eBPF. La API permite el control y la observabilidad en tiempo real de:

- **Sesiones PFCP**: Gestión del ciclo de vida de la sesión y asociación
- **Reglas de Detección de Paquetes (PDR)**: Clasificación del tráfico para enlace ascendente y descendente (IPv4 e IPv6)
- **Reglas de Acción de Reenvío (FAR)**: Acciones de reenvío, almacenamiento en búfer y descarte de paquetes
- **Reglas de Aplicación de QoS (QER)**: Políticas de Calidad de Servicio y limitación de tasa
- **Reglas de Reporte de Uso (URR)**: Seguimiento y reporte del volumen de datos
- **Buffers de Paquetes**: Funcionalidad de almacenamiento en búfer y reproducción de paquetes
- **Estadísticas**: Métricas en tiempo real para paquetes, rutas, XDP y interfaces N3/N6
- **Gestión de Rutas**: Sincronización de rutas de UE con el demonio de enrutamiento FRR
- **Configuración**: Gestión de la configuración de UPF y plano de datos

Documentación de la API de 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 que muestra los puntos finales de la API de OmniUPF con documentación detallada.

Accediendo a la Interfaz de Swagger

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
```

```
## Ver También

- [Documentación de Gestión de Rutas de UE](./routes.md) - Guía detallada sobre la integración de FRR y la sincronización de rutas
- [Guía de Operaciones](../OPERATIONS.md) - Operaciones de la interfaz web y monitoreo
- [Interfaz de Swagger](http://10.98.0.20:8080/swagger/index.html) - Documentación interactiva de la API
```

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

Descripción 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 de red actual.

¿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**: Rastrear 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. Se define la configuración de FRR una vez como una **plantilla Jinja2** en tu playbook de Ansible, y Ansible la propaga automáticamente a todas tus instancias de UPF durante el despliegue.

Una plantilla de configuración Jinja2 típica de 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 kernel
 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**: Establecer variables en tu inventario de Ansible para cada host de UPF
3. **Desplegar en Todas Partes**: Ejecutar el playbook 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 del kernel redistribuir (por ejemplo, `redistribute kernel`)
- **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 básica de FRR se despliega en cada instancia de UPF, el UPF agrega dinámicamente direcciones IP de UE como **rutas de host /32** a la tabla de enrutamiento del kernel basándose en sesiones PFCP activas. Estas rutas son luego:

1. **Instaladas en la tabla de enrutamiento del kernel** por el motor de sincronización de rutas de UPF
2. **Recogidas por FRR** a través de la directiva `redistribute kernel`
3. **Anunciadas a protocolos de enrutamiento** (OSPF, BGP) de acuerdo a tu configuración de FRR
4. **Propagadas a la red** para que el tráfico de UE pueda ser enrutado a esta instancia de UPF

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 protocolo de enrutamiento (áreas OSPF, vecinos BGP, autenticación, políticas de ruta, etc.)
- **UPF maneja rutas dinámicas**: Cada instancia de UPF gestiona dinámicamente solo las rutas de IP de UE /32 basándose en sus sesiones PFCP activas
- **Anuncio automático de rutas**: FRR en cada UPF redistribuye automáticamente las rutas locales de UE de acuerdo a tus políticas configuradas
- **Gestión centralizada**: Actualiza la plantilla de Ansible y vuelve a ejecutar el playbook 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 la 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
- **Pares BGP**: Estado de la sesión BGP y estadísticas de prefijos (cuando está configurado)
- **Rutas Redistribuidas OSPF**: Vista completa de LSAs externas mostrando cómo se anuncian las rutas de UE

La página de Rutas proporciona 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

Beneficios

- **Provisionamiento Sin Contacto**: 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 de sesión
- **Escalabilidad**: Soporta miles de rutas de UE concurrentes

- **Resiliencia**: Las operaciones de sincronización fallidas se rastrean y pueden reintentarse
- **Visibilidad**: Visibilidad completa sobre el 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` - Lista todas las rutas de UE rastreadas sin sincronización
- `POST /api/v1/routes/sync` - Sincroniza rutas con FRR y devuelve la lista actualizada
- `GET /api/v1/route_stats` - Obtiene estadísticas de enrutamiento detalladas
- `GET /api/v1/routing/sessions` - Obtiene sesiones de protocolo de enrutamiento (vecinos OSPF, pares BGP)
- `GET /api/v1/ospf/database/external` - Obtiene la base de datos LSA externa de 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 direcciones IP simples (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

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:

- **Network LSA (10.98.0.20)**: El anuncio de red propio del UPF
- **Router LSA (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