Arquitectura de OmniTWAG

Resumen

OmniTWAG es un Gateway de Acceso WLAN de Confianza (TWAG) según la Sección 16 de 3GPP TS 23.402. Conecta puntos de acceso WLAN de confianza al Núcleo de Paquete Evolucionado (EPC), proporcionando autenticación, gestión de sesiones y túnel de plano de datos.

Tabla de Contenidos

- Arquitectura de Componentes y Procesos
- Árbol de Supervisión
- Referencia de Componentes
- Modos de Operación
- Integración de AP (Túnel GRE)
- Comportamiento de DHCP
- Autenticación
- Ciclo de Vida de la Sesión (Enrutado a Casa)
- Interfaces de Diámetro
- Resumen de Configuración
- Derivación de Claves
- Páginas Relacionadas

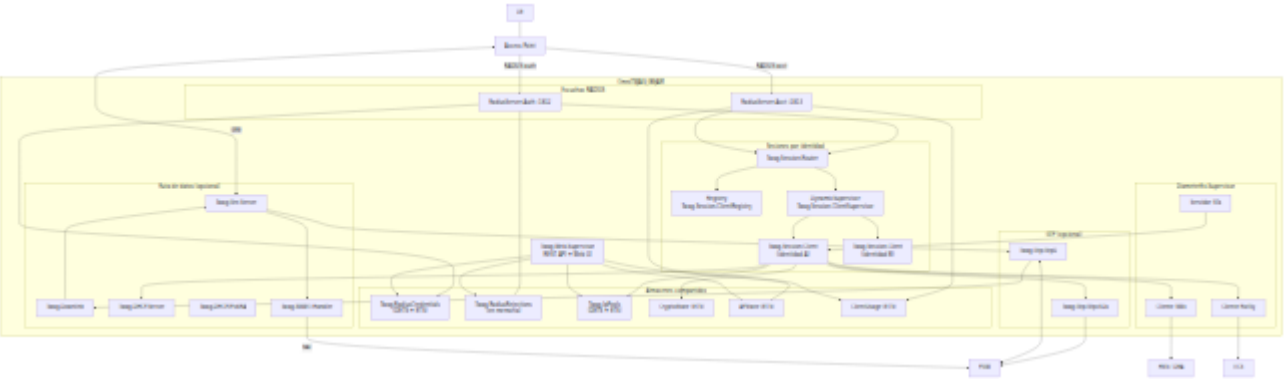
Arquitectura de Componentes y Procesos

OmniTWAG se ejecuta como una única aplicación BEAM (Erlang VM). El supervisor de nivel superior inicia un conjunto fijo de servicios compartidos y,

según la configuración, un conjunto de hijos de ruta de datos e interfaz opcionales. El estado por suscriptor vive en un GenServer para cada identidad, creado bajo demanda.

Los grupos principales son:

- **Web y OAM**: la API REST y la interfaz web (`Twag.Web.Supervisor`).
- **Sesiones por identidad**: un GenServer `Twag.Session.Client` para cada suscriptor, encontrado a través de un `Registry` y comenzado por un `DynamicSupervisor`. `Twag.Session.Router` realiza la búsqueda o creación.
- **Almacenes compartidos**: `Twag.RadiusCredentials` (secretos compartidos duraderos), `Twag.RadiusRejections` (fuentes rechazadas en memoria), `Twag.IpPools` (piscinas de direcciones), y los almacenes respaldados por ETS `CryptoState`, `APState` y `ClientUsage`.
- **Escuchas RADIUS**: `RadiusServer.Auth` (puerto 1812) y `RadiusServer.Acct` (puerto 1813).
- **Diámetro**: `DiameterEx.Supervisor` lleva los pares SWx (vectores de autenticación), STa (re-autenticación y aborto), y Ro/Gy (cobro en línea).
- **GTP**: `Twag.Gtp.GtpcS2a` (control) y `Twag.Gtp.GtpU` (plano de usuario), iniciados solo cuando `gtp.enabled` es verdadero.
- **Ruta de datos**: `Twag.Gre.Server`, `Twag.Downlink`, `Twag.DHCP.Server`, `Twag.DHCP.IPv6RA`, y `Twag.NSWO.Handler`, cada uno iniciado solo cuando su bloque de configuración está habilitado.



Árbol de Supervisión

El supervisor raíz utiliza una estrategia `:one_for_one` (máx. 10 reinicios en 60 segundos). Los hijos se inician en este orden, de modo que los secretos por IP

de origen se resuelvan antes de que llegue el primer paquete RADIUS:

Los hijos marcados con `*` se inician solo cuando su bloque de configuración está habilitado (ver Resumen de Configuración).

Referencia de Componentes

Componente	Tipo	Iniciado cuando
`Twag.Web.Supervisor`	Supervisor	Siempre
`DiameterEx.Supervisor`	Supervisor	Siempre
`Twag.Session.ClientRegistry`	Registro	Siempre
`Twag.Session.ClientSupervisor`	DynamicSupervisor	Siempre
`Twag.Session.Router`	Módulo	Bajo demanda
`Twag.Session.Client`	GenServer	Por identidad
`Twag.RadiusCredentials`	GenServer	Siempre

Componente	Tipo	Iniciado cuando
`Twag.RadiusRejections`	GenServer	Siempre
`Twag.IpPools`	GenServer	Siempre
`RadiusServer.Auth`	GenServer	Siempre
`RadiusServer.Acct`	GenServer	Siempre
`CryptoState`	GenServer (ETS)	Siempre
`APState`	GenServer (ETS)	Siempre

Componente	Tipo	Iniciado cuando
`ClientUsage`	GenServer (ETS)	Siempre
`Twag.Gtp.GtpcS2a`	GenServer	`gtp.enabled`
`Twag.Gtp.GtpU`	GenServer	`gtp.enabled`
`Twag.Downlink`	GenServer	`downlink.enabled`
`Twag.Gre.Server`	GenServer	`gre.enabled`
`Twag.DHCP.Server`	GenServer	`dhcp.enabled`
`Twag.DHCP.IPv6RA`	GenServer	`ipv6_ra.enabled`

Componente	Tipo	Iniciado cuando
`Twag.NSWO.Handler`	GenServer	`nswo.enabled`

Modos de Operación

OmniTWAG soporta dos modos de plano de datos. El modo utilizado para una sesión se resuelve a partir de las credenciales del AP coincidente (ver Secretos Compartidos RADIUS), con la configuración global `gtp` / `nswo` como predeterminada.

Modo Enrutado a Casa (Túnel GTP)

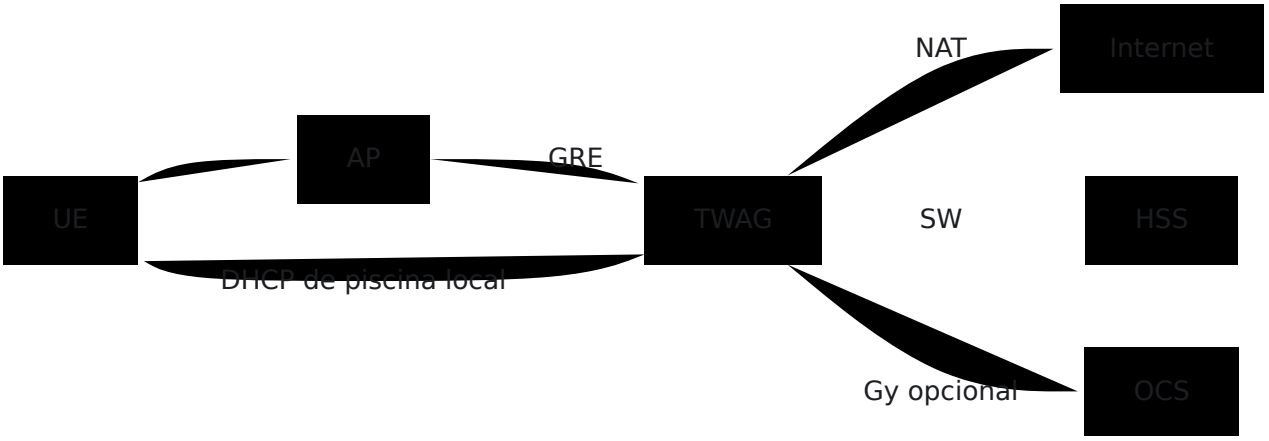
El tráfico se tunela a través de GTP-C/GTP-U (interfaz S2a) hacia el PGW. El PGW asigna la dirección IP del UE, aplica políticas/cobro y proporciona acceso a internet.

- **IP del UE**: Asignada por el PGW en la Respuesta de Crear Sesión (IE PAA).
- **Entrega de IP**: El TWAG proporciona DHCP al UE a través de GRE, entregando la IP asignada por el PGW.
- **Uplink**: El AP envía el tráfico del UE a través de GRE. El TWAG encapsula en GTP-U hacia el PGW.
- **Downlink**: El PGW envía a través de GTP-U. El TWAG desencapsula y envía a través de GRE al AP, luego al UE.
- **Cobro**: Manejado por el PGW (PCEF) a través de Gy hacia el OCS.
- **Config**: `gtp.enabled: true`, `gre.enabled: true`.

Modo de Ruptura Local (NSWO)

El tráfico sale directamente del TWAG hacia internet. No se involucra al PGW. El TWAG realiza NAT/mascarada y opcionalmente realiza su propio cobro en línea

a través de Gy.



- **IP del UE**: Asignada de una piscina de IP local del TWAG (`Twag.IpPools`).
- **Entrega de IP**: El TWAG proporciona DHCP al UE a través de GRE, desde la piscina.
- **Uplink**: El AP envía a través de GRE. El TWAG aplica NAT hacia internet.
- **Downlink**: La ruta de internet regresa a través de NAT del TWAG, luego GRE al AP, luego al UE.
- **Cobro**: El TWAG actúa como PCEF a través de Gy hacia el OCS (opcional, deshabilitado por defecto). Ver Cobro en Línea.
- **Config**: `nswo.enabled: true`, `gtp.enabled: false`.

La Ruptura Local se selecciona configurando `nswo.enabled: true` (con `gtp.enabled: false`), o por AP mediante la credencial `breakout_type`. Cuando está habilitado, se inicia el manejador NSWO, las sesiones reciben una dirección de una piscina local en lugar de una asignada por el PGW, y el uplink se NAT/mascara directamente a internet. Ver Piano de Usuario y Ruta de Datos para el detalle de la ruta de datos.

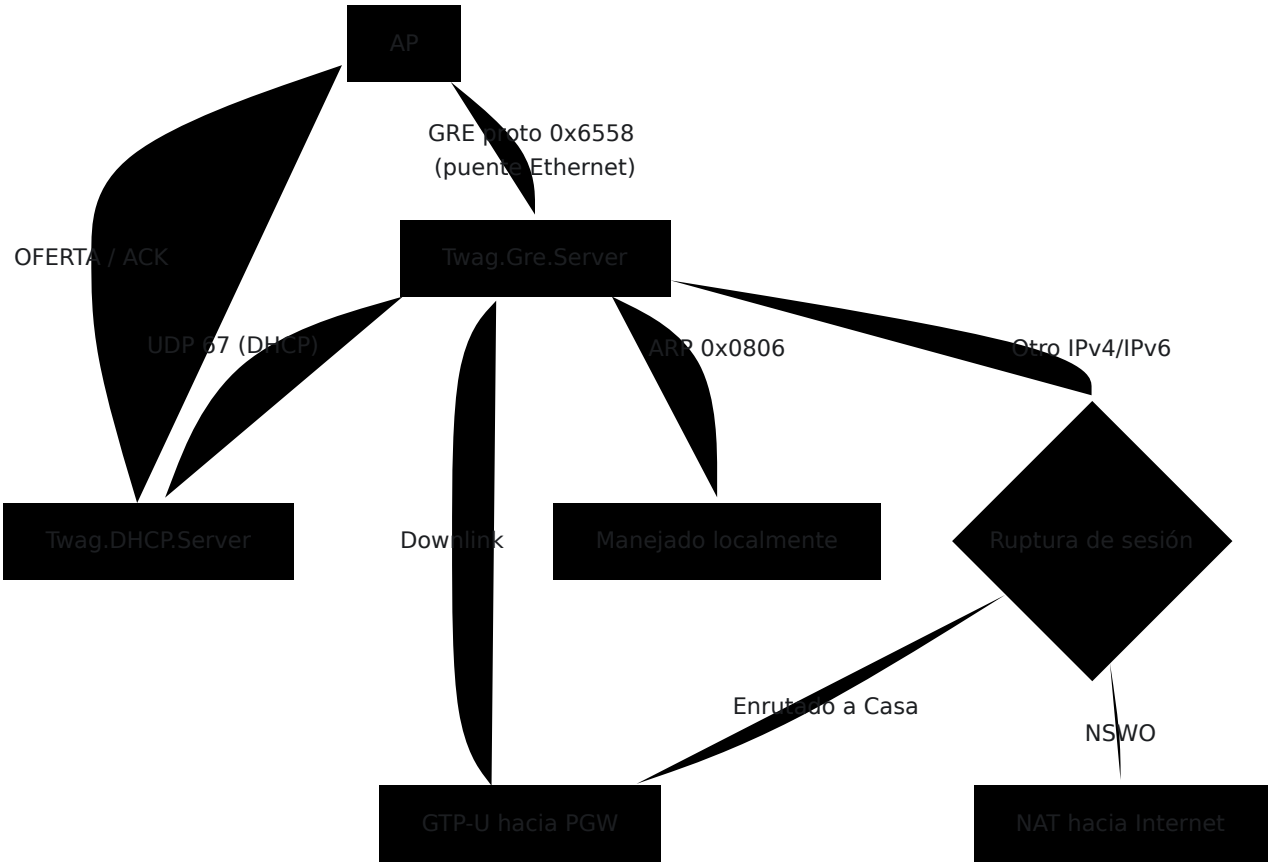
Integración de AP (Túnel GRE)

El AP tunela los tramas L2 del UE al TWAG a través de GRE (protocolo IP 47). Esto proporciona el enlace punto a punto requerido entre el UE y el TWAG según la Sección 16.1.2 de TS 23.402.

Configuración de AP Cisco

```
dot11 ssid ons
   tunnel CWAG
   authentication open eap eap_methods
   authentication key-management wpa
   accounting acct_methods

dot11 tunnel CWAG
    type gre
    source {AP_IP}
    destination {TWAG_IP}
    mss 1410
    mtu 1500
```

Flujo de Datos GRE



Comportamiento de DHCP

Modo	Fuente IP	Servidor DHCP	Notas
Enrutado a Casa	PGW (PAA en Respuesta de Crear Sesión)	TWAG pre-asigna la IP del PGW, la proporciona a través de DHCP sobre GRE	DNS del PGW PCO
Ruptura Local	Piscina local del TWAG (`Twag.IpPools`)	TWAG asigna de la piscina vinculada en DISCOVER (distribución de piscina)	DNS de la configuración de piscina

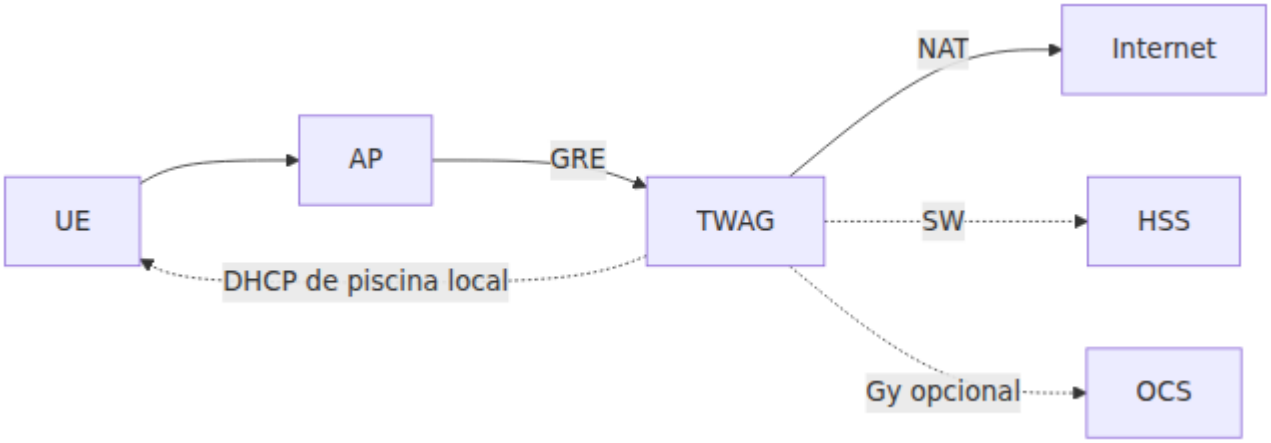
El servidor DHCP del TWAG:

1. Recibe un DHCP DISCOVER del UE a través del túnel GRE.
2. Proporciona la OFERTA. En modo enrutado a casa, esta es la IP asignada por el PGW que la sesión pre-asignó. En modo de ruptura local, cuando la MAC está vinculada a una piscina, el servidor entrega una dirección fresca de la piscina.
3. Maneja REQUEST/ACK para completar el intercambio.
4. Pasa todo el tráfico DHCP a través del túnel GRE hacia y desde el AP.

Autenticación

OmniTWAG actúa como un TWAG combinado y un Servidor AAA de 3GPP, realizando autenticación EAP-AKA localmente y obteniendo vectores de autenticación del HSS a través de Diámetro SWx. La máquina de estados EAP de cada suscriptor se ejecuta dentro de su propio `Twag.Session.Client`.

Flujo de Autenticación



Antes de que se procese cualquier paquete, el escuchador RADIUS autoriza la IP de origen contra `Twag.RadiusCredentials`. Una fuente sin subred de credenciales coincidente es rechazada y registrada como `unauthorized_source`. Una fuente que coincide pero falla la verificación de Message-Authenticator se registra como `bad_secret`. Ver Rechazos RADIUS.

Modos de Autenticación

Modo	Config	Fuente de Vectores	Procesamiento EAP
`:swx`	`auth_mode: :swx`	HSS a través de SWx MAR/MAA	Local (TWAG)
`:local`	`auth_mode: :local`	Claves de prueba locales (Milenage)	Local (TWAG)

Método EAP

Configurable a través de `eap_method`. Predeterminado `:eap_aka`. Si el UE NAKs el método ofrecido, el TWAG negocia con el método preferido soportado por el UE. EAP-SIM y re-autenticación EAP (ERP) también son soportados. Ver Autenticación.

Ciclo de Vida de la Sesión

(Enrutado a Casa)

Autenticación EAP -> Acceso-Aceptar (MSK al AP)

El AP envía Contabilidad-Iniciar

Post-aceptación en Twag.Session.Client

Solicitud de Crear Sesión GTP-C -> PGW

PGW asigna IP, Respuesta de Crear Sesión

Registrar túnel GTP-U (TEIDs locales + remotos)

Pre-asignar IP del PGW en DHCP para MAC del UE

Descubrimiento DHCP del UE -> OFERTA con IP del PGW

Flujos de datos: UE -> GRE -> GTP-U -> PGW -> Internet

Desmantelamiento: GTP-C Eliminar Sesión, limpieza de GTP-U + DHCP

Una sesión se desmantela por una de varias razones, cada una registrada en la métrica `twag.session.teardown`: `accounting_stop`, `admin_disconnect`, `network_disconnect` (STa ASR), `identity_timeout` (autenticación iniciada pero nunca completada), y `liveness_timeout`. Ver Referencia de Métricas y Sesiones.

Interfaces de Diámetro

Interfaz	ID de Aplicación	Dirección	Propósito
SWx	16777265	TWAG -> HSS	Obtención de vectores de autenticación (MAR/MAA)
STa	16777250	Entrante al TWAG	RAR (re-autenticación), ASR (aborto), STR (terminar)
Ro/Gy	4	TWAG -> OCS	Cobro en línea (solo NSWO)

Enrutamiento DRA

El DRA debe enrutar SWx al HSS:

```
%{rule_name: "swx_to_hss", match: :all,
  filters: [{:application_id, 16777265}, {:avp, {296, "
{realm}"}}],
  route: %{peers: ["{hss_host}"]}}
```

Resumen de Configuración

Cada parámetro está documentado en la tabla que sigue al bloque.

```
config :omnitwag,
  auth_mode: :swx,
  eap_method: :eap_aka,
  identity_timeout_ms: 600_000,

  gtp: %{
    enabled: true,
    pgw_ip: "10.179.1.21",
    local_ip: "10.179.1.41",
    pgw_port: 2123,
    gtpu_port: 2152,
    echo_enabled: true
  },

  gre: %{
    enabled: true,
    local_ip: "10.179.1.41"
  },

  downlink: %{
    enabled: true
  },

  dhcp: %{
    enabled: true,
    interface: "gre"
  },

  ipv6_ra: %{
    enabled: false
  },

  nswo: %{
    enabled: false,
    local_pool: "10.100.0.0/16",
    nat_interface: "eth0"
  },

  online_charging: %{
    mode: :disabled
  },

  rate_limit: %{
```

```
    max_failures: 10,
    window_ms: 60_000
  },

  prometheus: %{
    port: 9568
  }
```

Parámetro	Predeterminado	Descripci
`auth_mode`	`:swx`	`:swx` obtiene vectores del H `:local` utiliza claves de pru locales de Milenage.
`eap_method`	`:eap_aka`	Método EAP ofrecido (`:eap` o `:eap_aka_p`
`identity_timeout_ms`	`600000`	Tiempo que u sesión puede permanecer e fase de identi antes de ser desmantelada (razón de desmantelam `identity_tim`
`gtp.enabled`	`true`	Iniciar los hijo C/GTP-U (mod enrutado a ca
`gtp.pgw_ip`	`127.0.0.1`	Dirección del para GTP-C/G
`gtp.local_ip`	ninguno	Dirección loca TWAG para G
`gtp.pgw_port`	`2123`	Puerto GTP-C PGW.
`gtp.gtpu_port`	`2152`	Puerto GTP-U

Parámetro	Predeterminado	Descripci
`gtp.echo_enabled`	`true`	Enviar ecos G keepalives al
`gre.enabled`	`false`	Iniciar el serv GRE para túne de AP.
`gre.local_ip`	`0.0.0.0`	Dirección loca que se vincula servidor GRE.
`downlink.enabled`	`true`	Iniciar el reen downlink.
`dhcp.enabled`	`false`	Iniciar el serv DHCPv4.
`dhcp.interface`	ninguno	`"gre"` para m túnel, o un no de NIC para e directa.
`ipv6_ra.enabled`	`false`	Iniciar Anunci Router IPv6 y DHCPv6.
`nswo.enabled`	`false`	Iniciar el man de ruptura loc NSWO.
`nswo.local_pool`	ninguno	CIDR de respa para direccionamie de ruptura loc

Parámetro	Predeterminado	Descripci
`nswo.nat_interface`	ninguno	NIC de salida NAT/mascarad
`online_charging.mode`	`:disabled`	Modo Gy: `:disabled`, `:authorize_c` `:online_char`
`rate_limit.max_failures`	`10`	Fallos de autenticación NAS en la ven antes de que descarten las Solicitudes de Acceso.
`rate_limit.window_ms`	`60000`	Ventana de lím de tasa en milisegundos.
`prometheus.port`	`9568`	Puerto de mét de Prometheu
`radius_config.acct_interim_interval`	`300`	Intervalo de contabilidad interina (segu anunciado al A Acceso-Acept

Derivación de Claves

Para EAP-AKA' (según el Anexo A.2 de TS 33.402 y RFC 5448):

```
HSS proporciona: RAND, AUTN, XRES, CK, IK
TWAG deriva: CK' = KDF(CK||IK, FC=0x20, network_name, SQN(+)AK)
              IK' = (segunda mitad de la salida de KDF)
              MK  = PRF'(IK'||CK', "EAP-AKA'" || Identity)
              MSK, EMSK, K_aut de MK
```

Para EAP-AKA (RFC 4187):

```
HSS proporciona: RAND, AUTN, XRES, CK, IK
TWAG deriva: MK = SHA1(Identity || IK || CK)
              K_encr, K_aut, MSK, EMSK de PRF(MK)
```

Formato del nombre de red según TS 24.302:
`WLAN:mnc{MNC}.mcc{MCC}.3gppnetwork.org`

Páginas Relacionadas

- Autenticación
- Secretos Compartidos RADIUS
- Rechazos RADIUS
- Piscinas IP
- Sesiones
- Contabilidad por AP
- Piano de Usuario y Ruta de Datos
- Cobro en Línea
- Referencia de Métricas

Resumen

Autenticación (EAP-AKA / EAP-AKA' / EAP-SIM)

Descripción general

OmniTWAG autentica a cada suscriptor con un método EAP que se ejecuta sobre RADIUS entre el punto de acceso y el TWAG. El TWAG actúa como un servidor AAA combinado de TWAG y 3GPP. Ejecuta la máquina de estados EAP y obtiene los vectores de autenticación del HSS a través de Diameter SWx, o de claves de prueba locales. Al tener éxito, el TWAG entrega la Clave Maestra de Sesión (MSK) al punto de acceso como claves MPPE dentro del RADIUS Access-Accept. El punto de acceso luego completa el apretón de manos de 4 vías 802.11 con el suscriptor.

Para el contexto más amplio del sistema, consulte Arquitectura. Para cómo se resuelve el secreto compartido de RADIUS por punto de acceso, consulte Secretos Compartidos de RADIUS. La autenticación se ejecuta primero. La política de separación y carga por AP se aplica después de que la autenticación tenga éxito. Para la sesión autenticada y su ciclo de vida de estado, consulte Sesiones. Para la sesión GTP-C S2a que el TWAG crea después de la autenticación, consulte S2a GTP-C. Para la carga que sigue a una autenticación exitosa, consulte Carga en Línea (Gy).

Tabla de Contenidos

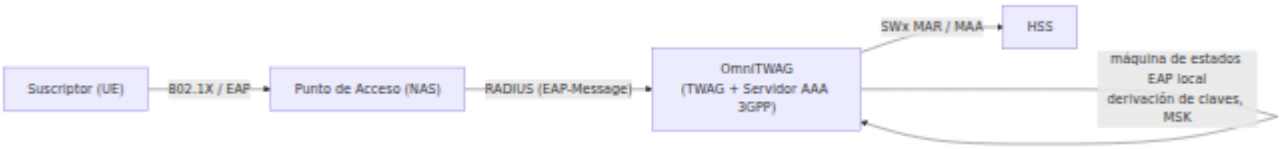
- Descripción general
- Métodos EAP Soportados
- Máquina de Estados EAP y Fases
- Secuencia de Autenticación
- Origen de Vectores de Autenticación
- Manejo de Identidad y NAI

- Derivación y Entrega de Claves
- Resincronización
- Reautenticación Rápida (ERP)
- Manejo de Fallos y Códigos de Resultado
- Configuración
- Observabilidad
- Solución de Problemas

Descripción general

El punto de acceso inicia un intercambio EAP IEEE 802.1X con el suscriptor. El punto de acceso envuelve cada mensaje EAP en un atributo RADIUS `EAP-Message` y lo envía al TWAG. El TWAG ejecuta una máquina de estados de autenticación por cada identidad de suscriptor. La máquina de estados impulsa el intercambio EAP hacia un resultado, luego devuelve uno de tres resultados a la capa RADIUS.

Resultado	Respuesta RADIUS	Significado
Desafío	`Access-Challenge`	El intercambio necesita otra ronda EAP.
Aceptar	`Access-Accept`	La autenticación tuvo éxito. La respuesta lleva la MSK como claves MPPE.
Rechazar	`Access-Reject`	La autenticación falló. La respuesta lleva un EAP-Failure.



El TWAG mantiene el estado criptográfico por identidad. Guarda la respuesta esperada (XRES), la clave de autenticación (`K_aut`), el RAND, la MSK y la MSK

extendida (EMSK) entre rondas EAP. Después de que el TWAG entrega las claves MPPE, el TWAG anula el material clave sensible.

Métodos EAP Soportados

El TWAG soporta tres métodos EAP. Cada método utiliza el mismo formato de atributo TLV, pero un valor de tipo EAP diferente y una jerarquía de claves diferente.

Método	Tipo EAP	Especificación	Vectores	K_aut	MAC	De d
EAP-AKA'	50 (`0x32`)	RFC 5448	quinteto UMTS	32 bytes	HMAC-SHA-256, truncado a 16 bytes	`CK'` lueg (HM 256
EAP-AKA	23 (`0x17`)	RFC 4187	quinteto UMTS	16 bytes	HMAC-SHA1, truncado a 16 bytes	`MK` `SHA` `| I` lueg 186
EAP-SIM	18 (`0x12`)	RFC 4186	2 o 3 tripletas GSM	16 bytes	HMAC-SHA1, truncado a 16 bytes	`MK` `SHA` `| K` `NON` `ver` lueg 186

Selección y negociación de método

El TWAG ofrece un método predeterminado. Establezca el método con la clave de configuración `eap_method`. El predeterminado es `:eap_aka_prime` (EAP-AKA').

El suscriptor puede rechazar el método ofrecido con un EAP-NAK (tipo EAP 3). El NAK enumera los tipos EAP que el suscriptor prefiere. El TWAG lee la lista y selecciona el primer tipo que soporta. El TWAG soporta estos tipos en el NAK: `50` (EAP-AKA'), `23` (EAP-AKA) y `18` (EAP-SIM).

- Si el NAK enumera un tipo soportado, el TWAG reinicia el intercambio con ese método. Para EAP-AKA' y EAP-AKA, el TWAG envía una solicitud de identidad AKA con `AT_PERMANENT_ID_REQ`. Para EAP-SIM, el TWAG envía una solicitud EAP-SIM/Start.
- Si el NAK no enumera ningún tipo soportado, el TWAG rechaza al suscriptor.

Máquina de Estados EAP y Fases

El TWAG ejecuta una máquina de estados por cada identidad de suscriptor, dentro de un proceso de cliente por identidad. La máquina de estados rastrea una fase EAP. La fase impulsa qué mensaje EAP espera el TWAG a continuación y qué telemetría emite.

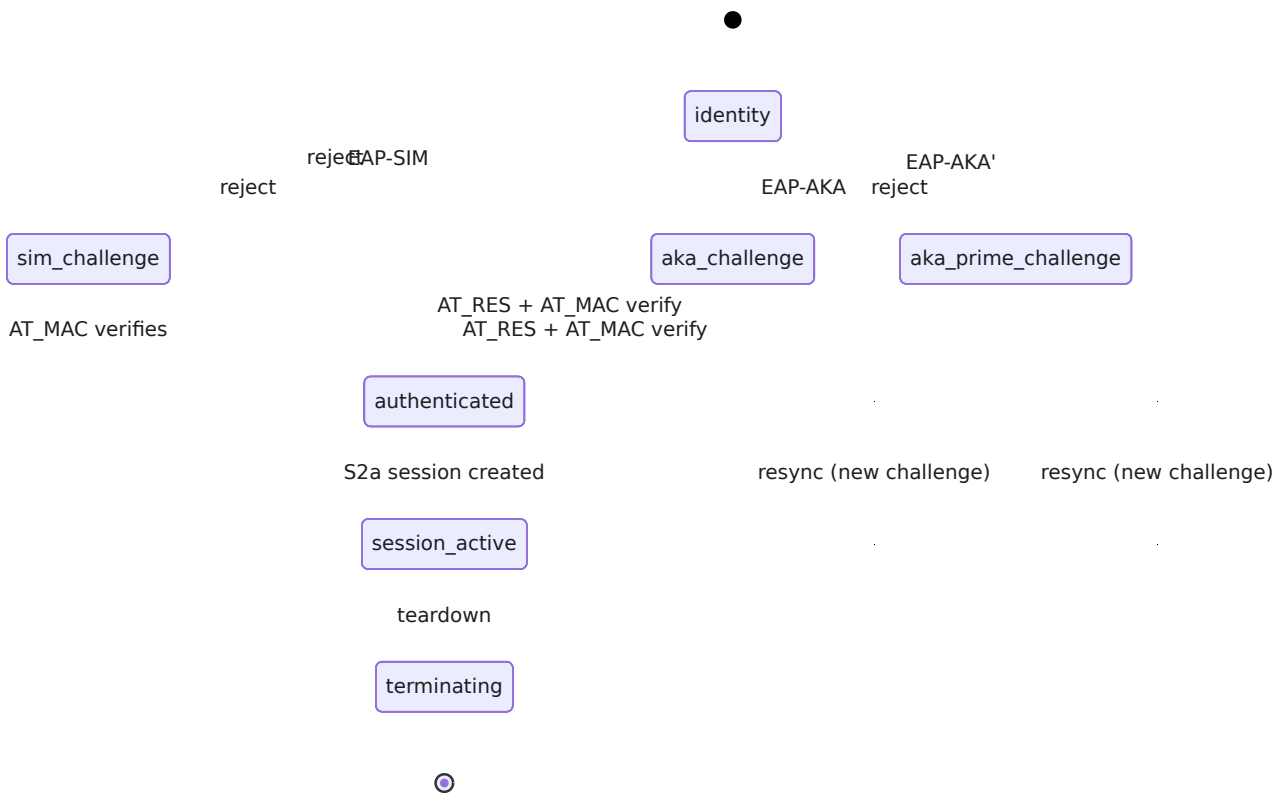
Fase (`eap_phase`)	Significado
`identity`	La fase inicial. El TWAG espera un EAP-Identity, o la primera ronda de AKA-Identity o EAP-SIM/Start. Una ronda fallida devuelve el estado a `identity`.
`aka_challenge`	El TWAG envió un desafío EAP-AKA y espera la respuesta de AKA-Challenge.
`aka_prime_challenge`	El TWAG envió un desafío EAP-AKA' y espera la respuesta de AKA-Challenge.
`sim_challenge`	El TWAG envió un desafío EAP-SIM y espera la respuesta SIM/Challenge.
`authenticated`	El intercambio EAP tuvo éxito. El TWAG tiene la MSK y está a punto de establecer la sesión S2a.
`session_active`	La sesión S2a está activa. El suscriptor puede pasar datos de usuario. La contabilidad RADIUS también puede promover una sesión a esta fase.

Existen dos valores de fase adicionales en el código:

- `resyncing` está reservado como una fase de autenticación en progreso. En la práctica, una ronda de resincronización reconstruye el desafío y devuelve la fase a `aka_challenge` o `aka_prime_challenge`, por lo que el estado no se asienta en `resyncing`.
- `terminating` marca una sesión que se está desmantelando.

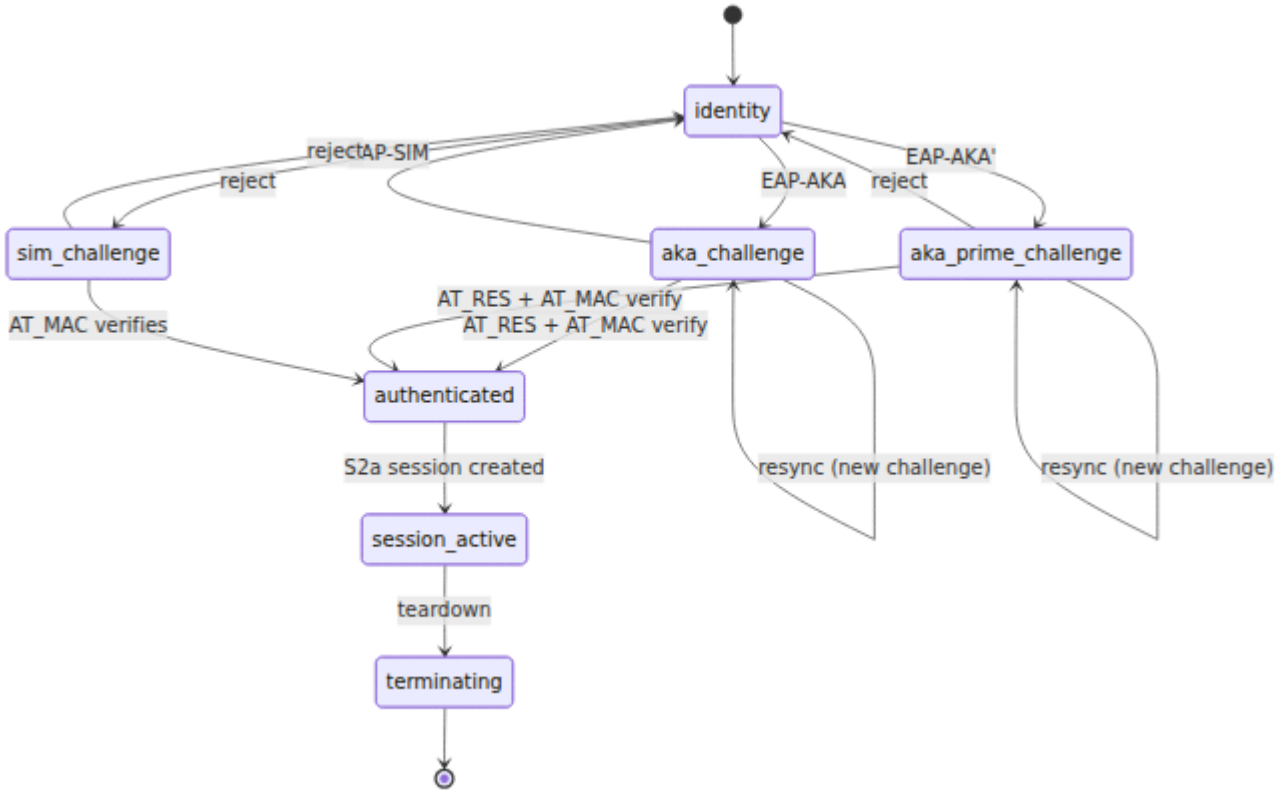
Una sesión que permanece en una fase de autenticación en progreso (`identity`, `aka_challenge`, `aka_prime_challenge`, `sim_challenge`, o `resyncing`) más allá del tiempo de espera de identidad es eliminada. Establezca el tiempo de espera con `identity_timeout_ms`. Consulte Configuración.

La página Sesiones mapea estas fases al estado de autenticación de la sesión mostrado en la interfaz de usuario y la API.



Secuencia de Autenticación

La secuencia completa tiene una fase de identidad y una fase de desafío. El ejemplo a continuación muestra EAP-AKA'. EAP-AKA tiene la misma forma. EAP-SIM utiliza una ronda de inicio en lugar de la ronda de AKA-Identity, y no envía `AT_RES`.



Fase de Identidad

1. El suscriptor envía una respuesta EAP-Identity que lleva el Identificador de Acceso a la Red (NAI).
2. Para EAP-AKA' o EAP-AKA, el TWAG responde con una solicitud de identidad AKA que lleva `AT_PERMANENT_ID_REQ`. Esto solicita al suscriptor la identidad permanente.
3. Para EAP-SIM, el TWAG responde con una solicitud EAP-SIM/Start que lleva la `AT_VERSION_LIST` (solo versión 1).
4. El suscriptor devuelve la identidad permanente (AKA-Identity con `AT_IDENTITY`), o la respuesta de inicio con `AT_NONCE_MT` para EAP-SIM.

Fase de Desafío

1. El TWAG extrae el IMSI de la identidad y obtiene los vectores de autenticación. Consulte Origen de Vectores de Autenticación.
2. El TWAG deriva las claves de sesión, incluida la MSK. Consulte Derivación y Entrega de Claves.
3. El TWAG construye el desafío. Para EAP-AKA' el desafío lleva `AT_RAND`, `AT_AUTN`, `AT_KDF_INPUT` (el nombre de la red), `AT_KDF`, `AT_BIDDING`,

`AT_RESULT_IND`, y `AT_MAC`. Para EAP-AKA el desafío lleva `AT_RAND`, `AT_AUTN`, y `AT_MAC`. Para EAP-SIM el desafío lleva `AT_RAND` (2 o 3 RANDs) y `AT_MAC`.

4. El suscriptor verifica la red (AUTN) y el MAC, luego devuelve la respuesta.
5. El TWAG verifica la respuesta. Consulte Verificación de Respuesta.
6. Con éxito, el TWAG devuelve un EAP-Success y la MSK. En caso de fallo, el TWAG devuelve un EAP-Failure.

Verificación de Respuesta

El TWAG verifica la respuesta al desafío en un orden fijo. El orden es el mismo para EAP-AKA y EAP-AKA'.

1. **Verificación de estado.** Si falta `K_aut` o XRES almacenado, el TWAG rechaza.
2. **Verificación de atributo.** Si falta `AT_RES` o `AT_MAC`, el TWAG rechaza.
3. **Verificación de MAC.** El TWAG anula el campo `AT_MAC`, recomputa el MAC sobre el paquete con `K_aut`, y compara. EAP-AKA utiliza HMAC-SHA1. EAP-AKA' utiliza HMAC-SHA-256. Ambos se truncan a 16 bytes.
4. **Verificación de respuesta.** El TWAG compara `AT_RES` con el XRES almacenado. La comparación es **en tiempo constante** para prevenir ataques de temporización. El TWAG utiliza una verificación de longitud y una comparación de bytes en tiempo constante.
5. **Resultado.** Si tanto el MAC como la respuesta coinciden, el TWAG acepta. De lo contrario, el TWAG rechaza.

Para EAP-SIM, el suscriptor no envía `AT_RES`. La verificación de `AT_MAC` es la única prueba de que el suscriptor posee las claves correctas. Si el MAC verifica, el TWAG acepta.

Origen de Vectores de Autenticación

El TWAG obtiene los vectores de autenticación de una de dos fuentes. Establezca la fuente con la clave de configuración `auth_mode`.

Modo	`auth_mode`	Fuente de vector	Uso
SWx	`:swx`	HSS, a través de Diameter SWx (MAR/MAA)	Producción
Local	cualquier valor diferente de `:swx` (predeterminado `:local`)	Milenage local de claves de prueba provisionadas	Pruebas

En ambos modos, el TWAG ejecuta la máquina de estados EAP localmente. Solo difiere la fuente del vector. Cualquier valor de `auth_mode` que no sea `:swx` selecciona la ruta de Milenage local.

> **El paso a través de STa es una ruta separada.** Un proveedor `Twag.Auth.StaAuth` puede retransmitir cargas útiles EAP en bruto a un servidor AAA 3GPP a través de Diameter STa (DER/DEA), por lo que el servidor AAA ejecuta el intercambio EAP-AKA y devuelve la MSK. Este proveedor existe pero no está en la ruta de autenticación RADIUS en vivo. La ruta en vivo obtiene vectores con `:swx` o `:local` y ejecuta la máquina de estados EAP en el TWAG.

SWx al HSS

En modo `:swx`, el TWAG envía una Solicitud de Autenticación Multimedia (MAR) al HSS a través de la interfaz SWx (3GPP TS 29.273, ID de Aplicación `16777265`). El MAR lleva:

- `User-Name`: el IMSI.
- `SIP-Auth-Data-Item` con `SIP-Authentication-Scheme` establecido en `EAP-AKA`. Para una resincronización, el ítem también lleva el `SIP-Authorization` (la información de resync: RAND concatenado con AUTS).
- `SIP-Number-Auth-Items`: 1.
- `ANID`: `WLAN`.
- `RAT-Type`: 0.
- `Auth-Session-State`: 1 (NO_STATE_MAINTAINED).

- `Vendor-Specific-Application-Id` con el SWx Auth-Application-Id `16777265` y Vendor-Id `10415`.

El HSS devuelve una Respuesta de Autenticación Multimedia (MAA). Cuando el `Result-Code` es `2001` (éxito), el TWAG lee el primer `SIP-Auth-Data-Item` y extrae el vector:

- `SIP-Authenticate` = RAND (16 bytes) concatenado con AUTN (16 bytes).
- `SIP-Authorization` = XRES.
- `Confidentiality-Key` = CK.
- `Integrity-Key` = IK.

Si el `Result-Code` no es `2001`, o los campos del vector están ausentes, el TWAG falla la obtención y lo trata como una denegación de autenticación (no como un fallo interno). La fase de desafío rechaza al suscriptor de manera limpia con un EAP-Failure.

Señalización de errores SWx (Experimental-Result)

SWx señala la mayoría de los errores en un AVP `Experimental-Result` agrupado (Vendor-Id `10415`), no en el `Result-Code` base. Un MAA fallido puede, por lo tanto, no llevar ningún `Result-Code` base en absoluto. El TWAG lee también el `Experimental-Result-Code`, y registra el nombre legible por humanos para que el operador vea la verdadera causa. Consulte Solución de Problemas para los códigos.

Un caso común es `DIAMETER_ERROR_USER_UNKNOWN` (5001) o `DIAMETER_ERROR_USER_NO_NON_3GPP_SUBSCRIPTION` (5450): el suscriptor no está habilitado para acceso no 3GPP en SWx, por lo que la autenticación falla aunque el mismo suscriptor puede ser conocido en S6a para acceso EPC.

El DRA debe enrutar la aplicación SWx al HSS. Consulte Arquitectura: Enrutamiento DRA.

Milenage Local

En modo `:local`, el TWAG genera el vector con el algoritmo Milenage (3GPP TS 35.206) a partir de las claves de prueba provisionadas (`test_keys`). Las claves son `ki`, `opc`, `amf`, y `sqn`. El TWAG computa RAND, AUTN, XRES, CK, e IK en el proceso. Use este modo solo para pruebas en laboratorio.

Vectores para EAP-SIM

EAP-SIM necesita 2 o 3 tripletas GSM. El TWAG deriva las tripletas de los mismos vectores de Milenage. Para cada vector, el TWAG toma RAND, toma los primeros 4 bytes de XRES como SRES, y deriva el `Kc` GSM de CK e IK con la función de conversión c3 (3GPP TS 33.102 Anexo C.3): `Kc = CK[0..7] XOR CK[8..15] XOR IK[0..7] XOR IK[8..15]`.

Manejo de Identidad y NAI

La identidad del suscriptor es un Identificador de Acceso a la Red (NAI). El NAI tiene la forma `{username}@{realm}`, por ejemplo `0{IMSI}@wlan.mnc{MNC}.mcc{MCC}.3gppnetwork.org`.

Extracción de IMSI

El TWAG extrae el IMSI del NAI en dos pasos:

1. El TWAG toma el nombre de usuario, que es la parte antes del `@`.
2. El TWAG elimina un dígito de tipo de identidad inicial. Un `0` inicial (identidad permanente, EAP-AKA/AKA') o un `6` inicial se mapean al IMSI puro. Los dígitos restantes son el IMSI.

El TWAG acepta un IMSI de 14 o 15 dígitos. El IMSI es el `User-Name` en el SWx MAR. El TWAG también utiliza el IMSI para derivar el MCC y el MNC para la sesión GTP-C: el MCC son los primeros 3 dígitos, y el MNC son los siguientes 2 dígitos.

Nombre de red (EAP-AKA')

EAP-AKA' vincula las claves al nombre de la red. El nombre de la red va en `AT_KDF_INPUT`, y es una entrada para la derivación de `CK'`/`IK'`. Establezca el nombre de la red con la clave de configuración `network_name`. Si la clave no está establecida, el TWAG deriva el nombre de la red del dominio Diameter según 3GPP TS 24.302:

- El TWAG elimina un `epc.` inicial del dominio.
- El TWAG antepone `WLAN:`.

Por ejemplo, el dominio `epc.mnc{MNC}.mcc{MCC}.3gppnetwork.org` da el nombre de red `WLAN:mnc{MNC}.mcc{MCC}.3gppnetwork.org`. Si el dominio no tiene el prefijo `epc.`, el TWAG utiliza `WLAN`.

Derivación y Entrega de Claves

Cada método deriva una Clave Maestra de Sesión (MSK) de 64 bytes y una MSK Extendida (EMSK) de 64 bytes. El TWAG entrega la MSK al punto de acceso. El TWAG mantiene la EMSK para una reautenticación rápida.

Jerarquía de claves EAP-AKA'

Según RFC 5448 y 3GPP TS 33.402 Anexo A:

1. `Key = CK \| IK`.
2. `S = FC(0x20) \| network_name \| len(network_name) \| (SQN XOR AK) \| len(SQN XOR AK)`.
3. `CK' \| IK' = HMAC-SHA-256(Key, S)`. `CK'` son los primeros 16 bytes. `IK'` son los últimos 16 bytes.
4. `MK = PRF'(IK' \| CK', "EAP-AKA'" \| Identity)`, 208 bytes. PRF' utiliza HMAC-SHA-256.
5. El TWAG divide `MK`: `K_encr` (16), `K_aut` (32), `K_re` (32), `MSK` (64), `EMSK` (64).

Jerarquía de claves EAP-AKA

Según RFC 4187:

1. `MK = SHA1(Identity \| IK \| CK)`.
2. El TWAG expande `MK` a 160 bytes con el FIPS 186-2 PRF.
3. El TWAG divide la salida: `K_encr` (16), `K_aut` (16), `MSK` (64), `EMSK` (64).

Jerarquía de claves EAP-SIM

Según RFC 4186:

1. `MK = SHA1(Identity \| Kc1 \| Kc2 \| [Kc3] \| NONCE_MT \| Version_List \| Selected_Version)`.
2. El TWAG expande `MK` a 160 bytes con el FIPS 186-2 PRF.
3. El TWAG divide la salida: `K_encr` (16), `K_aut` (16), `MSK` (64), `EMSK` (64).

MSK a MPPE al punto de acceso

El TWAG entrega la MSK de 64 bytes al punto de acceso en el RADIUS Access-Accept. El TWAG divide la MSK y codifica cada mitad como una clave MPPE de Microsoft (RFC 2548, RFC 3748):

- `MS-MPPE-Recv-Key` = `MSK[0..31]` (primeros 32 bytes).
- `MS-MPPE-Send-Key` = `MSK[32..63]` (últimos 32 bytes).

El TWAG cifra cada clave con el secreto compartido de RADIUS y el Request Authenticator, según RFC 2548. El TWAG utiliza una sal fresca de 2 octetos con el bit alto establecido para cada clave. El TWAG firma el Access-Accept con el secreto compartido. Consulte Secretos Compartidos de RADIUS para cómo el TWAG resuelve el secreto.

El punto de acceso descifra las claves MPPE, recupera la MSK y utiliza la MSK como la Clave Maestra Par a Par para el apretón de manos de 4 vías 802.11 con el suscriptor.

El Access-Accept requiere una MSK válida de 64 bytes. Si la MSK falta o tiene el tamaño incorrecto, el TWAG envía un Access-Reject en su lugar.

Resincronización

El suscriptor puede informar que su número de secuencia (SQN) está fuera de sincronización con la red. El suscriptor envía un AKA-Synchronization-Failure (subtipo `0x04`) con `AT_AUTS`. `AT_AUTS` tiene 14 bytes: 6 bytes de SQN oculto y 8 bytes de MAC-S.

El TWAG maneja la resincronización de la siguiente manera:

1. El TWAG construye la información de resincronización: el RAND almacenado concatenado con el AUTS de 14 bytes.
2. El TWAG ejecuta la resincronización. Esto recupera el SQN del suscriptor con la función Milenage f5*, verifica MAC-S, y devuelve un número de secuencia actualizado.
3. Con éxito, el TWAG genera un nuevo vector con el SQN actualizado, construye un nuevo desafío y envía un nuevo Access-Challenge.
4. En caso de fallo, el TWAG rechaza al suscriptor con un EAP-Failure.

El TWAG maneja la resincronización tanto para EAP-AKA como para EAP-AKA'. La resincronización utiliza las claves Milenage locales. El valor `AT_AUTS` debe ser exactamente de 14 bytes, de lo contrario, el TWAG genera un error y la ronda falla.

Reautenticación Rápida (ERP)

El TWAG soporta el Protocolo de Reautenticación EAP (ERP) según RFC 6696. ERP permite a un suscriptor reautenticarse sin un intercambio EAP completo. El ahorro proviene de una cadena de claves que el TWAG deriva de la EMSK durante la primera autenticación completa.

Después de una autenticación completa, el TWAG deriva y almacena las claves de reautenticación:

- `rRK` = Clave Raíz de Reautenticación, de la EMSK.
- `rIK` = Clave de Integridad de Reautenticación, de la rRK.

Todas las derivaciones de claves ERP utilizan HMAC-SHA-256.

Cuando el suscriptor envía un mensaje EAP-Initiate/Re-auth (tipo EAP 41), el TWAG hace lo siguiente:

1. Si el TWAG no tiene `rRK` o `rIK` almacenados, el TWAG recurre a una autenticación completa. El TWAG envía una solicitud de identidad AKA fresca.
2. Si el TWAG tiene las claves, el TWAG analiza el mensaje y verifica la etiqueta de autenticación con `rIK`.
3. Si la etiqueta verifica, el TWAG deriva la MSK de reautenticación (`rMSK`) de `rRK` y el número de secuencia del mensaje, construye una respuesta EAP-Finish/Re-auth (tipo EAP 42), y acepta. La `rMSK` se convierte en la nueva MSK para las claves MPPE.
4. Si la etiqueta no verifica, o el análisis falla, el TWAG recurre a una autenticación completa.

Manejo de Fallos y Códigos de

Resultado

Resultados EAP

Condición	Acción del TWAG
Fallo de SWx MAR, o vectores incompletos	La obtención de vectores es denegada. El TWAG rechaza al suscriptor de manera limpia con un EAP-Failure, y registra una denegación en lugar de un fallo.
La verificación de `AT_MAC` falla	Access-Reject con EAP-Failure.
`AT_RES` no coincide con XRES	Access-Reject con EAP-Failure.
Falta `K_aut` o estado XRES	Access-Reject con EAP-Failure.
Falta atributo `AT_RES` o `AT_MAC`	Access-Reject con EAP-Failure.
El suscriptor envía AKA-Authentication-Reject (subtipo `0x02`)	Access-Reject. El suscriptor rechazó la red.
El suscriptor envía AKA-Client-Error (subtipo `0x0E`)	Access-Reject.
EAP-NAK sin método soportado	Access-Reject.
MSK falta o no es de 64 bytes en Aceptar	Access-Reject.

Condición	Acción del TWAG
Mensaje EAP no manejado	Access-Reject con EAP-Failure.

Códigos de resultado Diameter SWx

El código de éxito es un `Result-Code` base. Los códigos de error a continuación llegan en el AVP `Experimental-Result`, no en el `Result-Code` base. Consulte la nota de Señalización de errores SWx y Solución de Problemas.

Código	Nombre	Significa
2001	DIAMETER_SUCCESS	El MAA l un vecto válido.
5001	DIAMETER_ERROR_USER_UNKNOWN	El HSS n tiene suscripc no 3GPP/SW para est IMSI. El suscripto puede se conocido S6a/EPC
5004	DIAMETER_ERROR_NOT_SUPPORTED_USER_DATA	Los dato de usuar solicitad no son soportad
5420	DIAMETER_ERROR_UNKNOWN_EPS_SUBSCRIPTION	No hay suscripc EPS para este suscripto
5450	DIAMETER_ERROR_USER_NO_NON_3GPP_SUBSCRIPTION	El suscri no tiene suscripc no 3GPP
Otro	(error Diameter)	La obtenció

Código	Nombre	Significa
		de vecto falla. La ronda rechaza.

Referencia: RFC 6733 Sección 7.1, 3GPP TS 29.273, 3GPP TS 29.229.

Eventos iniciados por la red (STa)

El Servidor AAA 3GPP puede enviar eventos al TWAG a través de la interfaz STa (3GPP TS 29.273, ID de Aplicación `16777250`). El TWAG responde a cada evento con `Result-Code` `2001`.

Mensaje	Disparador	Acción del TWAG
ASR (Abort-Session-Request)	El servidor AAA quiere una desconexión.	El TWAG desconecta la sesión del suscriptor, luego responde ASA.
STR (Session-Termination-Request)	Desconexión iniciada por el HSS.	El TWAG desconecta la sesión del suscriptor, luego responde STA.
RAR (Re-Auth-Request)	El servidor AAA quiere una reautenticación.	El TWAG responde RAA con éxito y registra la solicitud. La reautenticación completa de la sesión en vivo aún no se activa desde RAR.

Las tres respuestas llevan el Origin-Host y Origin-Realm de la identidad Diameter. Consulte Configuración.

Configuración

Establezca los parámetros de autenticación en la configuración de la aplicación `:omnitwag`. La identidad Diameter (`host` y `realm`) proviene de la configuración `:diameter_ex`, y el TWAG la utiliza como Origin-Host y Origin-Realm en SWx y STa.

```
config :omnitwag,
  # Fuente de vectores: :swx (producción) o :local (pruebas)
  auth_mode: :swx,

  # Método EAP ofrecido: :eap_aka_prime (predeterminado),
:eap_aka, o :eap_sim
  eap_method: :eap_aka_prime,

  # Nombre de red EAP-AKA'. Si no se establece, se deriva del
dominio Diameter.
  network_name: "WLAN:mnc001.mcc001.3gppnetwork.org",

  # Recoja una sesión que aún esté atascada en una fase de
autenticación en progreso después de este tiempo en
  # milisegundos. Predeterminado 600000 (10 minutos).
  identity_timeout_ms: 600_000,

  # Campos de Identificador TWAN, añadidos a STa DER/AAR cuando se
establecen
  twan_ssid: "ons",
  twan_bssid: "00:11:22:33:44:55",

  # Claves de prueba locales (utilizadas solo cuando auth_mode:
:local)
  test_keys: %{
    ki: "11111111111111111111111111111111",
    opc: "22222222222222222222222222222222",
    amf: "8000",
    sqn: 1200
  }
```

Parámetro	Tipo	Requerido	Predeterminado	Des
`auth_mode`	Átomo	No	`:local`	Fuente vector obtien del HS de SW genera a parti `test_l` Establ en pro
`eap_method`	Átomo	No	`:eap_aka_prime`	El mét ofrecid `:eap_a` `:eap_a` `:eap_s` suscrip negoci métod soport diferen EAP-NA
`network_name`	Cadena	No	derivado del dominio	El nom EAP-AK en `AT_KDI` Si no s establ TWAG del do Diame TS 24.

Parámetro	Tipo	Requerido	Predeterminado	Des
`identity_timeout_ms`	Entero	No	`600000`	Tiemp milise que ur puede perma una fa autent progre de que la reco
`twan_ssid`	Cadena	No	-	El SSID Cuand establ TWAG Identif TWAN mensa DER y
`twan_bssid`	Cadena	No	-	El BSS punto Combi `twan_s` forman Identif TWAN.
`test_keys`	Mapa	No	valores predeterminados incorporados	Claves locales Utiliza cuand `auth_r` `:local` Consul

Parámetro	Tipo	Requerido	Predeterminado	Des
				Parám claves prueba

Parámetros de claves de prueba

El TWAG lee `test_keys` solo en modo `:local`. No utilice `test_keys` en producción.

Parámetro	Tipo	Requerido	Predeterminado	Descripción
`ki`	Cadena (hex)	Sí (modo local)	`1111...`	Clave del suscriptor `Ki`, 16 bytes como 32 caracteres hexadecimales.
`opc`	Cadena (hex)	Sí (modo local)	`2222...`	Clave variante del operador `OPc`, 16 bytes como 32 caracteres hexadecimales.
`amf`	Cadena (hex)	Sí (modo local)	`8000`	Campo de Gestión de Autenticación, 2 bytes como 4 caracteres hexadecimales.
`sqn`	Entero	Sí (modo local)	`1200`	El número de secuencia inicial.

Observabilidad

El TWAG emite contadores de telemetría a través de la máquina de estados EAP y el controlador STa. El endpoint de Prometheus expone los contadores. Consulte la Referencia de Métricas para el endpoint y ejemplos de consultas.

Métricas EAP

Métrica	Significado
`eap_aka_identity_count`	Intercambios de Identidad EAP-AKA/AKA' (Fase 1).
`eap_aka_challenge_count`	Intercambios de Desafío EAP-AKA/AKA' (Fase 2).
`eap_aka_sync_failure_count`	Fallos de sincronización AKA (eventos de resync).
`eap_aka_auth_success_count`	Autenticaciones EAP-AKA/AKA' exitosas.
`eap_aka_auth_reject_count`	Autenticaciones EAP-AKA/AKA' rechazadas.

El TWAG también emite eventos de telemetría para EAP-SIM (`start`, `auth_success`, `auth_reject`), para AKA `auth_reject` y `client_error`, y para ERP (`initiate`, `success`, `fallback`, `mac_failure`). Consulte la Referencia de Métricas.

Métricas de respuesta RADIUS

Métrica	Significado
`radius_access_challenge_count`	Respuestas Access-Challenge (ronda EAP en progreso).
`radius_access_accept_count`	Respuestas Access-Accept (autenticación exitosa).
`radius_access_reject_count`	Respuestas Access-Reject (autenticación fallida).

Métricas de eventos STa

El controlador STa emite un contador de telemetría para cada evento iniciado por la red: `[:sta, :asr, :received, :count]`, `[:sta, :str, :received, :count]`, y `[:sta, :rar, :received, :count]`, además de contadores de sesión `disconnect` y `reauth`.

Contexto de registro

El TWAG establece los metadatos de registro `identity` e `imsi` para cada sesión. Utilice estos campos para rastrear a un suscriptor a través del flujo de autenticación.

Soluci►►n de Problemas

Síntoma	Causa probable	
La autenticación falla y el registro muestra `DIAMETER_ERROR_USER_UNKNOWN (5001)`	El HSS no tiene suscripción no 3GPP/SWx para este IMSI. El suscriptor puede ser conocido en S6a/EPC pero no está habilitado para acceso no 3GPP.	Habilit no 3GP suscrip
La autenticación falla y el registro muestra `DIAMETER_ERROR_USER_NO_NON_3GPP_SUBSCRIPTION (5450)`	El perfil del suscriptor no tiene suscripción no 3GPP.	Provisi suscrip en el H
El registro muestra `Result-Code=nil` en un MAA fallido	SWx devolvió un error en el AVP `Experimental-Result`, no en el `Result-Code` base.	Lea el `Result` misma registr imprim legible
La autenticación nunca se completa y la sesión es eliminada	El suscriptor se detuvo en una fase de autenticación	Verifiq del inte Ajuste

Síntoma	Causa probable	
	en progreso más allá del tiempo de espera de identidad.	`identi` si es n
El suscriptor envía AKA-Authentication-Reject	El UE no pudo autenticar la red (verificación AUTN).	Verifiqu nombr `OP`/`OPc` coincid en mod que los HSS se en mod
Repetidos AKA-Synchronization-Failure	El SQN del suscriptor se desvió del SQN de la red.	El TWA resincr autom persist manej HSS. C Resinc
Access-Reject con un EAP-Success válido esperado	La MSK faltaba o no era de 64 bytes en Aceptar.	Confirr fuente devolv comple el SWx conjun locales

Para rechazos en la capa RADIUS que ocurren antes o alrededor del intercambio EAP (secreto compartido incorrecto, punto de acceso desconocido), consulte Secretos Compartidos de RADIUS y la referencia de estado de Sesiones.

Flujos de Llamadas

Descripción general

Esta página muestra los flujos de mensajes de extremo a extremo para OmniTWAG. Cada flujo es un diagrama de secuencia entre el equipo del usuario (UE), el punto de acceso WiFi (AP), el TWAG y los pares de la red central: el HSS (SWx), el PGW (S2a) y el OCS (Gy).

Lee las páginas por función para obtener los detalles detrás de cada paso: Autenticación, S2a GTP-C, Plano de Usuario, Gestión de Direcciones IP, Grupos de IP, Carga en Línea, Credenciales RADIUS, y Sesiones.

Tabla de Contenidos

- Actores e interfaces
- Adjunto exitoso - tunelado, cargado
- Adjunto - NSWO con un grupo de IP
- Adjunto - NSWO con distribución DHCP
- Adjunto - salida local (plano de datos del AP)
- Fallo de autenticación - suscriptor no provisionado para SWx
- Contabilidad e informes de uso
- Agotamiento de cuota y re-grant
- Desmantelamiento - desconexión del suscriptor
- Desmantelamiento - expulsión del operador desde la API
- Desmantelamiento - tiempo de espera de actividad
- Variantes de modo de carga

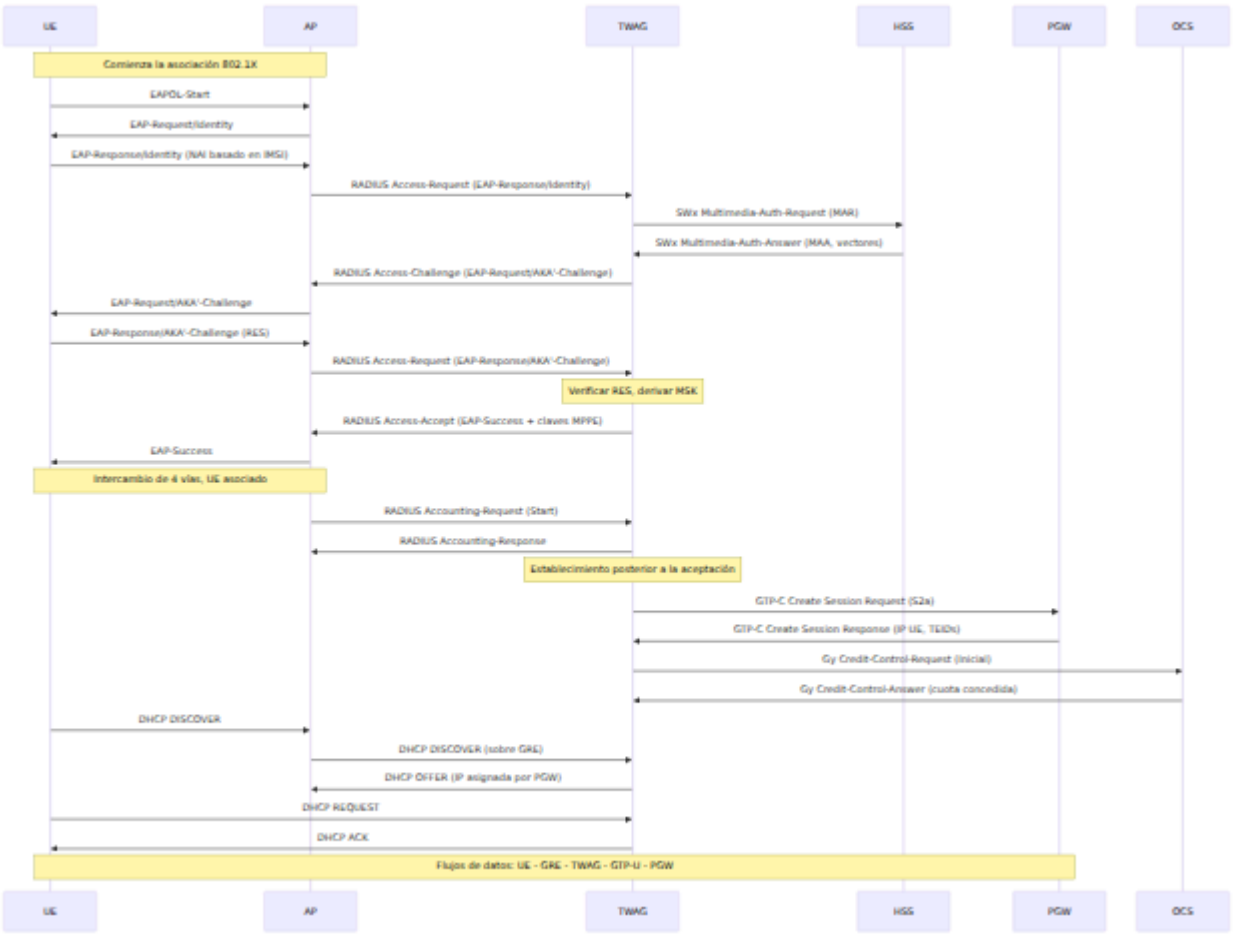
Actores e interfaces

Actor	Rol	Interfaz al TWAG
UE	Dispositivo suscriptor con una SIM	802.11 / EAPOL (a través del AP)
AP	Punto de acceso WiFi y cliente RADIUS	RADIUS (UDP 1812 autenticación, 1813 contabilidad)
TWAG	Puerta de enlace de acceso WiFi de confianza y servidor AAA 3GPP	-
HSS	Servidor de suscriptores local	Diameter SWx (vectores de autenticación)
PGW	Puerta de enlace de paquetes	GTP-C/GTP-U sobre S2a (solo tunelado)
OCS	Sistema de carga en línea	Diameter Gy (solo cargado)

El tipo de salida y el modo de carga de una sesión provienen de las credenciales del AP. Consulta Credenciales RADIUS.

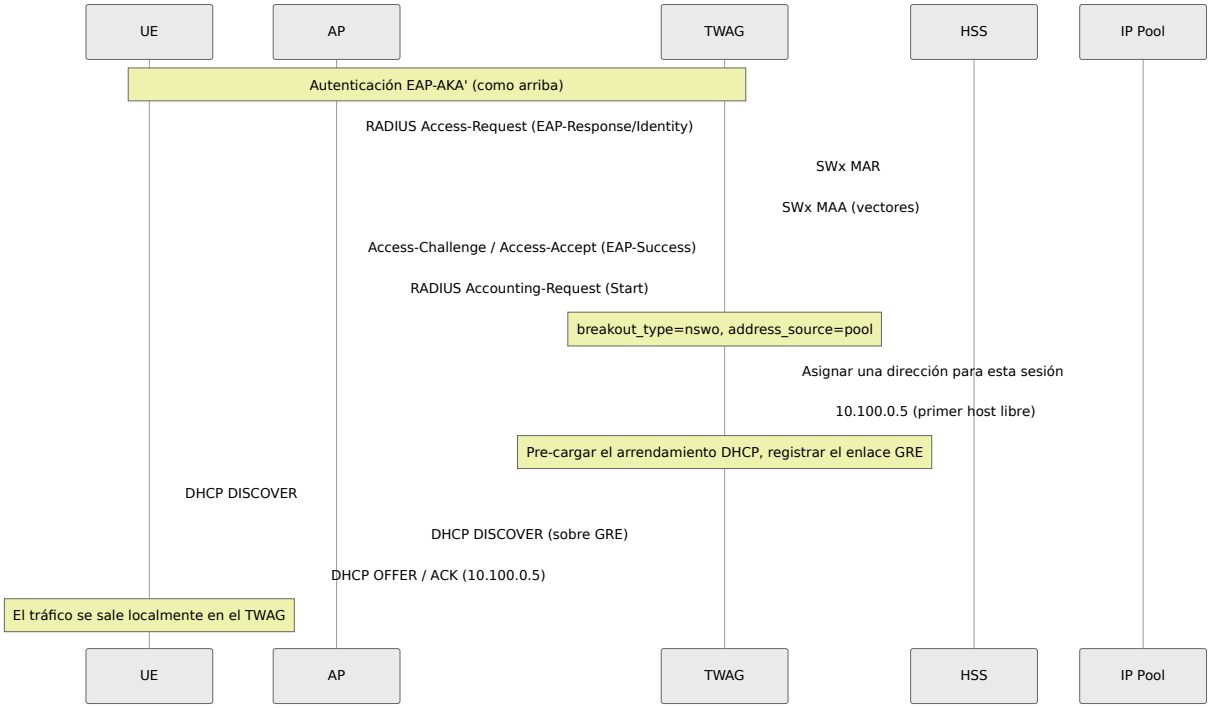
Adjunto exitoso - tunelado, cargado

Este es el camino feliz completo para `breakout_type: tunneled` y `charging_mode: charged`. El TWAG autentica al suscriptor con EAP-AKA', establece una sesión GTP con el PGW, entrega la IP asignada por el PGW a través de DHCP y abre una sesión de carga Gy.



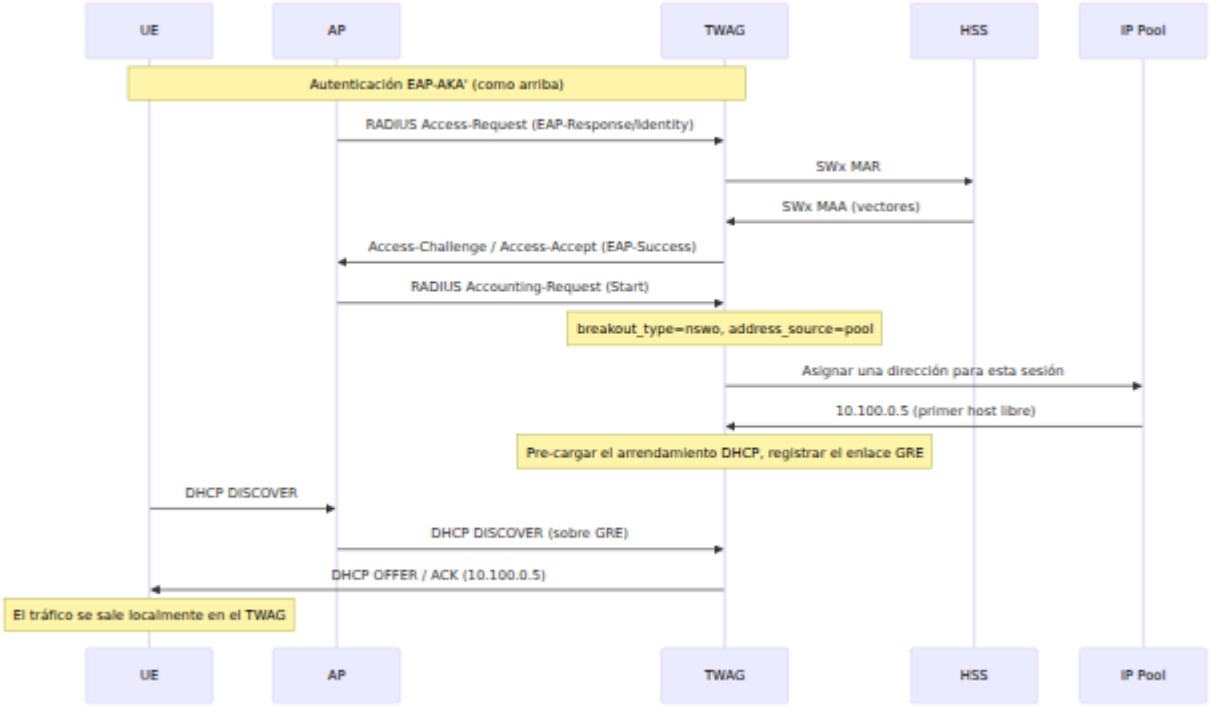
Adjunto - NSWO con un grupo de IP

Para `breakout_type: nswo` y `address_source: pool`, el TWAG no contacta al PGW. El TWAG pre-asigna una dirección fija del grupo de IP asociado y la entrega a través de DHCP. El tráfico del suscriptor se sale localmente.



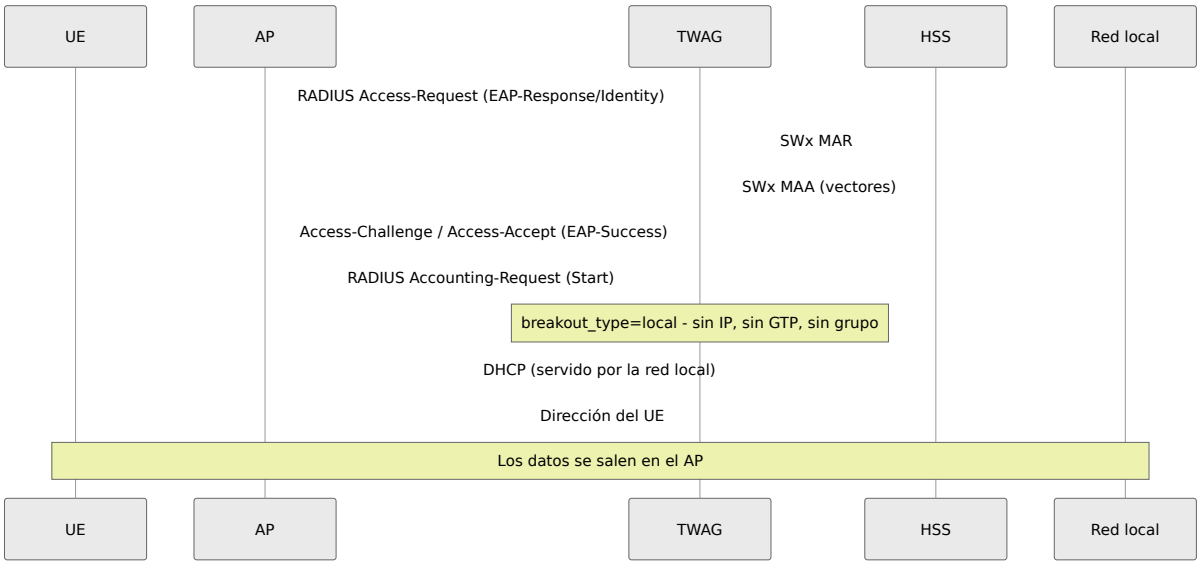
Adjunto - NSWO con distribución DHCP

Para `breakout_type: nswo` y `address_source: dhcp`, el TWAG vincula la MAC del cliente al grupo. El servidor DHCP distribuye la siguiente dirección libre del rango del grupo cuando el UE envía un DISCOVER. Consulta Grupos de IP.



Adjunto - salida local (plano de datos del AP)

Para `breakout_type: local`, el AP posee el plano de datos. El TWAG autentica al suscriptor y no asigna IP. El UE obtiene su dirección de la red local. El TWAG no establece ninguna sesión GTP ni arrendamiento DHCP.



Fallo de autenticación - suscriptor no provisionado para SWx

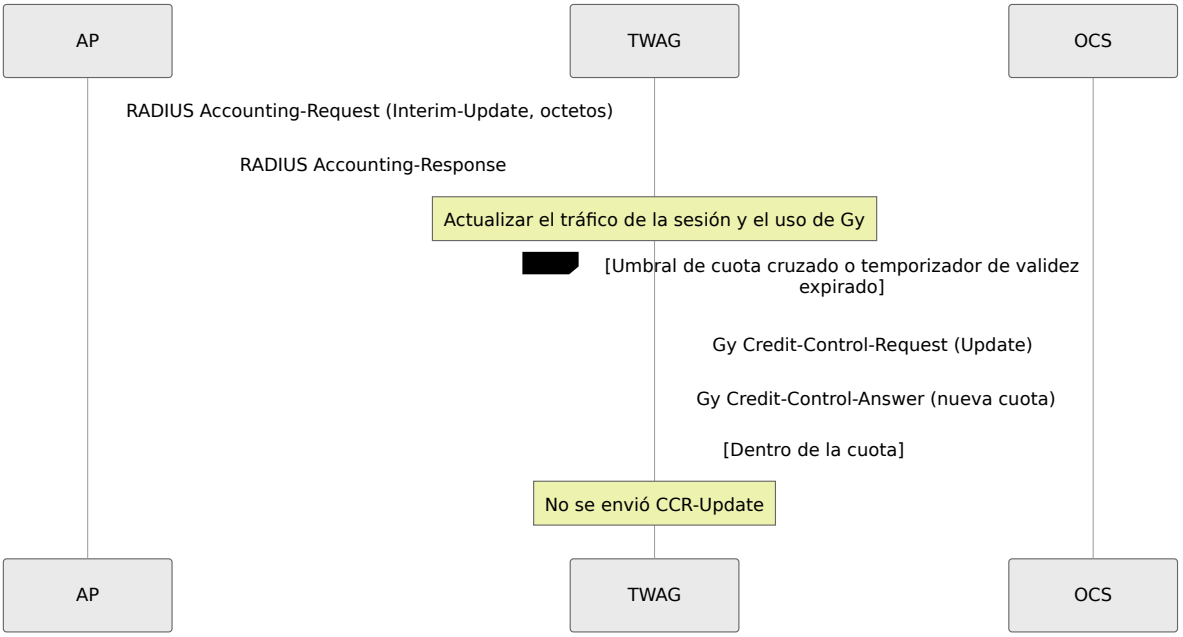
El HSS señala un error de SWx en el AVP agrupado `Experimental-Result`, no en el `Result-Code` base. Un suscriptor conocido en S6a aún puede fallar en SWx si el HSS no tiene una suscripción no 3GPP para el IMSI. El TWAG rechaza la sesión. Consulta Autenticación.

Parse error on line 11: ... Experimental-Result TWAG->>AP: RADI ----------^ Expecting 'SOLID_OPEN_ARROW', 'DOTTED_OPEN_ARROW', 'SOLID_ARROW', 'BIDIRECTIONAL_SOLID_ARROW', 'DOTTED_ARROW', 'BIDIRECTIONAL_DOTTED_ARROW', 'SOLID_CROSS', 'DOTTED_CROSS', 'SOLID_POINT', 'DOTTED_POINT', got 'NEWLINE'

Intente de nuevo

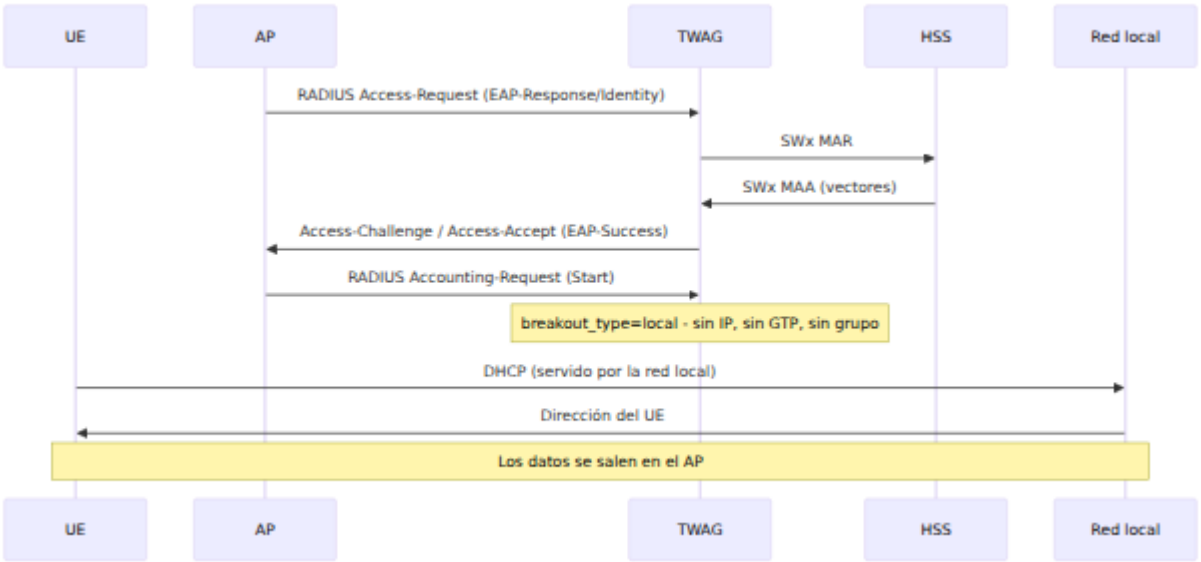
Contabilidad e informes de uso

El AP envía contabilidad RADIUS. El TWAG fusiona el tráfico en la sesión (ver Sesiones) y, en modo `charged`, informa el uso al OCS en un umbral de cuota o una expiración de temporizador de validez, no en cada actualización interina. Consulta Carga en Línea.



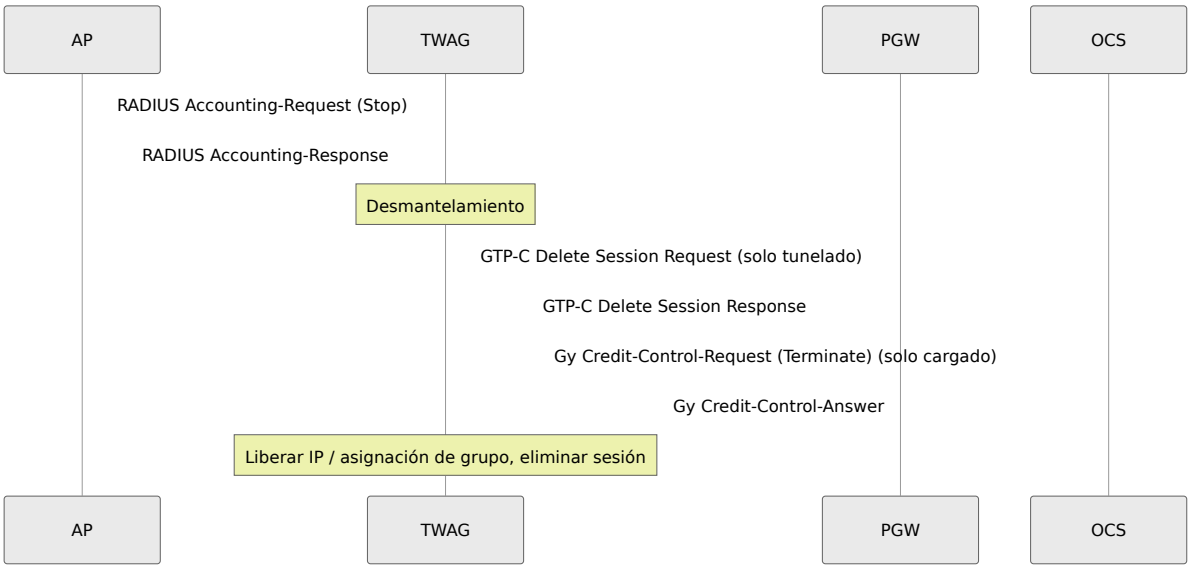
Agotamiento de cuota y re-grant

Cuando el OCS no concede más cuota, o establece una Final-Unit-Action, el TWAG bloquea el camino de datos del UE. El TWAG mantiene la sesión, el arrendamiento y la contabilidad para que un recargo pueda desbloquear al UE. Consulta Carga en Línea.



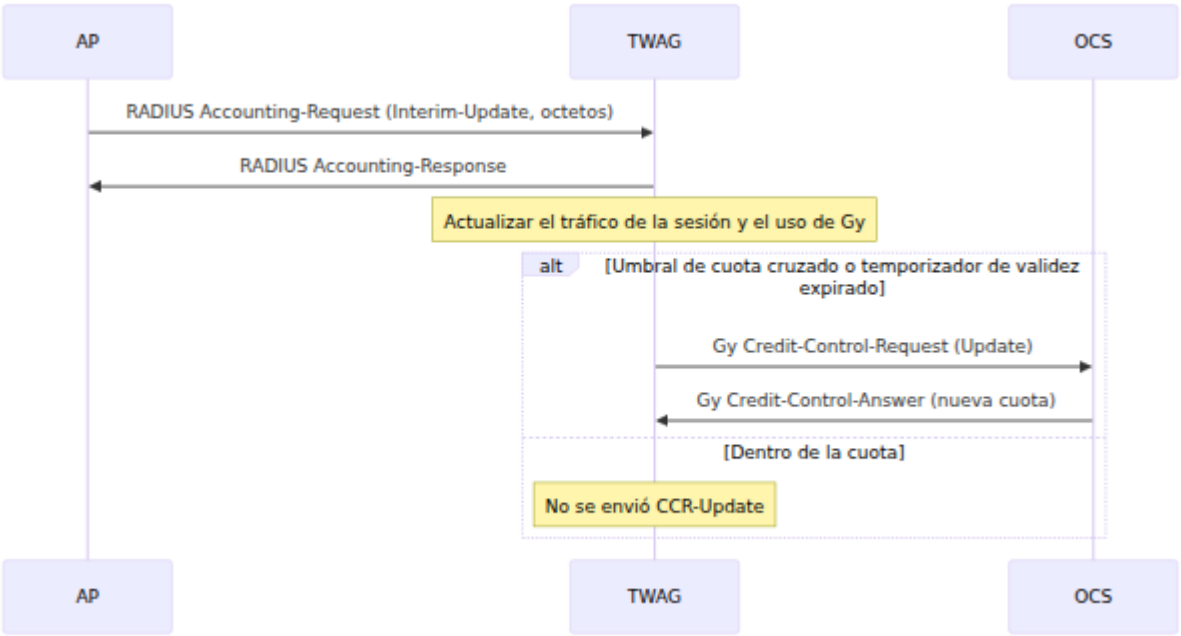
Desmantelamiento - desconexión del suscriptor

Cuando el UE se desconecta, el AP envía un Accounting-Stop. El TWAG desmantela la sesión: elimina la sesión GTP (tunelada), envía un Gy CCR-Terminate (cargado), libera la dirección y borra el estado de la sesión.



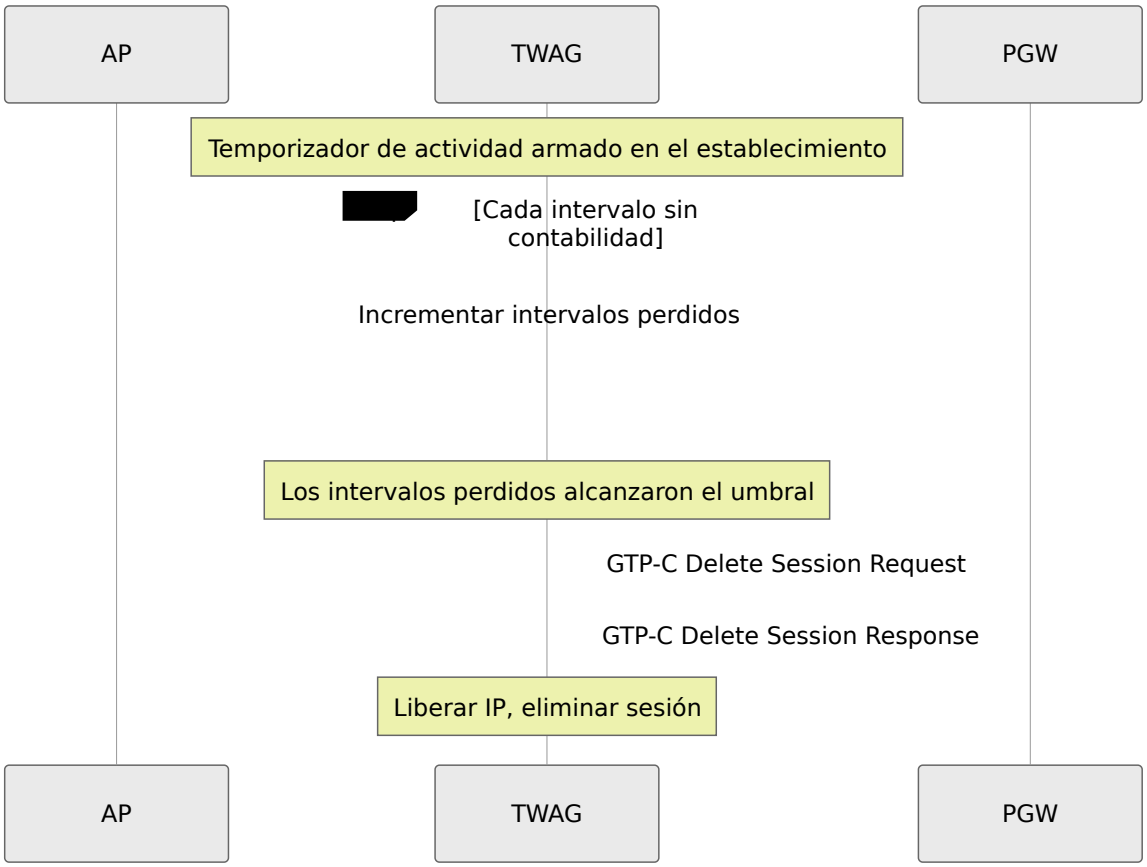
Desmantelamiento - expulsión del operador desde la API

Un operador puede eliminar una sesión a través de la API. El TWAG envía un RADIUS CoA-Disconnect (Disconnect-Message) al AP, que desconecta al UE, y luego desmantela la sesión. Consulta Sesiones.



Desmantelamiento - tiempo de espera de actividad

El TWAG monitorea una sesión establecida con un temporizador de actividad. Las actualizaciones interinas de contabilidad reinician el temporizador. Si el AP no envía actualizaciones durante el número configurado de intervalos, el TWAG declara la sesión muerta y la desmantela. Una sesión que nunca completa la autenticación es eliminada por un tiempo de espera de identidad separado. Consulta Sesiones.

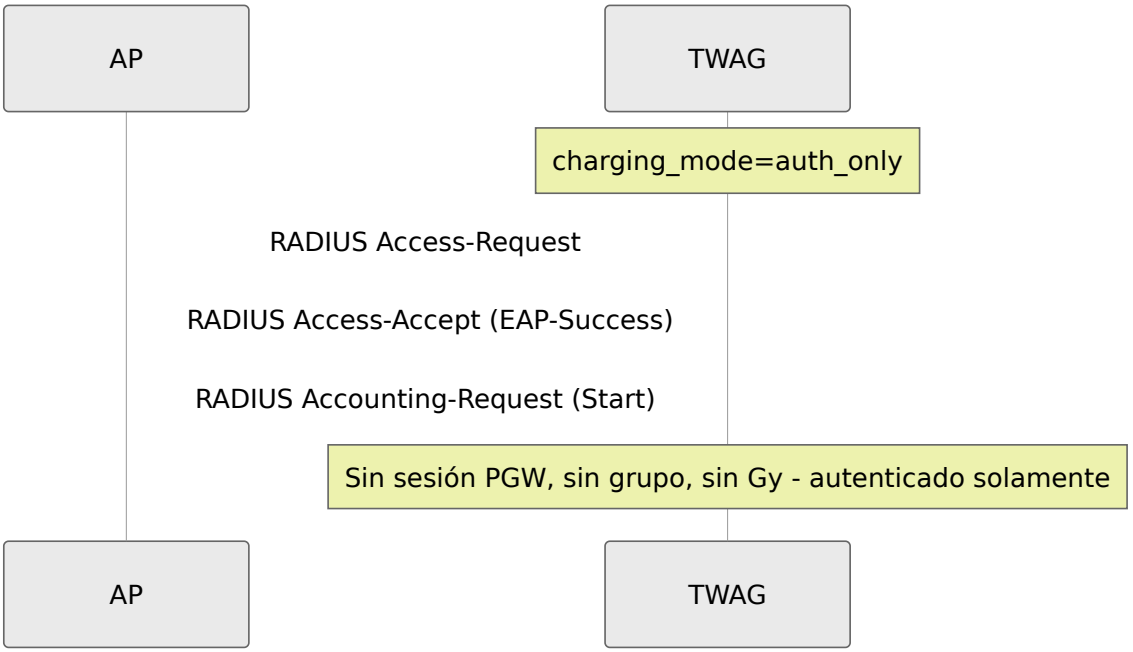


Variantes de modo de carga

El `charging_mode` en las credenciales del AP cambia los pasos posteriores a la aceptación. Consulta Credenciales RADIUS.

charging_mode	Sesión de datos	Gy
`charged`	Establecida (por `breakout_type`)	CCR-Initial / Update / Terminate
`uncharged`	Establecida (por `breakout_type`)	Sin sesión Gy
`auth_only`	No establecida	Sin sesión Gy

Para `uncharged`, el flujo es el adjunto tunelado o NSWO anterior, pero el TWAG no envía mensajes Gy. Para `auth_only`, el flujo se detiene después del Access-Accept y Accounting-Start: el TWAG no establece ninguna sesión de datos.



Descripción general

Modos de Conexión, WLCP y Movilidad

Resumen

Esta guía explica los modos de conexión TWAG, el Protocolo de Control WLAN (WLCP) y el comportamiento de transferencia. Describe lo que el software OmniTWAG hace hoy. También marca cada área que es parcial o aún no está integrada, para que los operadores no planifiquen en torno a un comportamiento que el código no ejecuta.

El TWAG sigue el modelo WLAN de confianza en la sección 16 de 3GPP TS 23.402. Para el plano de datos, autenticación y señalización S2a en los que se basan estos modos, consulte:

- Arquitectura: plano de datos, ciclo de vida de la sesión e interfaces
- Autenticación: EAP-AKA / EAP-AKA' y material clave
- S2a GTP-C: Crear Sesión, Eliminar Sesión y señalización de portadoras
- Sesiones: la sesión autenticada y su ciclo de vida de estado
- Plano de Usuario: entrega de GRE, GTP-U y DHCP
- Secretos Compartidos de RADIUS: separación por AP y cobro aplicado después de la autenticación

Tabla de Contenidos

- Resumen y Comparación de Modos
- Negociación de Modos
- Modo de Conexión Simple Transparente (TSCM)
- Modo de Conexión Simple (SCM)
- Modo de Conexión Múltiple (MCM) y WLCP
- Transferencia y Movilidad
- Configuración
- Observabilidad

- Estado de Implementación

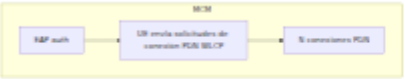
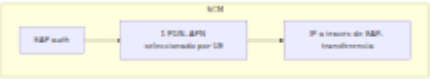
Resumen y Comparación de Modos

El TWAG admite tres modos de conexión para un UE que accede al EPC a través de una WLAN de confianza. El modo controla cómo el UE aprende su dirección IP, si el UE puede señalar un APN y si el UE puede mantener más de una conexión PDN.

Propiedad	TSCM	SCM	MCM
Código de modo	`0`	`1`	`2`
Nombre completo	Modo de Conexión Simple Transparente	Modo de Conexión Simple	Modo de Conexión Múltiple
Conexiones PDN por UE	Una	Una	Muchas
Entrega de dirección IP	DHCP	EAP	EAP / WLCP
Selección de APN señalada por el UE	No	Sí	Sí (por conexión)
Transferencia con preservación de IP	No	Sí	Sí
Portadoras dedicadas	No	Sí	Sí
Protocolo de control del UE	Ninguno	Ninguno	WLCP (TS 24.244)
UE consciente de S2a	No	Sí	Sí

TSCM es el modo predeterminado. TSCM coincide con el comportamiento anterior a la Release-12, donde el UE no señala a la red. El TWAG selecciona el APN, crea una conexión PDN y entrega la dirección IP asignada por el PGW a través de DHCP.

Los códigos de modo `0`, `1` y `2` siguen el AVP de Modo de Conexión TWAN en 3GPP TS 29.273.

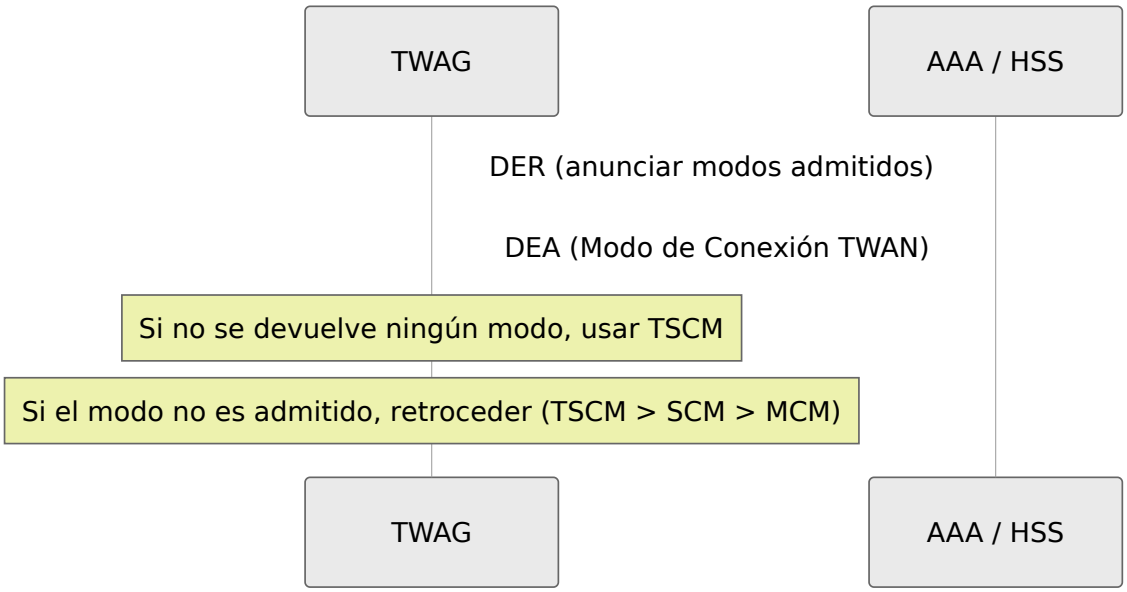



Negociación de Modos

El modo de conexión se negocia durante la autenticación, en el intercambio STa que lleva EAP. Consulte Autenticación para el flujo EAP en sí. El diseño es:

1. El TWAG anuncia los modos que admite al servidor AAA. El conjunto anunciado proviene de la clave de configuración `connection_modes`.
2. El servidor AAA devuelve el modo seleccionado en la respuesta STa DEA, en el AVP de Modo de Conexión TWAN.
3. Si el servidor AAA no indica un modo, el TWAG utiliza TSCM como predeterminado.

Si el servidor AAA selecciona un modo que el TWAG no admite, el TWAG retrocede. El orden de retroceso prefiere TSCM, luego SCM, luego MCM. El TWAG selecciona el primer modo admitido de este orden.



El TWAG lee el modo seleccionado de cualquiera de estas claves en la respuesta AAA, en orden: `twan_connection_mode`, `TWAN-Connection-Mode`, `connection_mode`, o `Connection-Mode`. El valor puede ser un código numérico (`0`/`1`/`2`) o un átomo (`:tscm`/`:scm`/`:mcm`). Un valor no reconocido predetermina a TSCM.

En la ruta STa, el DER STa anuncia el modo preferido admitido por el TWAG en el AVP `TWAN-Connection-Mode`. El análisis DEA STa lee el modo negociado de vuelta y predetermina a TSCM cuando el DEA no lleva ninguno.

> **Estado: el modo negociado no se aplica en la ruta de sesión en vivo.** El cliente STa (`StaClient`) sí anuncia el modo admitido en el DER y lee el modo negociado del DEA. El módulo de modo de conexión está presente y probado unitariamente. Sin embargo, la ruta de autenticación de paso STa no es la ruta de autenticación RADIUS en vivo, y el cliente en vivo nunca aplica el modo negociado: el punto de entrada `store_connection_mode` no tiene un llamador en la ruta de conexión, y la ruta de conexión y datos siempre se comportan como TSCM (IP a través de DHCP, sin WLCP, sin transferencia señalada por el UE o APN). Hoy, cada sesión en vivo sigue la ruta TSCM. Consulte Estado de Implementación.

Modo de Conexión Simple Transparente (TSCM)

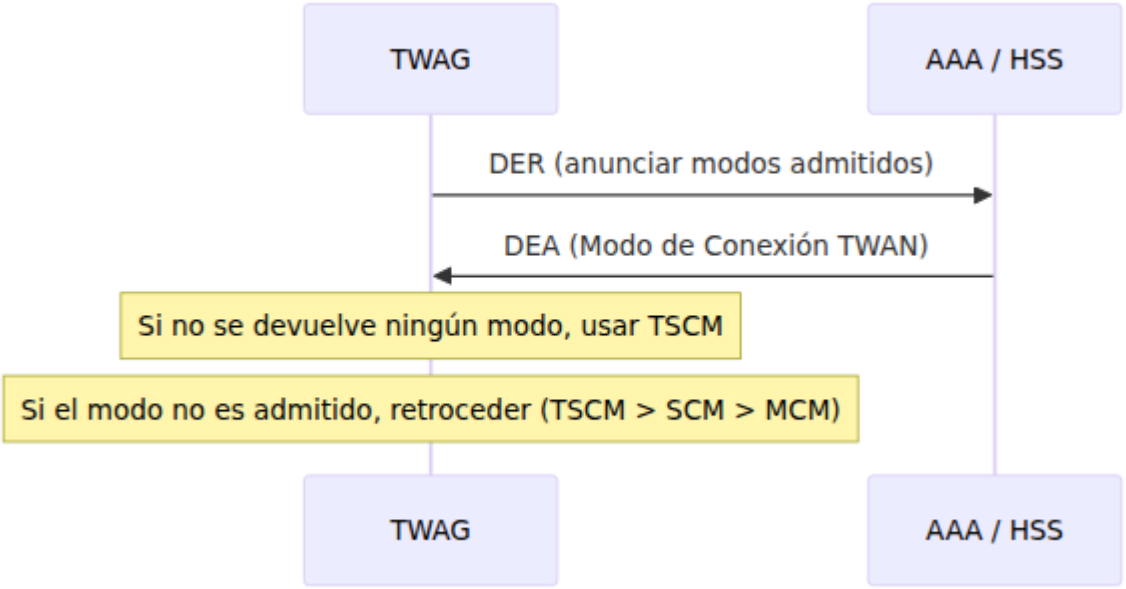
TSCM es el modo predeterminado y el comportamiento que la ruta de sesión en vivo ejecuta hoy. El UE no señala a la red. TSCM se describe en el ciclo de vida de sesión de la Arquitectura y se resume aquí.

En TSCM:

- El UE mantiene una conexión PDN.
- El TWAG selecciona el APN. El UE no puede solicitar un APN.
- El UE recibe su dirección IP a través de DHCP, servido por el TWAG a través del túnel GRE. Consulte Plano de Usuario.
- No hay señalización WLCP.
- La transferencia con preservación de IP, portadoras dedicadas y selección de APN señalada por el UE no están disponibles.

El TWAG crea la sesión S2a automáticamente después de la autenticación. Envía una Solicitud de Crear Sesión al PGW con la bandera de Indicación de Transferencia clara. Luego pre-asigna la dirección IP asignada por el PGW en su servidor DHCP para la dirección MAC del UE. Consulte S2a GTP-C para el detalle

de Crear Sesión y Sesiones para la sesión resultante y su ciclo de vida de estado. El cobro por la sesión, cuando está habilitado, sigue a través de Gy. Consulte Cobro en Línea (Gy).



Modo de Conexión Simple (SCM)

SCM mantiene una sola conexión PDN por UE, pero el UE es consciente de S2a. SCM agrega estas capacidades sobre TSCM:

- El UE recibe su dirección IP a través de EAP, no DHCP.
- El UE puede seleccionar un APN no predeterminado.
- El UE mantiene su dirección IP a través de la transferencia de 3GPP a WLAN y de WLAN a 3GPP.
- El PGW puede crear portadoras dedicadas.

El módulo de modo de conexión informa estas capacidades de SCM. Clasifica SCM como un modo que entrega IP a través de EAP, admite transferencia, admite selección de APN y admite portadoras dedicadas.

> **Estado: SCM aún no es una ruta de ejecución distinta.** El modelo de capacidad para SCM existe, pero el código de establecimiento de sesión aún no ramifica en el modo negociado. Un UE que el servidor AAA coloca en SCM aún se sirve a través de la ruta TSCM (PDN auto-creada, IP a través

> de DHCP). No confíe en la IP entregada por EAP o en la selección de APN señalada por el UE para SCM en esta versión.

Modo de Conexión Múltiple (MCM) y WLCP

MCM permite que un UE mantenga varias conexiones PDN al mismo tiempo. El UE gestiona cada conexión con el Protocolo de Control WLAN (WLCP), definido en 3GPP TS 24.244. Después de la autenticación, el TWAG no auto-crea una conexión PDN. En su lugar, el UE envía Solicitudes de Conexión PDN WLCP. El TWAG mapea cada conexión PDN a una sesión S2a GTP-C separada hacia el PGW.

Transporte WLCP

El TWAG ejecuta un servidor WLCP en el puerto UDP `2156`, el puerto estándar WLCP. El servidor enruta cada paquete a un controlador de sesión WLCP por UE, basado en la identidad del UE. El servidor mapea la IP y el puerto de origen del UE a la identidad después de la autenticación.

En producción, la señalización WLCP se ejecuta sobre DTLS según la sección 16.1.4A.3.1 de TS 23.402. El servidor actual maneja solo la capa UDP simple. DTLS aún no se aplica.

Formato de Mensaje WLCP

Cada mensaje WLCP tiene un encabezado fijo, seguido de una lista de Elementos de Información (IEs).

```
Tipo de Mensaje (1 byte) | Longitud (2 bytes) | ID de Transacción
(1 byte) | IEs...
```

Cada IE tiene este formato:

```
Tipo de IE (1 byte) | Longitud de IE (2 bytes) | Valor de IE
(variable)
```

El campo `Longitud` cuenta el mensaje completo, incluido el encabezado de 4 bytes.

Tipos de Mensajes WLCP

Código	Mensaje	Dirección	Propósito
`0x01`	Solicitud de Conexión PDN	UE a TWAG	Solicitar una nueva conexión PDN
`0x02`	Aceptar Conexión PDN	TWAG a UE	Conexión creada, con IP / DNS / QoS
`0x03`	Rechazar Conexión PDN	TWAG a UE	Conexión rechazada, con una causa
`0x04`	Solicitud de Desconexión PDN	UE a TWAG	Liberar una conexión PDN
`0x05`	Aceptar Desconexión PDN	TWAG a UE	Conexión liberada
`0x06`	Notificación de Datos Descendentes	TWAG a UE	Notificar al UE sobre datos pendientes descendentes

Elementos de Información WLCP

Código	IE	Notas
`0x01`	APN	Cadena de Nombre de Punto de Acceso
`0x02`	Tipo de PDN	`1` IPv4, `2` IPv6, `3` IPv4v6
`0x03`	PCO	Opciones de Configuración de Protocolo
`0x04`	Dirección PDN	Dirección IP asignada al UE (IPv4, IPv6 o IPv4v6)
`0x05`	QoS de Portadora	Valores de QCI y MBR / GBR
`0x06`	TFT	Plantilla de Flujo de Tráfico
`0x07`	Causa	Valor de causa del resultado
`0x08`	ID de Portadora	Identidad de Portadora EPS (EBI)
`0x09`	Dirección DNS	Una o más direcciones de servidor DNS

Valores de Causa WLCP

Código	Causa	Significado
`0x10`	request_accepted	Solicitud aceptada
`0x20`	apn_not_allowed	APN no permitido para este UE
`0x21`	no_resources	No hay recursos disponibles
`0x22`	unknown_apn	APN no conocido
`0x23`	user_auth_failed	Fallo en la autenticación del usuario
`0x24`	network_failure	Fallo de red (por ejemplo, fallo de S2a)
`0x25`	pdn_type_not_supported	Tipo de PDN solicitado no soportado
`0x30`	request_rejected	Solicitud rechazada

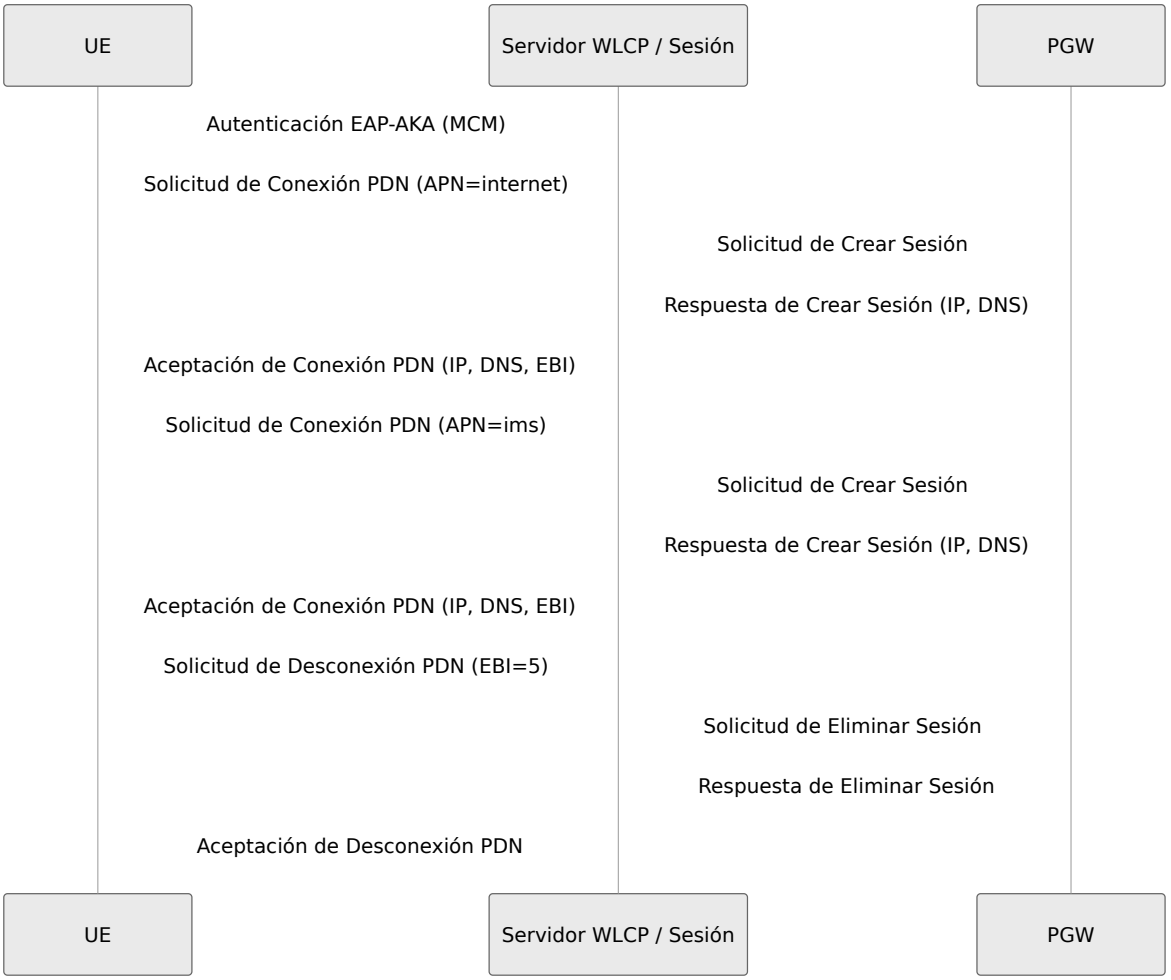
Configuración de Conexión PDN

Cuando llega una Solicitud de Conexión PDN, el controlador de sesión WLCP realiza estos pasos:

1. Lee el APN, tipo de PDN y PCO del mensaje. Si el APN está ausente, utiliza `internet`. Si el tipo de PDN está ausente, utiliza IPv4.
2. Verifica que el APN esté permitido. Consulte `allowed_apns` en Configuración.
3. Si el UE no tiene un IMSI conocido, rechaza la solicitud con la causa `user_auth_failed`.
4. Crea una sesión S2a GTP-C hacia el PGW para ese APN, con tipo RAT `3` (WLAN) y la bandera de Indicación de Transferencia clara.
5. En caso de éxito, registra el túnel GTP-U y almacena la conexión PDN, basada en el EBI. Luego envía una Aceptación de Conexión PDN con la

dirección IP del UE y, cuando esté presente, los servidores DNS, QoS de portadora y EBI.

6. En caso de fallo de S2a, envía un Rechazo de Conexión PDN con la causa `network_failure`.



Desconexión de Conexión PDN

Cuando llega una Solicitud de Desconexión PDN, el controlador encuentra la conexión PDN por EBI. Si el EBI es desconocido, responde con un Rechazo de Conexión PDN y causa `request_rejected`. Si el EBI es conocido, elimina la sesión S2a GTP-C, desregistra el túnel GTP-U, elimina la conexión y responde con una Aceptación de Desconexión PDN. Cuando el UE no tiene conexiones PDN restantes, el controlador de sesión WLCP se detiene.

Si el PGW elimina una portadora, el controlador elimina la conexión PDN correspondiente y libera el túnel GTP-U local. No envía una Eliminar Sesión en este caso, porque el PGW ya liberó la portadora.

Derivación de Clave WLCP

La clave WLCP `K_WLCP` protege la señalización WLCP. Se deriva de la Clave de Sesión Maestra (MSK) que produce la autenticación EAP. La derivación sigue la sección 6.2 y el Anexo A de 3GPP TS 33.402.

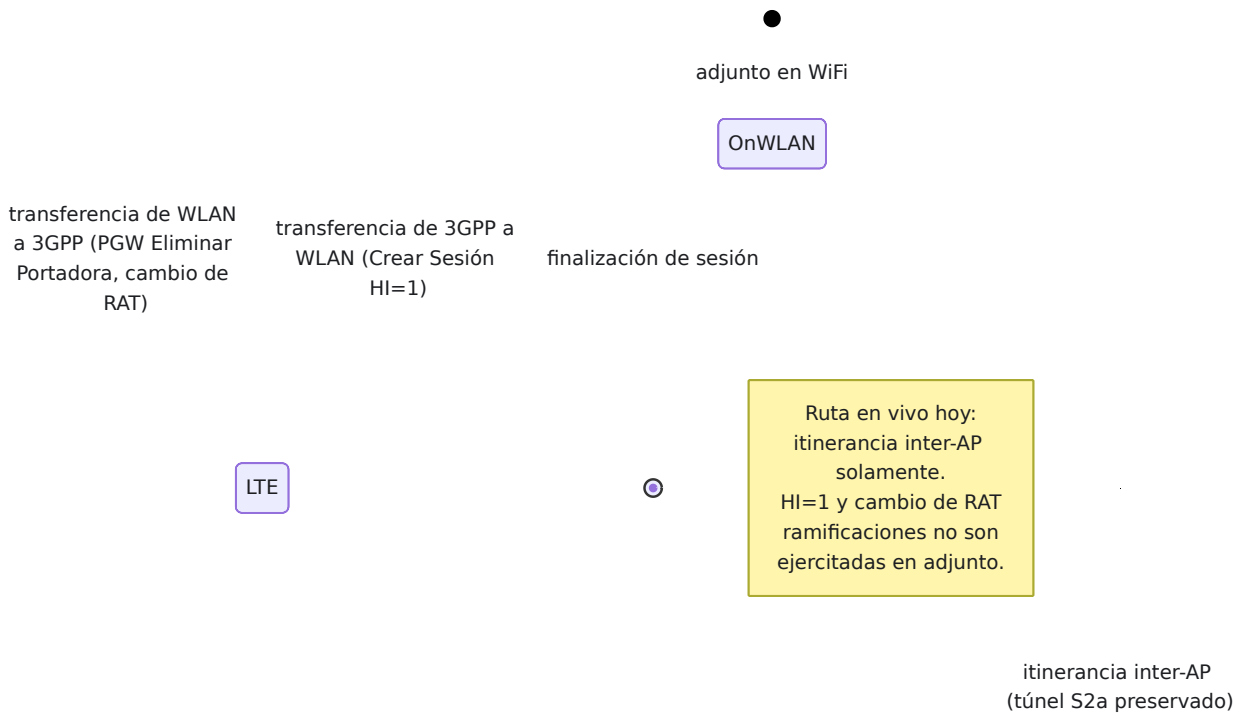
```
K_WLCP = HMAC-SHA-256(MSK, FC | P0 | L0)

FC = 0x21
P0 = dirección IP del TWAG (4 bytes para IPv4, 16 bytes para IPv6)
L0 = longitud de P0 (2 bytes, big-endian)
```

La salida es de 32 bytes (256 bits). El TWAG vincula la clave a su propia dirección IP a través del parámetro `P0`. Una variante simplificada deriva la clave de la MSK y una etiqueta fija cuando la IP del TWAG aún no se conoce.

> **Estado: MCM y WLCP son parciales y no están integrados.** El códec WLCP, el servidor UDP WLCP, el controlador de sesión por UE y la derivación de clave existen y están probados unitariamente. No son iniciados por la aplicación y no son impulsados por el flujo de autenticación. No se han implementado lo siguiente:
>
> - Protección DTLS para WLCP. Solo se maneja UDP simple.
> - Gestión de portadoras dedicadas dentro de MCM. El códec lleva QoS y TFT IEs, pero el controlador no actúa sobre la creación de portadoras dedicadas iniciada por el PGW.
> - Transferencia dentro de MCM.
> - El mensaje de Notificación de Datos Descendentes está definido en el códec pero no es generado por el controlador de sesión.
>
> Trate MCM como una vista previa. No lo habilite para UEs en producción.

Transferencia y Movilidad

El TWAG modela tres escenarios de movilidad según TS 23.402. Solo la movilidad inter-AP está activa en la ruta de sesión en vivo hoy.



Transferencia de 3GPP a WLAN

Cuando un UE se mueve de LTE a WiFi, el TWAG debe mantener la sesión IP existente. Para hacer esto, el TWAG envía una Solicitud de Crear Sesión con la bandera de Indicación de Transferencia (HI) establecida. El PGW luego preserva la dirección IP existente y el contexto de portadora, en lugar de asignar una nueva dirección. Consulte S2a GTP-C para el detalle de Crear Sesión y el IE de Indicación.

La Solicitud de Crear Sesión S2a admite la bandera de Indicación de Transferencia. El controlador de transferencia puede enrutar un evento de transferencia a la UE con la bandera de transferencia establecida.

> **Estado: la ruta de sesión principal no establece la bandera de transferencia.** El establecimiento automático de sesión siempre envía Crear Sesión con la bandera de Indicación de Transferencia clara. Como resultado, la preservación de IP en la transferencia de 3GPP a WLAN no es ejercitada por la ruta de adjunto normal en esta versión.

Transferencia de WLAN a 3GPP

Cuando un UE se mueve de WiFi a LTE, el PGW envía una Solicitud de Eliminar Portadora. El valor de causa indica un cambio de RAT. El TWAG trata estos valores de causa como una transferencia de cambio de RAT:

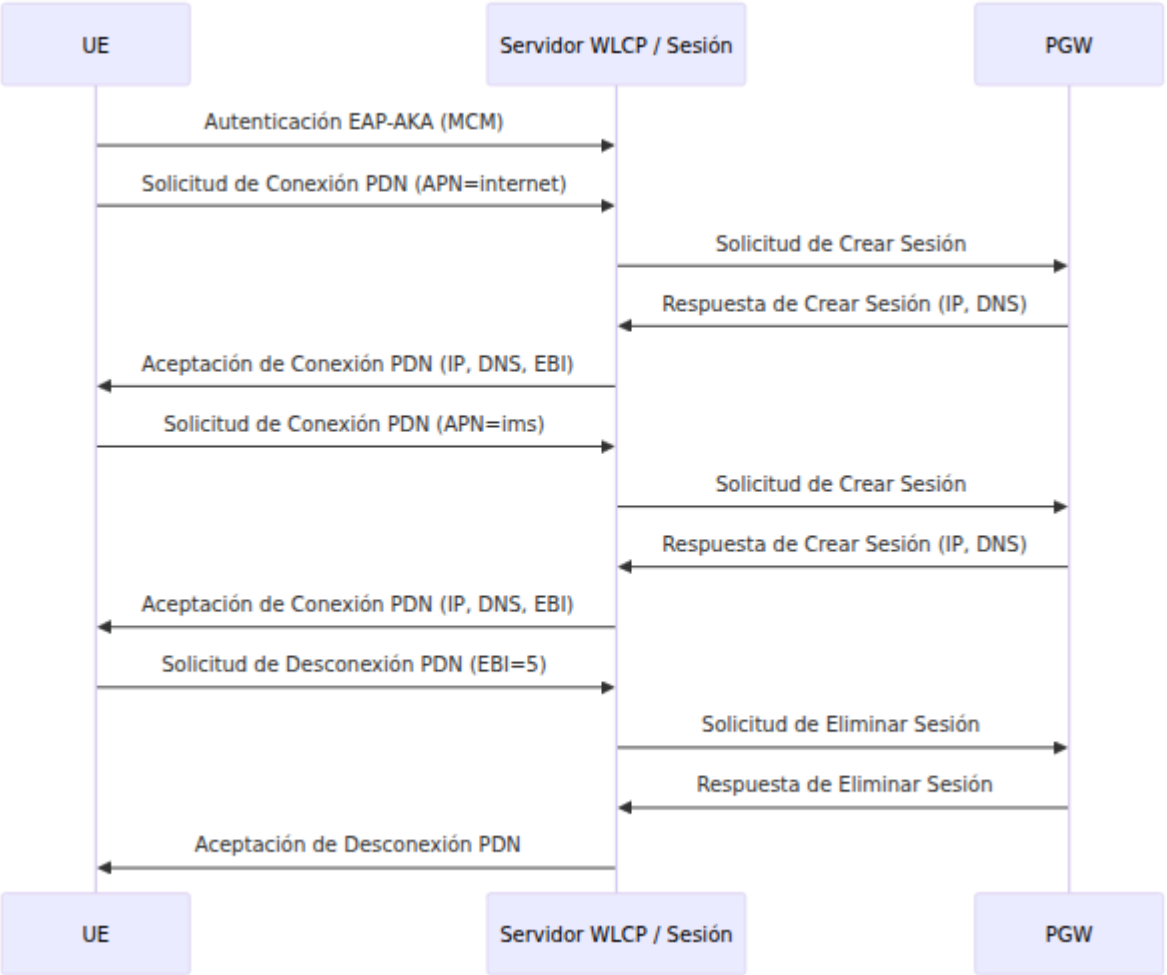
Código de causa	Nombre
`4`	RAT cambiado (3GPP a no 3GPP)
`5`	desactivación de ISR
`10`	Acceso cambiado de no 3GPP a 3GPP

En un Eliminar Portadora de cambio de RAT, el TWAG limpia la sesión de WiFi localmente. No envía una Eliminar Sesión GTP-C, porque el PGW ya liberó la portadora. El UE mantiene su dirección IP en el lado LTE.

> **Estado: el controlador de eliminar-portadora no ramifica en causa hoy.** El proceso del cliente UE maneja un Eliminar Portadora del PGW ejecutando limpieza local y luego deteniéndose, para cualquier causa. La clasificación de cambio de RAT existe como un ayudante, pero el cliente no usa la causa para elegir un camino de finalización diferente.

Movilidad Inter-AP

Cuando un UE itera entre APs de WiFi dentro del mismo TWAN, el túnel S2a GTP permanece activo. El UE mantiene la misma identidad, pero llega con una dirección NAS-IP diferente y un Called-Station-Id. El TWAG actualiza los datos de seguimiento de AP en el proceso del cliente UE y no finaliza la sesión. Esto mantiene la dirección IP del UE estable a través del cambio de AP.



La movilidad inter-AP es el único camino de movilidad que el código de sesión en vivo ejecuta. Actualiza los campos del proceso del cliente UE para NAS-IP y Called-Station-Id, y registra el nuevo AP en el estado de seguimiento de AP. Consulte Contabilidad por AP para los datos de seguimiento de AP.

Configuración

Modos de Conexión

Establezca los modos que el TWAG anuncia con la clave `connection_modes`. El valor es una lista de átomos de modo. El predeterminado es `[:tscm]`.

```
config :omnitwag,
  # Modos de conexión que este TWAG anuncia al servidor AAA.
  connection_modes: [:tscm, :scm]
```

Parámetro	Tipo	Requerido	Predeterminado	Descripción
`connection_modes`	Lista	No	`[:tscm]`	Modos que el TWAG anuncia. Use `:tscm`, `:scm` y `:mcm`. El orden no importa. Los códigos anunciados se ordenan antes de su uso.

> **Nota.** El conjunto anunciado es leído por el módulo de modo de conexión y el cliente STa envía el modo preferido en el DER STa. La ruta de sesión RADIUS en vivo aún no actúa sobre el modo negociado. Consulte Estado de Implementación. Mantenga el predeterminado `[:tscm]` para producción hasta que SCM y MCM estén integrados.

Servidor WLCP (MCM)

Los ajustes del servidor WLCP se encuentran bajo la clave `:wlcp`. El servidor WLCP solo es relevante para MCM.

```
config :omnitwag,
  wlcp: %{
    enabled: true,
    port: 2156,
    allowed_apns: :all
  }
```

Parámetro	Tipo	Requerido	Predeterminado	Descripci
`enabled`	Booleano	No	`false`	Habilitar el servidor WLC Deje `false` a menos que e probando MC
`port`	Entero	No	`2156`	Puerto UDP p WLCP. `2156` puerto están WLCP.
`allowed_apns`	Átomo o Lista	No	`:all`	Lista de pern de APN para solicitudes d PDN WLCP. U `:all` para permitir todo APN, o una li de cadenas c APN, por ejer `["internet" "ims"]`. Una solicitud para APN no en la se rechaza c causa `apn_not_all`

> **Nota.** El servidor WLCP no es iniciado por el supervisor de la aplicación en esta versión. La clave `enabled` está definida para uso futuro.

Configuración Relacionada

La ruta de configuración de PDN MCM reutiliza los ajustes de GTP para sus sesiones S2a. Consulte el resumen de configuración de Arquitectura para el

bloque `gtp`, que incluye `pgw_ip`, `local_ip`, `mcc` y `mnc`.

Observabilidad

El servidor WLCP y los controladores de sesión emiten eventos de telemetría. Estos eventos alimentan la tubería de métricas. Consulte Métricas para el catálogo de métricas.

Evento de telemetría	Emitido cuando
`[:wlcp, :packet, :received]`	Llega un paquete WLCP. Lleva `bytes`.
`[:wlcp, :packet, :malformed]`	Un paquete no se puede decodificar.
`[:wlcp, :packet, :unregistered]`	Llega un paquete de una dirección de origen no mapeada.
`[:wlcp, :pdn, :request]`	Se recibe una Solicitud de Conexión PDN.
`[:wlcp, :pdn, :accept]`	Se establece una conexión PDN.
`[:wlcp, :pdn, :reject]`	Se rechaza una conexión PDN. Lleva la causa.
`[:wlcp, :pdn, :disconnect_request]`	Se recibe una Solicitud de Desconexión PDN.
`[:wlcp, :pdn, :disconnect_accept]`	Se libera una conexión PDN.
`[:twag, :handover, :in]`	Se enruta un evento de transferencia de 3GPP a WLAN. Lleva `source`.
`[:twag, :handover, :out]`	Se maneja un cambio de RAT de WLAN a 3GPP. Lleva `cause`.
`[:twag, :handover, :ap_change]`	Se maneja un evento de movilidad inter-AP.

El servidor WLCP y los controladores de sesión también registran a nivel de información. Registran cada solicitud de conexión PDN, aceptación, rechazo y desconexión, con la identidad del UE, APN y EBI. El controlador de transferencia registra cada transferencia de entrada, cambio de RAT y cambio de AP, con la identidad del UE.

Estado de Implementación

Esta tabla resume lo que se ejecuta en la ruta de sesión en vivo, lo que existe pero no está integrado y lo que no está implementado. Úselo para planificar en torno al comportamiento real.

Característica	Estado
TSCM (predeterminado, PDN auto-creada, IP a través de DHCP)	En Vivo
Movilidad inter-AP (túnel S2a preservado)	En Vivo
S2a Crear Sesión con bandera de Indicación de Transferencia	Soportado en el codificador S2a, pero la ruta de adjunto siempre la limpia
Lógica de negociación de modo de conexión y retroceso	Presente como un módulo, probado unitariamente
STa DER anuncia modos admitidos	Conectado (STa DER lleva el Modo de Conexión TWAN preferido, más SSID / BSSID)
STa DEA lee el modo seleccionado	Conectado (StaClient analiza el modo negociado, predetermina a TSCM)
Modo negociado aplicado a la sesión en vivo	No conectado (`store_connection_mode` no tiene llamador en la ruta de adjunto)
SCM como una ruta de ejecución distinta (IP EAP, selección de APN)	No integrado (servido como TSCM)
Códec WLCP (todos los tipos de mensajes e IEs)	Presente, probado unitariamente
Servidor UDP WLCP y controlador de sesión por UE	Presente, no iniciado por la aplicación
Derivación de clave WLCP (`K_WLCP`, FC=0x21)	Presente, probado unitariamente

Característica	Estado
MCM múltiples conexiones PDN concurrentes	Presente en el controlador, no integrado de extremo a extremo
Protección DTLS para WLCP	No implementado (solo UDP simple)
Gestión de portadoras dedicadas en MCM	No implementado
Generación de Notificación de Datos Descendentes WLCP	No implementado (solo códec)
Transferencia en MCM	No implementado
Preservación de IP de 3GPP a WLAN en la ruta de adjunto	No ejercitada (adjunto limpia la bandera HI)
Finalización de WLAN a 3GPP por causa de cambio de RAT	No ramificado (la limpieza de eliminar-portadora es independiente de la causa)

Gestión de Direcciones IP (DHCP / DHCPv6 / IPv6 RA / PCO)

Descripción general

OmniTWAG le da a cada UE una dirección IP y los parámetros de red que la acompañan. Este subsistema cubre DHCPv4, DHCPv6, Anuncio de Router IPv6 (SLAAC) y la entrega de direcciones DNS y P-CSCF. El subsistema no inventa una dirección al azar. Entrega una dirección que otra parte del sistema ya eligió. La fuente depende del modo por AP.

- En **modo tunelado (enrutado a casa)**, el PGW asigna la dirección a través de S2a GTP, y OmniTWAG entrega esa misma dirección al UE. Este diseño sigue 3GPP TS 23.402 Sección 16.2.
- En **modo NSWO (salida local)** no hay PGW. OmniTWAG asigna la dirección por sí mismo desde un **grupo de IP** que el operador define. Ver Grupos de IP.

El modo, la fuente de la dirección y el grupo provienen de las credenciales del AP que coinciden con la fuente RADIUS. Ver Credenciales RADIUS para cómo un AP se mapea a un `breakout_type`, un `address_source` y un `ip_pool_id`.

Para los modos de operación y la ruta de datos GRE, ver la referencia de Plano de Usuario. Para el plano de control GTP-C que transporta el PAA y PCO, ver S2a GTP-C. Para la máquina de estados por UE que impulsa la asignación, ver Sesiones.

Tabla de Contenidos

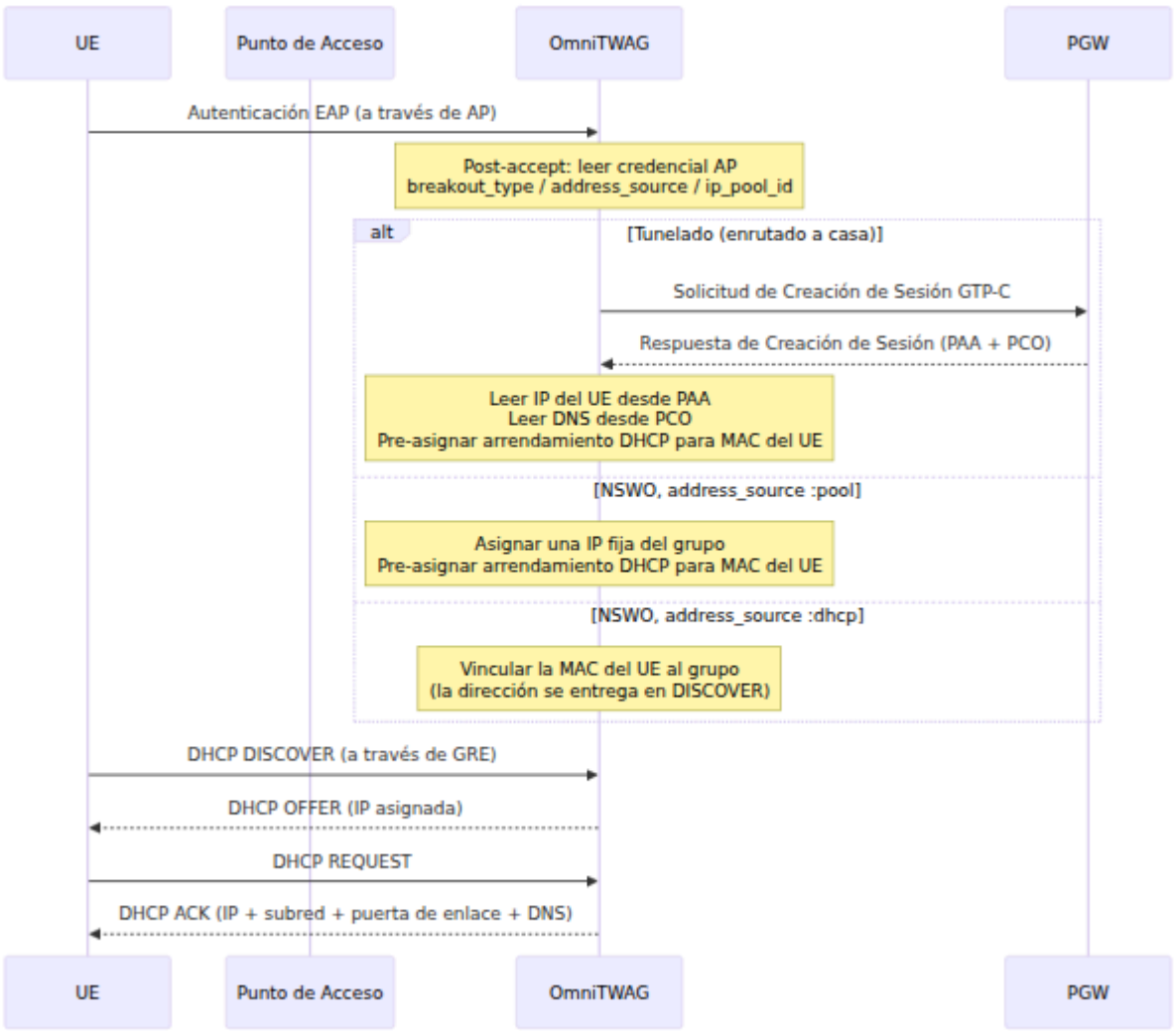
- Descripción general: quién asigna la dirección
- Fuentes de dirección
- Sesiones tuneladas: dirección asignada por PGW

- Sesiones NSWO: grupo y dhcp
- DHCPv4
- DHCPv6
- Anuncio de Router IPv6 (SLAAC)
- Entrega de DNS y PCO
- Grupos de direcciones y agotamiento
- Configuración
- Observabilidad
- Solución de problemas

Descripción general: quién asigna la dirección

OmniTWAG ejecuta un servidor DHCPv4 ligero, un servidor DHCPv6 sin estado y con estado, y un servidor de Anuncio de Router IPv6. Estos servidores no escanean un rango en busca de una dirección libre en el caso tunelado. Entregan la dirección que el PGW ya asignó. En modo NSWO, el mismo servidor DHCPv4 sirve una dirección que OmniTWAG tomó de un grupo de IP local.

El flujo de direcciones comienza cuando el UE se autentica. Después del Access-Accept, el proceso de sesión por UE ejecuta un paso de establecimiento posterior a la aceptación. Ese paso lee las credenciales del AP para decidir el modo y luego asigna la dirección por el camino correcto.



Fuentes de dirección

La credencial del AP establece la fuente de la dirección a través de dos campos:

- `breakout_type`: `:tunneled`, `:nswo` o `:local`.
- `address_source`: `:pool` o `:dhcp`. Esto se aplica solo cuando `breakout_type` es `:nswo`.

Modo	Credencial	Quién elige la dirección del UE	Cómo llega al UE
Tunelado (enrutado a casa)	`breakout_type::tunneled`	El PGW, en la Respuesta de Creación de Sesión GTP-C PAA.	Arrendamiento DHCPv4 pre-asignado, DHCPv6 o RA IPv6.
NSWO, fijo	`breakout_type::nswo`, `address_source::pool`	OmniTWAG, del grupo `ip_pool_id`, en el establecimiento.	Arrendamiento DHCPv4 pre-asignado.
NSWO, entrega	`breakout_type::nswo`, `address_source::dhcp`	OmniTWAG, del grupo `ip_pool_id`, en el DHCP DISCOVER.	Entrega de DHCPv4 del grupo en DISCOVER.
AP-local	`breakout_type::local`	La red local detrás del AP.	OmniTWAG no asigna ninguna dirección.

Para los campos de credencial y cómo una fuente RADIUS se mapea a una credencial, ver Credenciales RADIUS. Para el almacenamiento del grupo, ver Grupos de IP.

Sesiones tuneladas: dirección asignada por PGW

Para una sesión tunelada, OmniTWAG establece una sesión S2a con el PGW con una Solicitud de Creación de Sesión GTP-C. El PGW responde con una Respuesta de Creación de Sesión. La respuesta lleva dos cosas que este subsistema utiliza:

- La **Asignación de Dirección de PDN (PAA)**. Esta es la dirección IP del UE.
- Las **Opciones de Configuración de Protocolo (PCO)**. Esto lleva servidores DNS, direcciones P-CSCF y el MTU del enlace.

OmniTWAG lee el PAA y el PCO de la respuesta. OmniTWAG luego pre-asigna un arrendamiento DHCP para la dirección MAC del UE. Cuando el UE envía una solicitud DHCP, OmniTWAG responde con esta dirección pre-asignada. El UE nunca recibe una dirección que el PGW no asignó. Ver S2a GTP-C para el intercambio de Creación de Sesión.

El tipo de PAA establece qué dirección entrega OmniTWAG.

Tipo de PAA	Dirección entregada al UE
`ipv4`	La dirección IPv4 del PAA.
`ipv6`	La dirección IPv6 del PAA.
`ipv4v6`	La dirección IPv4 del PAA.

Si la Respuesta de Creación de Sesión no tiene PAA, OmniTWAG no pre-asigna un arrendamiento. En ese caso, el UE no recibe ninguna dirección.

Sesiones NSWO: grupo y dhcp

Una sesión NSWO no tiene PGW, por lo que OmniTWAG posee el espacio de direcciones. La dirección proviene del grupo de IP nombrado por la credencial `ip_pool_id`. Un grupo contiene un CIDR IPv4, una puerta de enlace, una lista de servidores DNS y un tiempo de arrendamiento. Ver Grupos de IP.

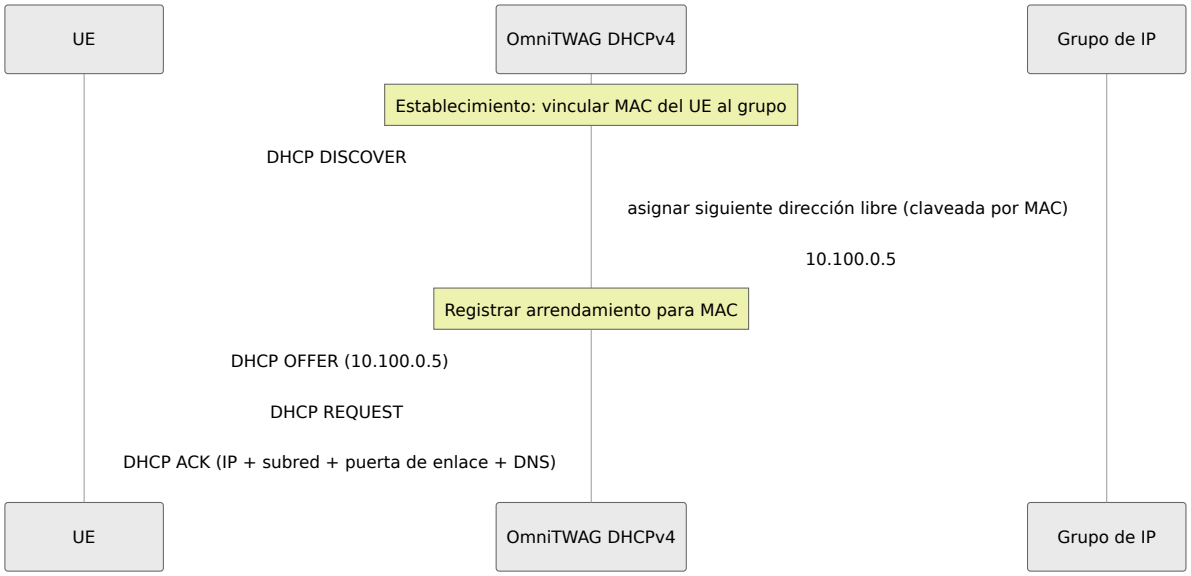
El campo `address_source` de la credencial selecciona cómo la dirección del grupo llega al UE.

`address_source: :pool` (fijo)

En el establecimiento posterior a la aceptación, OmniTWAG asigna una dirección del grupo, claveada por la identidad del suscriptor. La asignación es idempotente, por lo que la misma identidad mantiene la misma dirección a través de re-establecimientos. OmniTWAG pre-asigna un arrendamiento DHCPv4 para la MAC del UE con esa dirección, la puerta de enlace del grupo, los servidores DNS del grupo y una máscara de subred derivada del CIDR del grupo. También registra el UE en la ruta de datos GRE en este punto. Cuando el UE envía DISCOVER, el servidor ofrece la dirección pre-asignada.

`address_source: :dhcp` (entrega en DISCOVER)

En el establecimiento, OmniTWAG aún no asigna una dirección. **Vincula** la MAC del UE al grupo. En el primer DHCP DISCOVER de esa MAC, el servidor DHCPv4 asigna la siguiente dirección libre del rango del grupo, registra un arrendamiento y lo ofrece. Este es el nuevo camino de entrega dinámica. La dirección se elige en el momento de DISCOVER, no en el establecimiento.



En la finalización de la sesión, OmniTWAG libera la asignación del grupo y, para `:dhcp`, elimina la vinculación del grupo y el arrendamiento dinámico. Si el grupo está ausente o agotado, o el AP no tiene `ip_pool_id`, OmniTWAG no asigna ninguna dirección y el UE se auto-asigna.

DHCPv4

El servidor DHCPv4 escucha en el puerto UDP `67`. Almacena cada arrendamiento en una tabla ETS en memoria, claveada por la dirección MAC del UE. Un arrendamiento llega a la tabla por uno de dos caminos.

- **Pre-asignado.** El proceso de sesión inserta un arrendamiento cuando el PGW asigna una dirección (tunelado) o cuando OmniTWAG asigna una dirección fija del grupo (NSWO `address_source: :pool`).
- **Entregado en DISCOVER.** Para NSWO `address_source: :dhcp`, el proceso de sesión en su lugar vincula la MAC del UE a un grupo. El servidor mantiene una vinculación por MAC en memoria. En el primer DISCOVER de una MAC vinculada sin arrendamiento, el servidor asigna la siguiente dirección libre del grupo, registra el arrendamiento y lo ofrece.

Ciclo de vida del arrendamiento

Para un arrendamiento pre-asignado, el servidor sirve la dirección que ya existe en la tabla para la MAC solicitante. Para una MAC vinculada, el servidor crea el arrendamiento en DISCOVER al entregar desde el grupo.

Parse error on line 2: ...ar (PGW PAA o grupo :pool) [*] --> V ----------^ Expecting 'SPACE', 'NL', 'HIDE_EMPTY', 'scale', 'COMPOSIT_STATE', 'STRUCT_STOP', 'STATE_DESCR', 'ID', 'FORK', 'JOIN', 'CHOICE', 'CONCURRENT', 'note', 'acc_title', 'acc_descr', 'acc_descr_multiline_value', 'CLICK', 'classDef', 'style', 'class', 'direction_tb', 'direction_bt', 'direction_rl', 'direction_lr', 'EDGE_STATE', got 'DESCR'

Intente de nuevo

OmniTWAG maneja estos tipos de mensajes DHCP.

Mensaje	Código de tipo	Comportamiento de OmniTWAG
DISCOVER	1	Si existe un arrendamiento para la MAC, OmniTWAG envía una OFERTA. Si no existe arrendamiento pero la MAC está vinculada a un grupo, OmniTWAG entrega la siguiente dirección libre del grupo, registra un arrendamiento y envía una OFERTA. Si no existe arrendamiento ni vinculación, OmniTWAG registra una advertencia y no envía nada.
REQUEST	3	Si existe un arrendamiento y la IP solicitada (opción 50) coincide con el arrendamiento, o la solicitud no tiene IP solicitada, OmniTWAG envía un ACK. Si la IP solicitada no coincide, OmniTWAG envía un NAK. Si no existe arrendamiento, OmniTWAG envía un NAK.
RELEASE	7	OmniTWAG marca el arrendamiento como `liberado`. OmniTWAG mantiene el registro del arrendamiento. La conexión PDN permanece activa, según TS 23.402 §16.1.5.2 NOTA 2.
DECLINE	4	OmniTWAG registra una advertencia. OmniTWAG mantiene el registro del arrendamiento.
INFORM	8	OmniTWAG envía un ACK solo con configuración. El ACK lleva la puerta de enlace, la máscara de subred y DNS, pero no dirección (`yiaddr` es `0.0.0.0`).

Para INFORM, OmniTWAG utiliza la `default_gateway` y `default_subnet_mask` de la configuración. OmniTWAG utiliza `8.8.8.8` como el servidor DNS en la respuesta INFORM.

Contenidos del arrendamiento

Cada arrendamiento contiene la IP del UE, la máscara de subred, la puerta de enlace, los servidores DNS y el tiempo de arrendamiento. Para una sesión tunelada, el proceso de sesión establece la IP del UE y los servidores DNS de la respuesta del PGW, y la máscara de subred, la puerta de enlace y el tiempo de arrendamiento de la configuración DHCP. Para una sesión NSWO, estos valores provienen del grupo de IP: la IP del rango del grupo, la máscara de subred del prefijo CIDR del grupo, y la puerta de enlace, los servidores DNS y el tiempo de arrendamiento del registro del grupo. Ver Grupos de IP y Configuración.

Un registro de arrendamiento también tiene un tiempo `allocated_at` y un tiempo `expires_at`. El tiempo `expires_at` es `allocated_at` más el tiempo de arrendamiento. OmniTWAG utiliza estos valores para informar el estado del arrendamiento. Un recolector en segundo plano se ejecuta cada 60 segundos y elimina cualquier arrendamiento cuyo `expires_at` haya pasado. Un arrendamiento con un tiempo de arrendamiento de cero no tiene `expires_at` y nunca expira. Un arrendamiento también se elimina cuando se finaliza la sesión.

Opciones DHCP en una respuesta

OmniTWAG construye cada OFERTA y ACK con estas opciones DHCP.

Opción	Código	Contenido
Tipo de Mensaje	53	OFERTA o ACK.
Identificador del Servidor	54	La `server_ip` DHCP.
Máscara de Subred	1	La máscara de subred del arrendamiento.
Enrutador	3	La puerta de enlace del arrendamiento.
Servidor DNS de Nombre Recursivo	6	Los servidores DNS del arrendamiento. Omitido si la lista está vacía.
Tiempo de Arrendamiento de Dirección IP	51	El tiempo de arrendamiento en segundos. Omitido si el tiempo de arrendamiento es cero.
Dirección de Difusión	28	Calculada a partir de la IP y la máscara de subred. Enviada solo si el UE lista esta opción en su Lista de Solicitud de Parámetros (opción 55).

Transporte DHCP sobre GRE

Un punto de acceso que utiliza un túnel GRE reenvía la solicitud DHCP del UE dentro de un marco Ethernet encapsulado en GRE. El servidor GRE extrae la carga útil DHCP y la pasa al servidor DHCPv4. OmniTWAG etiqueta el paquete con la dirección del AP de origen. Cuando OmniTWAG construye la respuesta, envuelve la carga útil DHCP en un marco UDP/IP/Ethernet y envía el marco de vuelta a través del túnel GRE al mismo AP. La fuente es `10.0.0.1:67` y el destino es la dirección de difusión en el puerto del cliente `68`.

Una solicitud DHCP que llega directamente al socket UDP (no a través de GRE) se responde en el socket UDP. Si el puerto de origen no es el puerto estándar del cliente `68`, OmniTWAG responde a la dirección y puerto de origen. Este camino admite herramientas de prueba y relés. De lo contrario, OmniTWAG

sigue RFC 2131: utiliza la dirección de difusión si la bandera de difusión está establecida o `ciaddr` es cero, y utiliza `ciaddr` si esa dirección está establecida.

DHCPv6

El servidor DHCPv6 escucha en el puerto UDP `547` sobre IPv6. Soporta tanto un modo sin estado como un modo con estado.

- **Modo sin estado** responde a una Solicitud de Información (tipo 11) con una Respuesta de Información (tipo 12). La respuesta lleva los servidores DNS. Esto sigue RFC 3736 y RFC 3646. Un UE utiliza este modo cuando establece la bandera de Otra Configuración (O) desde su procesamiento de RA.
- **Modo con estado** responde a una Solicitud (tipo 1) con un Anuncio (tipo 2), y a una Solicitud (tipo 3) con una Respuesta (tipo 7). Este modo entrega una dirección IPv6 pre-asignada en una opción IA_NA. Esto sigue RFC 3315.

El servidor pre-asigna una dirección con estado por cliente DUID. El plano de control GTP inserta la asignación. El servidor mantiene estas asignaciones en una tabla en memoria, claveada por el DUID del cliente.

Manejo de mensajes DHCPv6

Mensaje	Tipo	Comportamiento de OmniTWAG
Solicitud de Información	11	OmniTWAG envía una Respuesta de Información con el DUID del Servidor, el ID del Cliente ecoado y los servidores DNS.
Solicitud	1	OmniTWAG envía un Anuncio. Si el DUID del cliente tiene una asignación, el Anuncio lleva la dirección IA_NA. Si no, el Anuncio lleva un estado `NoAddrsAvail` (código 2).
Solicitud	3	OmniTWAG envía una Respuesta. La Respuesta confirma la dirección IA_NA para el DUID del cliente.
Otros tipos	-	OmniTWAG ignora el mensaje.

Cada respuesta ecoa el ID de transacción del cliente, según RFC 3315 §15. Cada respuesta que incluye una dirección asignada establece los temporizadores DHCPv6 a partir del tiempo de arrendamiento. T1 es la mitad del tiempo de arrendamiento. T2 es cuatro quintos del tiempo de arrendamiento.

DUID del servidor

OmniTWAG construye un DUID del Servidor de tipo DUID-EN (Número de Empresa). Utiliza el número de empresa 3GPP `10415`. El identificador se deriva del nombre del nodo, por lo que es estable para un nodo dado.

Opciones DHCPv6 en una respuesta

Opción	Código	Contenido
Identificador del Cliente	1	El ID del Cliente ecoado desde la solicitud, si está presente.
Identificador del Servidor	2	El DUID del Servidor.
Asociación de Identidad para Direcciones No Temporales (IA_NA)	3	La dirección IPv6 asignada, o un estado `NoAddrsAvail`. Solo en modo con estado.
Servidor de Nombres DNS Recursivo	23	Los servidores DNS configurados. Omitido si la lista está vacía.
Lista de Búsqueda de Dominio	24	La lista de búsqueda configurada. Enviada solo si `dhcpv6_domain_list` está establecida.

Anuncio de Router IPv6 (SLAAC)

El servidor RA IPv6 entrega prefijos IPv6 asignados por PGW a los UEs con Autoconfiguración de Dirección Sin Estado (SLAAC). Implementa ICMPv6 Anuncio de Router según RFC 4861.

El plano de control GTP agrega un prefijo por interfaz. El servidor mantiene los prefijos en memoria, agrupados por nombre de interfaz. El servidor envía un Anuncio de Router en dos casos:

1. En un temporizador periódico. El intervalo predeterminado es de `30` segundos.
2. A demanda, cuando el plano de control solicita un RA en una interfaz.

El servidor también analiza mensajes de Solicitud de Router (ICMPv6 tipo 133).

Contenidos del Anuncio de Router

Cada Anuncio de Router lleva estos campos y opciones.

Campo u opción	Valor
Límite de Salto Actual	Desde `hop_limit`. Predeterminado `64`.
Tiempo de Vida del Enrutador	Desde `router_lifetime`. Predeterminado `1800` segundos.
Banderas M y O	Ambas `0`. El RA no señala configuración de dirección administrada ni otra configuración.
Dirección de Capa de Enlace de Origen (opción 1)	La MAC de origen, si es un valor de 6 bytes.
MTU (opción 5)	Incluido solo si se proporciona un MTU.
Información de Prefijo (opción 3)	Una por prefijo. Las banderas L (enlace) y A (autónoma) están ambas establecidas.
Servidor DNS Recursivo, RDNSS (opción 25)	Los servidores DNS de los prefijos, según RFC 8106. Omitido si no hay servidores DNS.

Cada opción de Información de Prefijo lleva el prefijo, la longitud del prefijo, el tiempo de vida válido (predeterminado `86400` segundos) y el tiempo de vida preferido (predeterminado `14400` segundos). Debido a que la bandera A está establecida, el UE forma su propia dirección IPv6 a partir del prefijo. El tiempo de vida de RDNSS es el mayor tiempo de vida válido entre los prefijos.

Entrega de DNS y PCO

OmniTWAG entrega información de DNS y P-CSCF de más de una manera. La fuente es el PCO del PGW donde sea posible, y la configuración de lo contrario.

PCO del PGW

El analizador de PCO lee las Opciones de Configuración de Protocolo de la Respuesta de Creación de Sesión, según 3GPP TS 24.008. El analizador devuelve estos campos.

Campo	ID de protocolo PCO	Significado
`dns_ipv4`	`0x000D`	Direcciones de servidores DNS IPv4.
`dns_ipv6`	`0x0003`	Direcciones de servidores DNS IPv6.
`pcscf_ipv4`	`0x000C`	Direcciones P-CSCF IPv4.
`pcscf_ipv6`	`0x0001`	Direcciones P-CSCF IPv6.
`mtu`	`0x0010`	MTU de enlace IPv4.
`ip_alloc_nas`	`0x000A`	Una bandera. Es `true` si el PGW señala la asignación de IP a través de señalización NAS.

El analizador también mantiene cada opción analizada en una lista `raw`. Si el binario de PCO está mal formado en algún momento, el analizador devuelve los campos que ha leído hasta ahora. No falla todo el análisis.

OmniTWAG también puede construir una solicitud de PCO para la Solicitud de Creación de Sesión. La solicitud enumera las opciones que OmniTWAG desea que el PGW devuelva, por ejemplo, los servidores DNS IPv4 y el MTU del enlace.

Entrega de DNS al UE

Los servidores DNS llegan al UE a través del transporte que utiliza el UE.

Transporte	Fuente DNS	Notas
DHCPv4 (opción 6)	Los servidores DNS IPv4 del PCO.	Si el PCO no tiene servidor DNS IPv4, OmniTWAG utiliza `8.8.8.8`.
DHCPv6 (opción 23)	La lista de servidores DNS DHCPv6.	Establecida desde el PCO del PGW o desde `dhcpv6_dns_servers`. Por defecto, utiliza Google Public DNS IPv6 si no está establecida.
RA IPv6 (RDNSS, opción 25)	Los servidores DNS IPv6 adjuntos al prefijo anunciado.	Entregados con el prefijo.

El arrendamiento DHCPv4 toma los servidores DNS IPv4 de la lista `dns_ipv4` del PCO. Si esa lista está vacía, OmniTWAG recurre a `8.8.8.8` para el arrendamiento.

> **Nota sobre P-CSCF.** El analizador de PCO lee las direcciones P-CSCF del PGW. Las rutas actuales de entrega de DHCP, DHCPv6 y RA no añaden una opción DHCP de P-CSCF o una opción RA. Los valores de P-CSCF se analizan y están disponibles, pero no se entregan al UE a través de este subsistema en el código auditado.

Grupos de direcciones y agotamiento

El comportamiento de agotamiento depende del modo.

- **Tunelado (enrutado a casa).** No hay grupo de direcciones local. El PGW posee el espacio de direcciones. OmniTWAG entrega exactamente la dirección que el PGW asigna para cada sesión. No hay rango local que

agotar. Si el PGW no devuelve un PAA, el UE no recibe ninguna dirección, y OmniTWAG registra la condición.

- **NSWO.** OmniTWAG asigna desde un grupo de IP local, por lo que un grupo puede agotarse. Un grupo es un CIDR IPv4 de como máximo un `/30`. El asignador omite las direcciones de red y de difusión y entrega el primer host libre. Cuando cada host está en uso, el asignador devuelve `pool_exhausted`. Para `address_source: :pool`, el UE se auto-asigna. Para `address_source: :dhcp`, el DISCOVER no recibe respuesta y OmniTWAG registra el agotamiento. Las asignaciones están en memoria y se reconstruyen a medida que las sesiones se re-establecen, igual que los arrendamientos DHCP. Ver Grupos de IP para la definición y capacidad del grupo.

El helper `Twag.Cidr` realiza los cálculos de subred para la plataforma. Analiza cadenas CIDR IPv4 e IPv6 y prueba si una dirección está dentro de una subred. El subsistema RADIUS utiliza este helper para la autorización de IP de origen. El almacenamiento del grupo de IP lo utiliza para derivar la máscara de subred del grupo, el rango de hosts utilizables y la puerta de enlace predeterminada.

Configuración

Los servidores de gestión de IP leen su configuración de la configuración de la aplicación `:omnitwag`. Los bloques principales son `dhcp` y `ipv6_ra`. DHCPv6 lee un pequeño conjunto de claves de nivel superior.

```
config :omnitwag,
  # Servidor DHCPv4
  dhcp: %{
    enabled: true,
    interface: "wlan0",
    server_ip: "10.0.0.1",
    default_lease_time: 3600,
    default_subnet_mask: "255.255.255.0",
    default_gateway: "10.0.0.1"
  },

  # Anuncio de Router IPv6
  ipv6_ra: %{
    enabled: true,
    interface: "wlan0",
    ra_interval: 30,
    hop_limit: 64,
    router_lifetime: 1800
  },

  # DHCPv6 (claves de nivel superior opcionales)
  dhcpv6_dns_servers: ["2001:4860:4860::8888",
"2001:4860:4860::8844"],
  dhcpv6_domain_list: ["example.com"]
```

Parámetros DHCPv4 (`dhcp`)

Parámetro	Tipo	Requerido	Predeterminado	Des
`enabled`	Booleano	No	`true`	Habi servi DHCI Estal `fals` desh por e un er prue
`interface`	Cadena	No	`wlan0`	Inter para DHCI para túne
`server_ip`	Cadena	No	`10.0.0.1`	El ide del s DHCI Omn envía en la DHCI como respu enca en G
`default_lease_time`	Entero	No	`3600`	Tiem arren en se Omn aplic

Parámetro	Tipo	Requerido	Predeterminado	Des
				valor arren pre-a
`default_subnet_mask`	Cadena	No	`255.255.255.0`	Másc subre arren y par cálcu difus 2131
`default_gateway`	Cadena	No	`10.0.0.1`	Puert enlac pred para arren (opci 3).

Parámetros RA IPv6 (`ipv6_ra`)

Parámetro	Tipo	Requerido	Predeterminado	Descrip
`enabled`	Booleano	No	`true`	Habilitar servidor R IPv6. Establece `false` pa deshabilit
`interface`	Cadena	No	`wlan0`	Interfaz d red para Anuncios Router.
`ra_interval`	Entero	No	`30`	Intervalo segundos entre Anuncios Router periódico
`hop_limit`	Entero	No	`64`	Campo de Límite de Salto Actu en el Anu de Router
`router_lifetime`	Entero	No	`1800`	Tiempo d Vida del Enrutado segundos el Anunci Router.

Parámetros DHCPv6 (nivel superior)

Parámetro	Tipo	Requerido	Predeterminado	D
`dhcpv6_dns_servers`	Lista de Cadena	No	DNS Público de Google IPv6 (`2001:4860:4860::8888`, `2001:4860:4860::8844`)	S D p r D P P t p e e t e
`dhcpv6_domain_list`	Lista de Cadena	No	(no establecido)	L b D r D (S e C o o

Observabilidad

Interfaz web

El panel de control tiene una página de **Grupo DHCP** en `/dhcp`. La página lista cada arrendamiento DHCPv4 con la dirección MAC del UE, la dirección IP, el tiempo de inicio del arrendamiento, el tiempo de expiración del arrendamiento y el estado. El estado es `Activo` o `Expirado`. La página muestra una cuenta regresiva en vivo para cada arrendamiento activo. Seleccione una fila de arrendamiento para ver los detalles completos del arrendamiento: máscara de subred, puerta de enlace, servidores DNS, duración del arrendamiento y marcas de tiempo. La página se actualiza cada segundo. Si el servidor DHCPv4 no está en funcionamiento, la página muestra `Servidor No Ejecutándose`.

API REST

La API REST sirve datos de arrendamiento DHCPv4 bajo la etiqueta `DHCP`.

Método	Ruta	Descripción
GET	`/dhcp` (índice)	Lista todos los arrendamientos DHCPv4 con un resumen de conteo.
GET	`/dhcp/{id}` (mostrar)	Obtener un arrendamiento por dirección MAC. El `id` es la MAC separada por dos puntos.

La respuesta del índice tiene una bandera `enabled`, la lista de `leases`, un `count` total, un conteo `active` y un conteo `expired`. Si el servidor DHCPv4 no está en funcionamiento, el índice devuelve `enabled: false` con una lista vacía. El endpoint de mostrar devuelve `404` si no hay arrendamiento que coincida con la MAC, y `503` si el servidor no está en funcionamiento.

Cada objeto de arrendamiento tiene estos campos.

Campo	Descripción
`mac`	Dirección MAC del UE, separada por dos puntos.
`ip`	Dirección IPv4 asignada.
`subnet_mask`	Máscara de subred del arrendamiento.
`gateway`	Puerta de enlace del arrendamiento.
`dns_servers`	Lista de servidores DNS.
`lease_time`	Duración del arrendamiento en segundos.
`allocated_at`	Tiempo de inicio del arrendamiento, en segundos desde la época Unix.
`expires_at`	Tiempo de expiración del arrendamiento, en segundos desde la época Unix.
`status`	`activo` o `expirado`.
`time_remaining`	Segundos hasta la expiración, o `0` si ha expirado.

Eventos de telemetría

Los servidores de gestión de IP emiten eventos de telemetría. Un operador puede adjuntar estos eventos a un exportador de métricas. Ver la Referencia de Métricas para el endpoint de Prometheus.

Evento	Fuente	Significado
`[:dhcp, :lease, :allocate]`	DHCPv4	Se pre-asignó un arrendamiento para una MAC.
`[:dhcp, :lease, :deallocate]`	DHCPv4	Se eliminó un arrendamiento en la finalización.
`[:dhcp, :lease, :reap]`	DHCPv4	Un arrendamiento expirado fue eliminado por el recolector en segundo plano.
`[:dhcp, :discover]`	DHCPv4	Un DISCOVER fue respondido con una OFERTA de un arrendamiento existente.
`[:dhcp, :discover, :pool_allocated]`	DHCPv4	Un DISCOVER de una MAC vinculada a un grupo fue respondido al entregar una nueva dirección del grupo (NSWO `:dhcp`).
`[:dhcp, :discover, :no_allocation]`	DHCPv4	Un DISCOVER llegó para una MAC sin arrendamiento y sin vinculación al grupo.
`[:dhcp, :ack]`	DHCPv4	Una REQUEST fue respondida con un ACK.
`[:dhcp, :nak]`	DHCPv4	Una REQUEST fue respondida con un NAK.
`[:dhcp, :release]`	DHCPv4	Un cliente liberó su arrendamiento.
`[:dhcp, :decline]`	DHCPv4	Un cliente rechazó su arrendamiento.
`[:dhcp, :inform]`	DHCPv4	Un INFORM fue respondido.

Evento	Fuente	Significado
`[:dhcp, :packet, :malformed]`	DHCPv4	Se recibió un paquete DHCP mal formado.
`[:dhcpv6, :info_request, :received]`	DHCPv6	Se recibió una Solicitud de Información.
`[:dhcpv6, :solicit, :received]`	DHCPv6	Se recibió una Solicitud.
`[:dhcpv6, :request, :received]`	DHCPv6	Se recibió una Solicitud.
`[:dhcpv6, :info_reply, :sent]`	DHCPv6	Se envió una Respuesta de Información.
`[:dhcpv6, :advertise, :sent]`	DHCPv6	Se envió un Anuncio.
`[:dhcpv6, :reply, :sent]`	DHCPv6	Se envió una Respuesta.
`[:ipv6_ra, :prefix, :add]`	RA IPv6	Se agregó un prefijo a una interfaz.
`[:ipv6_ra, :prefix, :remove]`	RA IPv6	Se eliminó un prefijo de una interfaz.
`[:ipv6_ra, :advertisement, :sent]`	RA IPv6	Se envió un Anuncio de Router.

Solución de problemas

El UE no recibe dirección IP

Síntomas: El UE se autentica pero no recibe una dirección IPv4. La página del Grupo DHCP muestra ningún arrendamiento para la MAC del UE. Se activa el evento `[:dhcp, :discover, :no_allocation]`.

Causas posibles (tunelado):

- La Respuesta de Creación de Sesión GTP no tenía PAA, por lo que OmniTWAG no pre-asignó un arrendamiento.
- La sesión GTP-C no se estableció, por lo que no se creó ningún arrendamiento.

Causas posibles (NSWO):

- La credencial del AP no tiene `ip_pool_id`, por lo que OmniTWAG no asigna ninguna dirección.
- El grupo está agotado, por lo que no se pudo entregar ninguna dirección libre.
- Para `address_source: :dhcp`, el DISCOVER llegó antes de que la MAC estuviera vinculada al grupo, o la vinculación no se ejecutó porque la sesión aún no tenía MAC.

Causa común:

- El servidor DHCPv4 no está en funcionamiento.

Resolución:

1. Verifique los registros de OmniTWAG para el camino de establecimiento (tunelado versus NSWO) y la IP del UE.
2. Para sesiones tuneladas, confirme que el PGW devuelve un PAA para el APN.
3. Para sesiones NSWO, confirme que la credencial del AP nombra un `ip_pool_id` válido y que el grupo tiene capacidad libre. Ver Grupos de IP y Credenciales RADIUS.

4. Confirme que la página del Grupo DHCP muestra el servidor DHCPv4 como en funcionamiento.

El UE recibe una dirección pero no DNS

Síntomas: El UE tiene una dirección IPv4 pero no puede resolver nombres.

Causas posibles:

- El PCO del PGW no devolvió ningún servidor DNS IPv4. En ese caso, OmniTWAG utiliza `8.8.8.8`.
- El UE no puede alcanzar el servidor DNS que el PGW proporcionó.

Resolución:

1. Lea los servidores DNS del arrendamiento en la página del Grupo DHCP o a través de la API REST.
2. Confirme que el PCO del PGW devuelve el servidor DNS esperado.
3. Confirme que el servidor DNS es alcanzable desde la ruta de datos del UE.

El UE recibe un NAK

Síntomas: El UE envía una REQUEST y recibe un NAK. Se activa el evento `[:dhcp, :nak]`.

Causas posibles:

- El UE solicitó una IP (opción 50) que no coincide con su arrendamiento pre-asignado.
- El arrendamiento fue eliminado, por ejemplo, después de una finalización de sesión.

Resolución:

1. Confirme que existe un arrendamiento para la MAC del UE.
2. Confirme que el UE solicita la misma IP que el PGW o el grupo asignó.
3. Si la sesión se finalizó, deje que el UE reinicie el intercambio DHCP después de que se re-autentique.

Páginas relacionadas

- Descripción general
- Grupos de IP
- Credenciales RADIUS
- Sesiones
- S2a GTP-C
- Plano de Usuario
- Carga en Línea (Gy)

Descripción general

Grupos de IP

Resumen

OmniTWAG asigna direcciones IPv4 a sesiones NSWO (Non-Seamless WLAN Offload) desde un **grupo de IP**. NSWO es un desvío local anclado en TWAG: OmniTWAG termina el tráfico del usuario y lo desvia localmente. Un grupo de IP define un bloque CIDR IPv4, una puerta de enlace, servidores DNS y un tiempo de arrendamiento. El operador añade, cambia y elimina grupos a demanda a través de la API REST o la interfaz web del panel de control.

Las definiciones de grupos son persistentes. OmniTWAG las almacena en un archivo DETS, por lo que los grupos sobreviven a un reinicio. Las asignaciones de direcciones son en memoria. OmniTWAG reconstruye las asignaciones a medida que las sesiones se restablecen después de un reinicio.

Tabla de Contenidos

- Conceptos
- Cómo una sesión obtiene una dirección
- Asignación de direcciones
- Modelo de grupo
- API REST
- Interfaz web
- Resolución de problemas

Conceptos

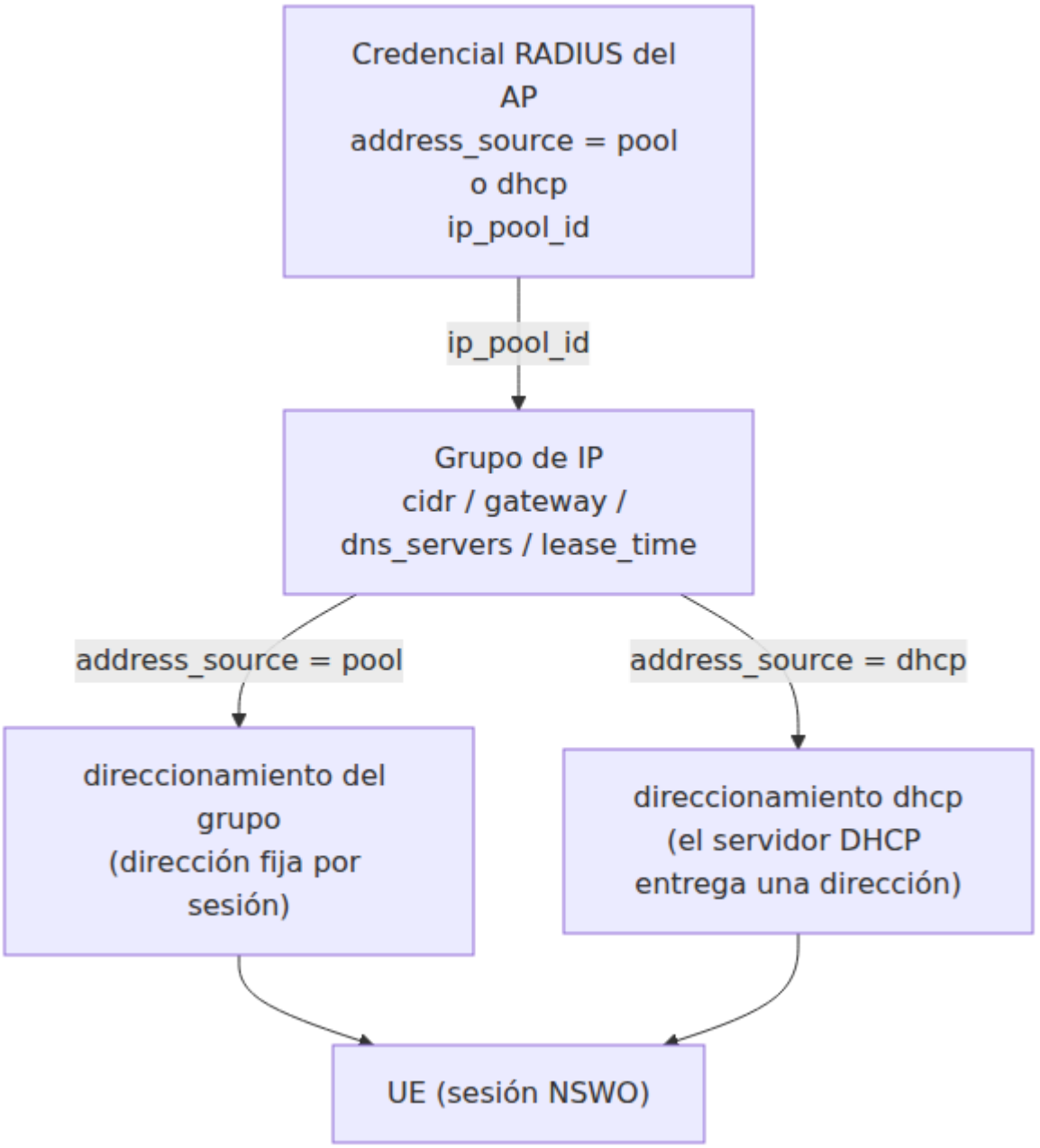
Un **grupo** es un conjunto de direcciones de un bloque CIDR IPv4. Cada grupo tiene una puerta de enlace, una lista de servidores DNS y un tiempo de arrendamiento.

Un grupo de IP está vinculado a un punto de acceso a través de las credenciales RADIUS del AP. La credencial tiene un campo `ip_pool_id`. Este

campo contiene el `id` del grupo. Consulta Secretos Compartidos de RADIUS para el modelo de credenciales. Las sesiones NSWO de ese punto de acceso se dirigen desde el grupo vinculado.

Cada credencial también tiene un campo `address_source`. Este campo selecciona cómo OmniTWAG proporciona la dirección al UE (equipo de usuario). Hay dos fuentes de dirección:

Fuente de dirección	Comportamiento
`pool`	OmniTWAG pre-asigna una dirección fija por sesión desde el grupo. La dirección se elige cuando se configura la sesión.
`dhcp`	El servidor DHCP de OmniTWAG entrega una dirección del rango del grupo en respuesta a un DHCP DISCOVER del UE.

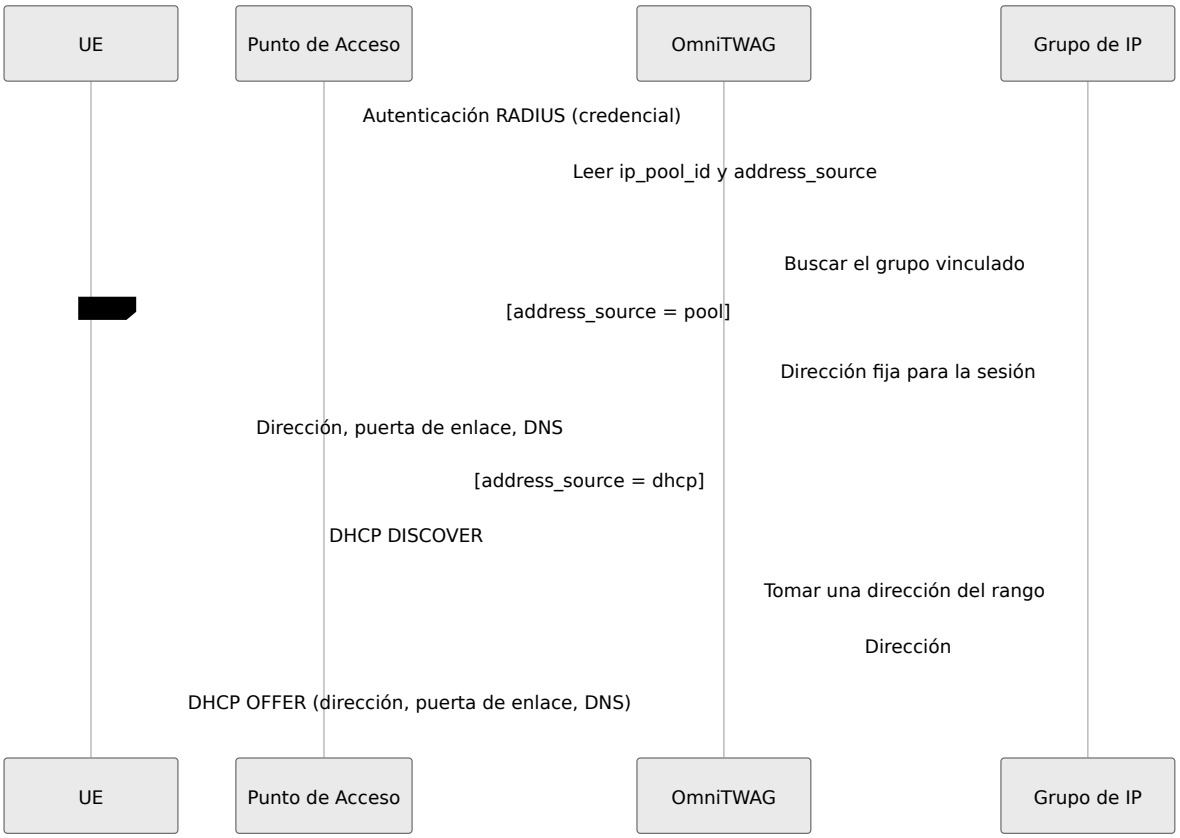


Cómo una sesión obtiene una dirección

OmniTWAG dirige cada sesión NSWO desde el grupo que está vinculado a la credencial del AP. Los pasos son:

1. El punto de acceso se autentica a través de su credencial RADIUS.
2. OmniTWAG lee el campo `ip_pool_id` en la credencial para encontrar el grupo.

3. OmniTWAG lee el campo `address_source` en la credencial.
4. Si `address_source` es `pool`, OmniTWAG pre-asigna una dirección fija para la sesión desde el grupo.
5. Si `address_source` es `dhcp`, OmniTWAG espera un DHCP DISCOVER del UE. El servidor DHCP luego entrega una dirección del rango del grupo.
6. OmniTWAG proporciona al UE la dirección, la puerta de enlace y los servidores DNS del grupo.



Asignación de direcciones

OmniTWAG asigna hosts desde el CIDR del grupo. La asignación omite la dirección de red y la dirección de difusión. El primer host utilizable es la dirección de red más uno.

El prefijo CIDR del grupo debe ser `/30` o más grande para tener hosts utilizables. Un bloque `/30` tiene 2 hosts utilizables. Un prefijo más pequeño tiene menos hosts utilizables, por lo que OmniTWAG lo rechaza.

Cuando cada host utilizable está asignado, OmniTWAG devuelve `pool_exhausted`. Ninguna sesión adicional del punto de acceso vinculado puede obtener una dirección hasta que se libere una asignación.

El campo `allocated` en el grupo mantiene el conteo actual de asignaciones. OmniTWAG calcula este conteo a partir de las asignaciones en memoria.

Modelo de grupo

Cada grupo tiene estos campos.

Campo	Tipo	Descripción
`id`	String	Identificador único (UUID). El servidor establece este valor.
`name`	String	Etiqueta legible por humanos, por ejemplo `WiFi de invitados de Melbourne`.
`cidr`	String	Bloque CIDR IPv4 para el grupo, por ejemplo `10.60.0.0/24`. El prefijo debe ser `/30` o más grande para tener hosts utilizables.
`gateway`	String	Dirección de puerta de enlace IPv4 dada al UE. Por defecto es el primer host utilizable del CIDR si se omite.
`dns_servers`	Lista de String	Direcciones de servidores DNS IPv4 dadas al UE.
`lease_time`	Integer	Tiempo de arrendamiento en segundos. Por defecto es `3600`.
`default`	Boolean	Cuando es `true`, este grupo es el rango predeterminado de DHCP. El servidor DHCP entrega desde él para cualquier cliente sin arrendamiento pre-asignado y sin vinculación de grupo. Solo un grupo puede ser el predeterminado a la vez. Por defecto es `false`.
`allocated`	Integer	Conteo de asignaciones actuales. El servidor calcula este valor.
`inserted_at`	Integer	Tiempo en que se creó el grupo, en milisegundos desde la época Unix. El servidor establece este valor.

Campo	Tipo	Descripción
`updated_at`	Integer	Tiempo en que se cambió por última vez el grupo, en milisegundos desde la época Unix. El servidor establece este valor.

El operador establece `name`, `cidr`, `gateway`, `dns_servers` y `lease_time`. El servidor establece `id`, `allocated`, `inserted_at` y `updated_at`.

API REST

La API REST de OmniTWAG ofrece estos endpoints a través de HTTPS en el puerto API configurado (`8444` por defecto). La ruta base es `/ip_pools`.

Método	Ruta	Acción	Descripción
GET	`/ip_pools`	index	Lista todos los grupos. El resultado está paginado y se puede buscar por `name` o `cidr`.
GET	`/ip_pools/{id}`	show	Obtiene un grupo por `id`.
POST	`/ip_pools`	create	Crea un grupo.
PUT	`/ip_pools/{id}`	update	Cambia un grupo.
DELETE	`/ip_pools/{id}`	delete	Elimina un grupo.

Cuerpo de la solicitud

`POST` y `PUT` aceptan un cuerpo JSON con estos campos.

Campo	Tipo	Requerido	Descripción
`name`	String	No	Etiqueta legible por humanos.
`cidr`	String	Sí	Bloque CIDR IPv4. El prefijo debe ser `/30` o más grande.
`gateway`	String	No	Dirección de puerta de enlace IPv4. Por defecto es el primer host utilizable del CIDR si se omite.
`dns_servers`	Lista de String	No	Direcciones de servidores DNS IPv4.
`lease_time`	Integer	No	Tiempo de arrendamiento en segundos. Por defecto es `3600`.
`default`	Boolean	No	Marca este grupo como el rango predeterminado de DHCP. Configurarlo borra la bandera en cualquier otro grupo.

`PUT` solo cambia los campos en el cuerpo. Los campos que no están en el cuerpo mantienen su valor actual.

Respuestas

Estado	Significado
`200 OK`	La solicitud fue exitosa (index, show, update, delete).
`201 Created`	El grupo fue creado.
`404 Not Found`	Ningún grupo tiene el `id` dado.
`422 Unprocessable Entity`	El `cidr` no es un bloque CIDR IPv4 válido, o el prefijo es demasiado pequeño para tener hosts utilizables.

Ejemplo: listar grupos

Solicitud:

```
GET /ip_pools
```

Respuesta:

```
{
  "count": 1,
  "ip_pools": [
    {
      "id": "0d1e2f3a-4b5c-6d7e-8f90-1a2b3c4d5e6f",
      "name": "WiFi de invitados de Melbourne",
      "cidr": "10.60.0.0/24",
      "gateway": "10.60.0.1",
      "dns_servers": ["8.8.8.8", "8.8.4.4"],
      "lease_time": 3600,
      "allocated": 12,
      "inserted_at": 1734512400000,
      "updated_at": 1734512400000
    }
  ]
}
```

Ejemplo: crear un grupo

Solicitud:

```
POST /ip_pools
```

```
{
  "name": "WiFi de invitados de Melbourne",
  "cidr": "10.60.0.0/24",
  "gateway": "10.60.0.1",
  "dns_servers": ["8.8.8.8", "8.8.4.4"],
  "lease_time": 3600
}
```

Una solicitud exitosa devuelve `201 Created` con el nuevo grupo, incluyendo su `id`. Si se omite `gateway`, el grupo utiliza el primer host utilizable del CIDR (`10.60.0.1` en este ejemplo).

Ejemplo: un CIDR que es demasiado pequeño

Solicitud:

```
{
  "name": "Grupo roto",
  "cidr": "10.60.0.0/31"
}
```

Respuesta (`422 Unprocessable Entity`):

```
{
  "error": "cidr demasiado pequeño: 10.60.0.0/31 no tiene hosts
utilizables"
}
```

Grupo predeterminado

Un grupo puede ser el rango predeterminado de DHCP. El servidor DHCP entrega una dirección del grupo predeterminado para cualquier cliente que no tenga un arrendamiento pre-asignado y sin vinculación de grupo explícita. Utiliza el grupo predeterminado para dar un rango definido por la API al servicio DHCP base, no solo a las sesiones NSWO.

Establece el predeterminado con el campo `default`, o la casilla de verificación en la página web. Solo un grupo puede ser el predeterminado a la vez. Cuando marcas un grupo como el predeterminado, OmniTWAG borra la bandera en el grupo predeterminado anterior.

Cada arrendamiento activo registra su fuente. Consulta la vista de Arrendamientos IP: un arrendamiento muestra `PGW (tunneled)`, un `grupo NSWO` nombrado, o el `grupo predeterminado`.

Interfaz web

El panel de control tiene una página de administración de **Grupos de IP**. La página lista cada grupo con su nombre, CIDR, puerta de enlace y conteo asignado. La página tiene controles para añadir, editar y eliminar un grupo.

La antigua página de **Grupo DHCP** ahora está etiquetada como **Arrendamientos IP**. La página de Arrendamientos IP muestra los arrendamientos activos.

Resolución de problemas

Un grupo está agotado

Síntomas: Nuevas sesiones NSWO de un punto de acceso no pueden obtener una dirección. OmniTWAG devuelve `pool_exhausted`.

Posibles causas:

- Cada host utilizable en el CIDR del grupo está asignado.
- El CIDR del grupo es demasiado pequeño para el número de sesiones activas.

Resolución:

1. Verifica el conteo `allocated` contra el número de hosts utilizables en el CIDR.
2. Amplía el CIDR del grupo para añadir más hosts utilizables, o libera asignaciones obsoletas finalizando sesiones inactivas.
3. Confirma que el tiempo de arrendamiento no sea tan largo que las direcciones liberadas sigan asignadas.

Un punto de acceso no utiliza un grupo

Síntomas: Las sesiones de un punto de acceso no se dirigen desde ningún grupo de IP.

Posibles causas:

- El punto de acceso utiliza un desvío local propiedad del punto de acceso, no NSWO anclado en TWAG. Un punto de acceso de desvío local, propiedad del AP, dirige a sus propios clientes y no utiliza un grupo de OmniTWAG.

- La credencial del AP no tiene `ip_pool_id`, por lo que no hay grupo vinculado.

Resolución:

1. Confirma que el punto de acceso utiliza NSWO anclado en TWAG. Solo las sesiones NSWO utilizan un grupo.
2. Si el punto de acceso debe utilizar un grupo, establece `ip_pool_id` en su credencial RADIUS al `id` del grupo. Consulta Secretos Compartidos de RADIUS.

El UE no obtiene dirección

Síntomas: El UE se conecta pero no recibe ninguna dirección IPv4.

Posibles causas:

- La `address_source` de la credencial no coincide con cómo el UE solicita una dirección. Con `dhcp`, el UE debe enviar un DHCP DISCOVER. Con `pool`, OmniTWAG pre-asigna la dirección.
- El grupo vinculado está agotado.
- El `ip_pool_id` en la credencial apunta a un grupo que no existe.

Resolución:

1. Confirma que la `address_source` en la credencial coincide con el comportamiento del UE.
2. Confirma que el grupo vinculado tiene direcciones libres. Verifica el conteo `allocated`.
3. Confirma que el `ip_pool_id` en la credencial apunta a un grupo válido.

Resumen

Referencia de Métricas TWAG

Descripción general

TWAG expone métricas de Prometheus en el puerto **9568** (establecido por `prometheus.port`). Las métricas de tipo Gauge se actualizan mediante un sondeo cada 5 segundos. Las métricas de tipo Counter se emiten desde la ruta de código que las activa.

Tabla de Contenidos

- Endpoint de Métricas
- Métricas de Autenticación RADIUS
- Métricas de Autorización y Límite de Tasa RADIUS
- Métricas de Contabilidad RADIUS
- Métricas EAP
- Gauges de Cliente, AP y Sesión
- Métricas de Rendimiento de Datos
- Métricas de Sesión y Ruta de Datos TWAG
- Métricas de Carga en Línea Gy
- Métricas de Diámetro STa
- Métrica de Mensaje de Diámetro
- Métricas NSWO
- Métricas DHCP
- Métricas DHCPv6 y RA IPv6
- Métricas WLCP
- Métricas de VM Erlang
- Consultas de Ejemplo de Grafana
- Reglas de Alerta
- Páginas Relacionadas

Endpoint de Métricas

```
http://<twag-host>:9568/metrics
```

Prometheus aplana cada nombre de evento en un nombre de métrica reemplazando los puntos por guiones bajos. Por ejemplo, el evento `[:radius, :access, :request, :count]` se expone como `radius_access_request_count`.

Métricas de Autenticación RADIUS

Estos contadores rastrean los flujos de Solicitudes/Respuestas de Acceso RADIUS por RFC 2865.

Métrica	Tipo	Etiquetas	Descripción
`radius_access_request_count`	Counter	none	Paquetes de Solicitud de Acceso RADIUS recibidos.
`radius_access_accept_count`	Counter	none	Paquetes de Aceptación de Acceso RADIUS enviados (autenticación exitosa).
`radius_access_reject_count`	Counter	none	Paquetes de Rechazo de Acceso RADIUS enviados (autenticación fallida).
`radius_access_challenge_count`	Counter	none	Paquetes de Desaf►►o de Acceso RADIUS enviados (intercambio EAP en progreso).

Consultas de ejemplo:

```
# Tasa de solicitudes por segundo
rate(radius_access_request_count[5m])

# Tasa de desafíos (negociaciones EAP activas)
rate(radius_access_challenge_count[5m])

# Proporción de éxito de autenticación
sum(rate(radius_access_accept_count[5m])) /
  (sum(rate(radius_access_accept_count[5m])) +
sum(rate(radius_access_reject_count[5m])))
```

Métricas de Autorización y Límite de Tasa RADIUS

Estos contadores rastrean paquetes que son descartados o marcados antes o durante la autenticación. Ver Rechazos RADIUS y Secretos Compartidos RADIUS.

Métrica	Tipo	Etiquetas	
`radius_auth_unauthorized_source_count`	Counter	`source`	Pac ori cor cre des es
`radius_auth_bad_secret_count`	Counter	`source`, `credential_id`	Pac coi cre ver Aut cor
`radius_access_rate_limited_count`	Counter	`nas_ip`	Sol des NA `ra` en

Consultas de ejemplo:

```
# APs desconocidos apareciendo (agregar una credencial para estos)
increase(radius_auth_unauthorized_source_count[1h])

# APs con un secreto compartido incorrecto
sum by (source) (increase(radius_auth_bad_secret_count[1h]))

# Dispositivos NAS limitados por tasa
sum by (nas_ip) (increase(radius_access_rate_limited_count[15m]))
```

Métricas de Contabilidad RADIUS

Rastrea los mensajes de Contabilidad RADIUS por RFC 2866.

radius_accounting_request_count

Tipo: Counter

Etiqueta	Descripción
`status_type`	Tipo de estado de contabilidad (ver abajo).

Valores de tipo de estado: `start`, `stop`, `interim_update`, `accounting_on`, `accounting_off`.

Consultas de ejemplo:

```
# Inicios de sesión por minuto
rate(radius_accounting_request_count{status_type="start"}[5m]) *
60

# Detenciones de sesión por minuto
rate(radius_accounting_request_count{status_type="stop"}[5m]) * 60
```

Métricas EAP

Estos contadores rastrean las fases EAP-AKA por RFC 4187, además de EAP-SIM, EAP re-autenticación (ERP) y fallos genéricos de EAP.

Métrica	Tipo	Descripción
`eap_aka_identity_count`	Counter	Intercambios de identidad EAP-AKA (Fase 1).
`eap_aka_challenge_count`	Counter	Intercambios de desafío EAP-AKA (Fase 2).
`eap_aka_sync_failure_count`	Counter	Fallos de sincronización EAP-AKA (SQN fuera de sincronización, desencadena resincronización).
`eap_aka_auth_success_count`	Counter	Autenticaciones EAP-AKA exitosas.
`eap_aka_auth_reject_count`	Counter	Autenticaciones EAP-AKA rechazadas (UE no pudo autenticar la red / fallo AUTN).
`eap_aka_client_error_count`	Counter	Notificaciones de error de cliente EAP-AKA recibidas del UE.
`eap_auth_rejected_count`	Counter	Negaciones de autenticación esperadas (por ejemplo, HSS USER_UNKNOWN) que rechazan al UE limpiamente.
`eap_process_crash_count`	Counter	Errores inesperados al procesar un mensaje EAP (UE rechazado, sesión preservada).
`eap_sim_start_count`	Counter	Intercambios de inicio EAP-SIM.
`eap_sim_auth_success_count`	Counter	Autenticaciones EAP-SIM exitosas.

Métrica	Tipo	Descripción
`eap_sim_auth_reject_count`	Counter	Autenticaciones EAP-SIM rechazadas.
`eap_erp_initiate_count`	Counter	Iniciaciones de re-autenticación EAP ERP.
`eap_erp_success_count`	Counter	Re-autenticaciones EAP ERP exitosas.
`eap_erp_mac_failure_count`	Counter	Fallos de verificación MAC EAP ERP.
`eap_erp_fallback_count`	Counter	Reversiones EAP ERP a autenticación completa. Etiqueta `reason`.

Consultas de ejemplo:

```
# Fallos de sincronización (deberían ser raros - alerta si > 0)
increase(eap_aka_sync_failure_count[1h])

# Autenticaciones AKA exitosas por minuto
rate(eap_aka_auth_success_count[5m]) * 60

# Fallos de procesamiento de EAP (deberían mantenerse estables)
increase(eap_process_crash_count[1h])
```

Gauges de Cliente, AP y Sesión

Estos gauges se muestrean cada 5 segundos por el sondeo.

Métrica	Tipo	Etiquetas	Descripción
`radius_active_clients_count`	Gauge	none	Clientes actualmen autenticados (procesos `Twag.Session.Clie` en vivo).
`radius_access_points_count`	Gauge	none	Puntos de acceso registrados (entrad `APState`).
`radius_clients_per_ap_count`	Gauge	`nas_ip`, `ssid`	Clientes conectados por punto de acces
`radius_active_sessions_count`	Gauge	`status`	Sesiones agrupadas por estado de contabilidad (`start` `stop`, `interim_update`).
`radius_active_sessions_total`	Gauge	none	Sesiones en estado `start` o `interim_update`.

Consultas de ejemplo:

```
# Los 5 APs más ocupados
topk(5, radius_clients_per_ap_count)

# Clientes por SSID
sum by (ssid) (radius_clients_per_ap_count)

# Alerta sobre una caída repentina en sesiones activas
delta(radius_active_sessions_total[5m]) < -10
```

Métricas de Rendimiento de Datos

Estos gauges se suman a través de todas las sesiones rastreadas desde los contadores de contabilidad RADIUS.

Métrica	Tipo	Unidad	Descripción
`radius_data_bytes_in`	Gauge	bytes	Total de bytes recibidos a través de sesiones (Acct-Input-Octets).
`radius_data_bytes_out`	Gauge	bytes	Total de bytes enviados a través de sesiones (Acct-Output-Octets).
`radius_session_total_time`	Gauge	seconds	Tiempo total de sesión a través de sesiones (Acct-Session-Time).

Consultas de ejemplo:

```
# Rendimiento total en MB
(radius_data_bytes_in + radius_data_bytes_out) / 1024 / 1024

# Duración promedio de sesión
radius_session_total_time / radius_active_sessions_total
```

Métricas de Sesión y Ruta de Datos TWAG

Estos contadores rastrean eventos del ciclo de vida por sesión emitidos por `Twag.Session.Client`.

Métrica	Tipo	Etiquetas	Des
`twag_session_teardown_count`	Counter	`reason`	Termina sesión. `account` `admin_c` `network` `identit` `livenes`
`twag_bearer_modify_command_count`	Counter	none	Eventos Modify-E Comma a una se
`twag_bearer_update_count`	Counter	none	Eventos Update- aplicado sesión.
`twag_detach_cleanup_start_count`	Counter	none	Inicios d separac
`twag_detach_cleanup_complete_count`	Counter	none	Finaliza limpieza separac
`twag_detach_pgw_delete_count`	Counter	none	Solicitud Delete-S durante separac
`twag_detach_liveness_timeout_count`	Counter	none	Separac desenca un tiem de activ

Métrica	Tipo	Etiquetas	Des
`twag_liveness_arp_probe_success_count`	Counter	none	Probes A actividad
`twag_liveness_arp_probe_failure_count`	Counter	none	Probes A actividad
`twag_handover_in_count`	Counter	`source`	Eventos transfer entrada
`twag_handover_out_count`	Counter	`cause`	Eventos transfer salida.
`twag_handover_ap_change_count`	Counter	none	Eventos de AP in

Consultas de ejemplo:

```
# Razones de terminación en la última hora
sum by (reason) (increase(twag_session_teardown_count[1h]))

# Sesiones eliminadas por no completar la autenticación
increase(twag_session_teardown_count{reason="identity_timeout"}
[1h])

# Sesiones descartadas porque el UE se quedó en silencio
increase(twag_detach_liveness_timeout_count[1h])
```

> Nota: los eventos de cuota Gy `twag_gy_quota_blocked` y `twag_gy_quota_unblocked` son emitidos por `Twag.Session.Client` pero no están declarados como métricas de Prometheus hoy, por lo que no aparecen en el endpoint `/metrics`. Son visibles en los registros estructurados.

Métricas de Carga en Línea Gy

Estos contadores rastrean el control de crédito Diameter Ro/Gy, utilizado solo para el desglose local NSWO. Ver Carga en Línea.

Métrica	Tipo	Etiquetas	Descripción
`twag_gy_ccr_sent_count`	Counter	`type`	Solicitudes de Control de Crédito enviadas al OCS. `type` es `initial`, `update` o `terminate`.
`twag_gy_ccr_error_count`	Counter	`type`	Fallos en el envío o construcción de CCR.
`twag_gy_cca_received_count`	Counter	`type`, `result`	Respuestas de Control de Crédito recibidas del OCS.

Consultas de ejemplo:

```
# Tasa de envío de CCR por tipo
sum by (type) (rate(twag_gy_ccr_sent_count[5m]))

# Fallos de CCR
sum by (type) (increase(twag_gy_ccr_error_count[1h]))

# Resultados de CCA
sum by (result) (rate(twag_gy_cca_received_count[5m]))
```

Métricas de Diámetro STa

Estos contadores rastrean las solicitudes entrantes STa manejadas por el TWAG.

Métrica	Tipo	Etiquetas	Descripción
`sta_asr_received_count`	Counter	none	Solicitudes de Abort-Session recibidas.
`sta_rar_received_count`	Counter	none	Solicitudes de Re-Auth recibidas.
`sta_str_received_count`	Counter	none	Solicitudes de Session-Termination recibidas.
`sta_session_disconnect_count`	Counter	`reason`	Desconexiones de sesión STa.
`sta_session_reauth_count`	Counter	`re_auth_type`	Re-autenticaciones de sesión STa.

Métrica de Mensaje de Diámetro

diameter_message_count

Tipo: Counter

Etiqueta	Descripción
`application`	Aplicación de Diámetro (por ejemplo, `swx`, `sta`).
`command`	Nombre del comando (por ejemplo, `MAR`, `MAA`).
`direction`	`inbound` o `outbound`.

Consultas de ejemplo:

```
# Tasa de mensajes de Diámetro por aplicación
sum by (application) (rate(diameter_message_count[5m]))
```

Métricas NSWO

Métrica	Tipo	Descripción
`nswo_allocate_count`	Counter	Asignaciones de IP locales NSWO.
`nswo_release_count`	Counter	Liberaciones de IP locales NSWO.

Métricas DHCP

Estos contadores rastrean el servidor DHCPv4. El evento `discover.pool_allocated` cubre la ruta de entrega NSWO, donde el servidor asigna una dirección fresca de un pool vinculado en DISCOVER.

Métrica	Tipo	Descripción
`dhcp_lease_allocate_count`	Counter	Asignaciones de arrendamiento.
`dhcp_lease_deallocate_count`	Counter	Desasignaciones de arrendamiento.
`dhcp_discover_count`	Counter	Paquetes DISCOVER servidos desde un arrendamiento preasignado.
`dhcp_discover_no_allocation_count`	Counter	DISCOVERs sin arrendamiento y sin pool vinculado (sin dirección para ofrecer).
`dhcp_discover_pool_allocated_count`	Counter	DISCOVERs servidos al entregar una dirección fresca de un pool vinculado (NSWO).
`dhcp_ack_count`	Counter	Paquetes ACK enviados.
`dhcp_nak_count`	Counter	Paquetes NAK enviados.
`dhcp_release_count`	Counter	Paquetes RELEASE recibidos.
`dhcp_decline_count`	Counter	Paquetes DECLINE recibidos.

Métrica	Tipo	Descripción
`dhcp_inform_count`	Counter	Paquetes INFORM recibidos.
`dhcp_packet_malformed_count`	Counter	Paquetes DHCP malformados.

Consultas de ejemplo:

```
# Tasa de entrega de pool (direccionamiento NSWO)
rate(dhcp_discover_pool_allocated_count[5m])

# DISCOVERs que no pudieron ser atendidos
increase(dhcp_discover_no_allocation_count[15m])
```

> Nota: el servidor DHCP también emite `dhcp_lease_reap` cuando recoge un arrendamiento expirado. Este evento no está declarado como una métrica de Prometheus hoy y no aparece en el endpoint `/metrics`.

Métricas DHCPv6 y RA IPv6

Métrica	Tipo	Descripción
`dhcpv6_info_request_received_count`	Counter	Paquetes de Solicitud de Información DHCPv6 recibidos.
`dhcpv6_solicit_received_count`	Counter	Paquetes de Solicitud DHCPv6 recibidos.
`dhcpv6_request_received_count`	Counter	Paquetes de Solicitud DHCPv6 recibidos.
`dhcpv6_info_reply_sent_count`	Counter	Paquetes de Respuesta de Información DHCPv6 enviados.
`dhcpv6_advertise_sent_count`	Counter	Paquetes de Publicidad DHCPv6 enviados.
`dhcpv6_reply_sent_count`	Counter	Paquetes de Respuesta DHCPv6 enviados.
`ipv6_ra_prefix_add_count`	Counter	Adiciones de prefijo RA IPv6.
`ipv6_ra_prefix_remove_count`	Counter	Eliminaciones de prefijo RA IPv6.
`ipv6_ra_advertisement_sent_count`	Counter	Publicidades de Router IPv6 enviadas.

Métricas WLCP

Estos contadores rastrean el Protocolo de Control WLAN utilizado para el modo de conexión múltiple.

Métrica	Tipo	Unidad	Descripción
`wlcp_packet_received_count`	Counter	none	Paquetes WLCP recibidos.
`wlcp_packet_received_bytes`	Sum	bytes	Total de bytes de paquetes WLCP recibidos.
`wlcp_packet_malformed_count`	Counter	none	Paquetes WLCP malformados.
`wlcp_packet_unregistered_count`	Counter	none	Paquetes WLCP de clientes no registrados.
`wlcp_pdn_request_count`	Counter	none	Solicitudes de conexión PDN.
`wlcp_pdn_accept_count`	Counter	none	Aceptaciones de conexión PDN.
`wlcp_pdn_reject_count`	Counter	none	Rechazos de conexión PDN.
`wlcp_pdn_disconnect_request_count`	Counter	none	Solicitudes de desconexión PDN.

Métrica	Tipo	Unidad	Descripción
`wlcp_pdn_disconnect_accept_count`	Counter	none	Aceptaciones de desconexión PDN.

Métricas de VM Erlang

Estas métricas ofrecen visibilidad sobre el tiempo de ejecución BEAM.

Memoria

Métrica	Descripción
`vm_memory_total`	Memoria total asignada (bytes).
`vm_memory_processes`	Memoria utilizada por procesos Erlang.
`vm_memory_processes_used`	Memoria activamente utilizada por procesos.
`vm_memory_system`	Memoria utilizada por el tiempo de ejecución.
`vm_memory_atom`	Memoria utilizada por átomos.
`vm_memory_atom_used`	Memoria activamente utilizada por átomos.
`vm_memory_binary`	Memoria utilizada por binarios.
`vm_memory_code`	Memoria utilizada por código cargado.
`vm_memory_ets`	Memoria utilizada por tablas ETS (estado de sesión y AP).

Sistema

Métrica	Descripción
`vm_system_info_process_count`	Número actual de procesos Erlang.
`vm_system_info_port_count`	Número actual de puertos.
`vm_system_info_atom_count`	Número actual de átomos.
`vm_system_info_schedulers`	Total de hilos de programador.
`vm_system_info_schedulers_online`	Hilos de programador en línea.

Programadores

Métrica	Descripción
`vm_statistics_run_queue`	Longitud total de la cola de ejecución.
`vm_total_run_queue_lengths_total`	Total de la cola de ejecución del programador.
`vm_total_run_queue_lengths_cpu`	Cola de ejecución del programador de CPU.
`vm_total_run_queue_lengths_io`	Cola de ejecución del programador de IO.

Consultas de ejemplo:

```
# Memoria total en MB
vm_memory_total / 1024 / 1024

# Saturación de la cola de ejecución (debería ser baja)
avg_over_time(vm_statistics_run_queue[5m])
```

Consultas de Ejemplo de Grafana

Descripción General de Autenticación

```
# Aceptaciones por segundo
sum(rate(radius_access_accept_count[5m]))

# Rechazos por segundo
sum(rate(radius_access_reject_count[5m]))

# Porcentaje de tasa de éxito
100 * sum(rate(radius_access_accept_count[5m])) /
  (sum(rate(radius_access_accept_count[5m])) +
sum(rate(radius_access_reject_count[5m])))
```

Estadísticas por AP

```
# Solicitudes por AP (Access-Request no tiene etiqueta nas_ip;
usar contabilidad en su lugar)
sum by (nas_ip) (rate(radius_accounting_request_count[5m]))

# Clientes por AP
radius_clients_per_ap_count
```

Análisis de Rechazo y Límite de Tasa

```
# Rechazos de autenticación por clase de razón
sum(increase(radius_access_reject_count[1h]))
sum(increase(radius_auth_unauthorized_source_count[1h]))
sum(increase(radius_auth_bad_secret_count[1h]))

# Razones de terminación
sum by (reason) (increase(twag_session_teardown_count[1h]))
```

Uso de Datos

```
# Rendimiento total en MB
(radius_data_bytes_in + radius_data_bytes_out) / 1024 / 1024
```

Reglas de Alerta

Alertas Críticas

```
groups:
  - name: twag_critical
    rules:
      - alert: HighAuthenticationFailureRate
        expr: >
          sum(rate(radius_access_reject_count[5m])) /
          (sum(rate(radius_access_accept_count[5m])) +
sum(rate(radius_access_reject_count[5m]))) > 0.1
        for: 5m
        labels:
          severity: critical
        annotations:
          summary: "Alta tasa de fallos de autenticación (> 10%)"

      - alert: EAPSyncFailures
        expr: increase(eap_aka_sync_failure_count[5m]) > 0
        for: 1m
        labels:
          severity: warning
        annotations:
          summary: "Se detectaron fallos de sincronización EAP-
AKA"

      - alert: UnknownRadiusSources
        expr: increase(radius_auth_unauthorized_source_count[15m])
> 0
        for: 5m
        labels:
          severity: warning
        annotations:
          summary: "Paquetes RADIUS de fuentes sin credenciales
coincidentes"

      - alert: BadRadiusSecret
        expr: increase(radius_auth_bad_secret_count[15m]) > 0
        for: 5m
        labels:
          severity: warning
```

```
        annotations:
          summary: "Paquetes RADIUS rechazados por un secreto
compartido incorrecto"

      - alert: NoActiveClients
        expr: radius_active_clients_count == 0
        for: 15m
        labels:
          severity: warning
        annotations:
          summary: "Sin clientes activos durante 15 minutos"
```

Alertas de Recursos

```
groups:
  - name: twag_resources
    rules:
      - alert: HighMemoryUsage
        expr: vm_memory_total > 1073741824
        for: 10m
        labels:
          severity: warning
        annotations:
          summary: "El uso de memoria de TWAG supera 1GB"

      - alert: HighRunQueue
        expr: avg_over_time(vm_statistics_run_queue[5m]) > 10
        for: 5m
        labels:
          severity: warning
        annotations:
          summary: "Alta cola de ejecución del programador indica
saturación de CPU"
```

Páginas Relacionadas

- Arquitectura
- Rechazos RADIUS
- Secretos Compartidos RADIUS

- Sesiones
- Carga en Línea
- Contabilidad por AP

Descripción general

Carga en Línea (Gy)

Resumen

OmniTWAG puede cargar sesiones de datos en tiempo real contra un Sistema de Carga en Línea (OCS). OmniTWAG utiliza la interfaz Diameter Ro (Gy) para enviar Solicitudes de Control de Crédito (CCR) y recibir Respuestas de Control de Crédito (CCA). Esto permite a un operador aplicar control de crédito prepago al tráfico WLAN que el TWAG sirve.

La carga en línea se aplica a sesiones donde el TWAG posee la ruta de datos, es decir, **NSWO (salida local)** y tráfico anclado en TWAG. En este caso, el TWAG es el punto donde el tráfico sale a Internet, por lo que el TWAG actúa como el punto de aplicación de carga. En **modo de enrutamiento en casa (GTP)**, el PGW realiza la carga, por lo que un operador normalmente deja el modo de carga global `:disabled`. Consulte la referencia de Plano de Usuario para los modos de plano de datos.

Dos controles deciden si Gy se ejecuta para un UE dado:

- El **modo de carga global** (`online_charging.mode`). Esto selecciona qué Solicitudes de Control de Crédito OmniTWAG puede enviar.
- El **modo de carga por AP** en la credencial del AP (`charging_mode`). Esto determina, por punto de acceso, si OmniTWAG abre una sesión Gy para ese UE. Consulte Credenciales RADIUS.

Ambos controles deben permitir la carga para que un CCR-Initial salga. Consulte Modo de Carga por AP.

Tabla de Contenidos

- Conceptos
- Modos de Carga
- Modo de Carga por AP
- Flujo de Control de Crédito
- Cuota, Uso y Reautorización

- Aplicación de Cuota
- Grupo de Calificación y Unidades de Servicio
- Configuración
- Comportamiento Operacional y Manejo de Fallas
- Observabilidad
- Tablas de Referencia

Conceptos

La interfaz Gy es una aplicación Diameter que transporta mensajes de control de crédito entre un elemento de red y un OCS. OmniTWAG utiliza el diccionario Ro de 3GPP con Auth-Application-Id `4`, como se define en RFC 4006 y 3GPP TS 32.299. OmniTWAG utiliza el Service-Context-Id `32251@3gpp.org`, que es el contexto de carga PS de 3GPP.

OmniTWAG envía tres tipos de Solicitudes de Control de Crédito durante una sesión:

- **CCR-Initial (CCR-I)**: al inicio de la sesión. OmniTWAG pide al OCS que inicie el control de crédito para el suscriptor.
- **CCR-Update (CCR-U)**: durante la sesión. OmniTWAG informa los datos utilizados y pide más cuota.
- **CCR-Termination (CCR-T)**: al final de la sesión. OmniTWAG informa los datos finales utilizados y cierra la sesión de control de crédito.

El OCS responde a cada solicitud con una Respuesta de Control de Crédito (CCA). Un CCA lleva un `Result-Code`, una concesión opcional de cuota (`Granted-Service-Unit`), y un `Validity-Time` opcional.

OmniTWAG mantiene un estado Gy separado para cada sesión de suscriptor. El estado Gy contiene el `Session-Id` de Diameter, el `CC-Request-Number` actual, los contadores de uso acumulativo, y la última cuota que el OCS otorgó. OmniTWAG crea este estado cuando envía el CCR-I, y descarta el estado cuando la sesión termina.

Modos de Carga

El parámetro `online_charging.mode` es el modo de carga **global**. Selecciona uno de tres modos y controla qué Solicitudes de Control de Crédito OmniTWAG puede enviar. El `charging_mode` por AP (ver Modo de Carga por AP) luego decide, para cada UE, si OmniTWAG abre una sesión Gy o no.

Modo	CCR-I al inicio	CCR-U durante la sesión	CCR-T al final
`:disabled`	No	No	No
`:authorize_only`	Sí	No	Sí
`:online_charging`	Sí	Sí	Sí

`:disabled`

OmniTWAG no contacta al OCS. Este es el valor predeterminado. Utilice este modo para la operación de Enrutamiento en Casa (GTP), donde el PGW realiza la carga.

`:authorize_only`

OmniTWAG envía un CCR-I al inicio de la sesión y un CCR-T al final de la sesión. OmniTWAG no envía mensajes interinos CCR-U. Utilice este modo para verificar el saldo del suscriptor al inicio y para informar el uso total al final, sin gestión de cuota en vivo.

`:online_charging`

OmniTWAG envía un CCR-I al inicio, un CCR-U en cada actualización interina de contabilidad, y un CCR-T al final. Utilice este modo para la gestión completa de cuota, donde el OCS otorga y refresca la cuota durante la sesión.

Modo de Carga por AP

Cada credencial de AP lleva un campo `charging_mode`. Esto es independiente del `online_charging.mode` global. OmniTWAG resuelve la credencial desde la fuente RADIUS de la sesión, por lo que cada punto de acceso puede tener su propio comportamiento. Consulte Credenciales RADIUS.

`charging_mode` por AP	Significado	Efecto en Gy
`:charged`	Cargar el tráfico de este AP. Este es el valor predeterminado.	OmniTWAG abre una sesión Gy (CCR-Initial) para el UE cuando el modo global también lo permite.
`:uncharged`	Servir este AP pero no cargarlo.	OmniTWAG establece la sesión de datos pero no abre ninguna sesión Gy. No se envía ningún CCR.
`:auth_only`	Solo autenticar, sin sesión de datos.	OmniTWAG autentica al suscriptor y se detiene. No construye ninguna sesión de datos y no abre ninguna sesión Gy.

OmniTWAG envía un CCR-Initial solo cuando **todas** estas condiciones se cumplen:

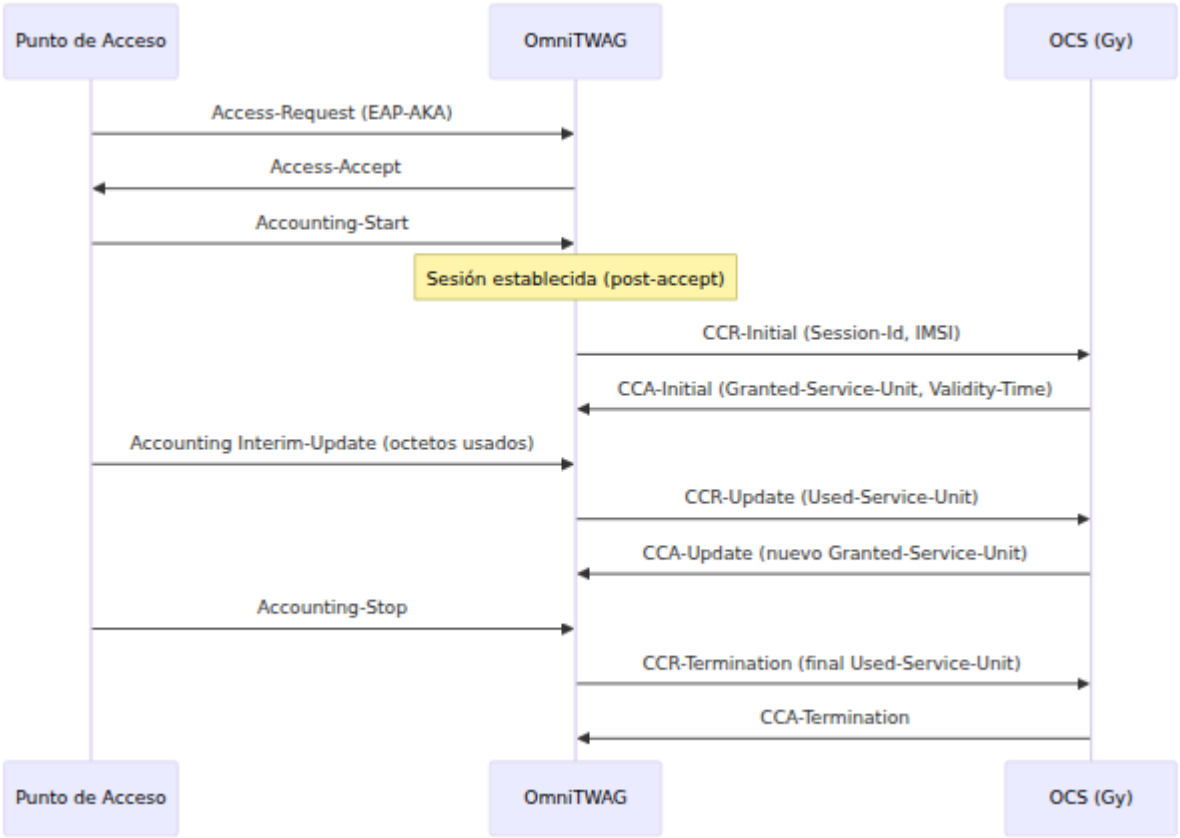
- El `online_charging.mode` global no es `:disabled` (el cliente Gy está habilitado).
- La sesión tiene un IMSI.
- La credencial del AP `charging_mode` es `:charged`.

Si alguna condición falla, OmniTWAG no abre ninguna sesión Gy para el UE, y no se envía ningún CCR-U o CCR-T posterior. El modo por AP, por lo tanto, controla Gy independientemente del modo global: incluso con el modo global

configurado en `:online_charging`, un AP marcado como `:uncharged` no produce tráfico Gy.

Flujo de Control de Crédito

La sesión de control de crédito sigue a la sesión del suscriptor. OmniTWAG activa cada Solicitud de Control de Crédito desde un punto específico en el ciclo de vida de la sesión RADIUS.



Inicio de sesión (CCR-Initial)

OmniTWAG envía el CCR-I después de establecer la sesión. El establecimiento de la sesión se ejecuta en el paso post-accept, que el servidor RADIUS activa después del Access-Accept y el Accounting-Start. OmniTWAG envía el CCR-I solo cuando el modo de carga global está habilitado, la sesión tiene un IMSI, y la credencial del AP `charging_mode` es `:charged`. Consulte Modo de Carga por AP.

El CCR-I lleva el IMSI del suscriptor en un AVP `Subscription-ID` con tipo `1` (END_USER_IMSI). El CCR-I establece `Multiple-Services-Indicator` en `1` e incluye un grupo `Multiple-Services-Credit-Control` (MSCC) vacío y el `Rating-Group` configurado.

Actualizaciones interinas (CCR-Update)

En modo `:online_charging`, cada actualización interina de contabilidad RADIUS incorpora los octetos informados y el tiempo de sesión en los contadores de uso de Gy. OmniTWAG no envía un CCR-U en cada interino. Envía un CCR-U solo cuando se activa uno de estos desencadenantes:

- El volumen utilizado ha cruzado la fracción configurada del volumen otorgado (el `quota.update_threshold`, predeterminado `0.8`).
- El `Validity-Time` de la cuota otorgada ha expirado.

Un interino que no cruza un desencadenante solo registra el uso localmente, por lo que no se consulta al OCS en cada interino. Cuando un CCR-U se envía, OmniTWAG lee los octetos utilizados y el tiempo de sesión de los atributos de contabilidad y los informa en un AVP `Used-Service-Unit`. El CCR-U también incluye un `Requested-Service-Unit` vacío, para que el OCS pueda otorgar más cuota.

Un CCR-U también se envía fuera de la ruta de contabilidad cuando se activa el temporizador de validez de cuota. Consulte Aplicación de Cuota.

OmniTWAG lee estos atributos de contabilidad RADIUS para el CCR-U:

Atributo RADIUS	Campo Gy
`Acct-Input-Octets`	`CC-Input-Octets`
`Acct-Output-Octets`	`CC-Output-Octets`
`Acct-Session-Time`	`CC-Time`

OmniTWAG calcula `CC-Total-Octets` como la suma de los octetos de entrada y salida.

Fin de sesión (CCR-Termination)

OmniTWAG envía un CCR-T cuando la sesión se desmantela, por ejemplo, en un RADIUS Accounting-Stop. OmniTWAG informa el uso acumulativo final en un AVP `Used-Service-Unit` y establece el `Termination-Cause` en `1` (DIAMETER_LOGOUT). OmniTWAG envía el CCR-T tanto en modo `:authorize_only` como en modo `:online_charging`.

Cuota, Uso y Reautorización

Cuota otorgada

Cada CCA puede llevar un `Granted-Service-Unit` dentro del primer grupo MSCC. OmniTWAG almacena los valores otorgados en el estado Gy:

- `CC-Total-Octets`, `CC-Input-Octets`, `CC-Output-Octets`: el volumen de datos otorgado.
- `CC-Time`: el tiempo otorgado.
- `Validity-Time`: el tiempo en segundos que la concesión permanece válida. OmniTWAG convierte esto en un plazo absoluto y activa un temporizador contra él.

OmniTWAG también lee el `Final-Unit-Indication` de MSCC y el `Result-Code` a nivel de MSCC de cada CCA. Una `Final-Unit-Action` de TERMINATE, REDIRECT o RESTRICT_ACCESS, o un `Result-Code` de MSCC de `4010` (END_USER_SERVICE_DENIED) o `4012` (CREDIT_LIMIT_REACHED), se almacena como una señal de agotamiento de crédito. Consulte Aplicación de Cuota.

OmniTWAG actualiza la concesión almacenada en cada CCA-I y CCA-U exitosos.

Reporte de uso

OmniTWAG mantiene contadores de uso acumulativo en el estado Gy. Los contadores de contabilidad RADIUS son acumulativos para la sesión, por lo que cada actualización interina informa el total hasta el momento. OmniTWAG registra los contadores informados y los envía como el `Used-Service-Unit`. El

CCR-U establece el `Reporting-Reason` en `4` (VALIDITY_TIME). El CCR-T establece el `Reporting-Reason` en `2` (FINAL).

Reautorización

OmniTWAG reautoriza en dos caminos:

- **En tráfico.** Una actualización interina de contabilidad RADIUS que cruza el umbral de cuota o encuentra la validez expirada activa un CCR-U. El intervalo interino es una propiedad del punto de acceso, por lo que un intervalo interino más corto proporciona una medición más precisa.
- **En tiempo.** OmniTWAG activa un temporizador de validez de cuota a partir del `Validity-Time` otorgado. Si el temporizador se activa con poco o ningún tráfico, OmniTWAG aún envía un CCR-U para reautorizar. Si el OCS no envía `Validity-Time`, OmniTWAG puede recurrir al `quota.validity_time` configurado. Sin ninguno presente, no se activa ningún temporizador de validez.

El `CC-Request-Number` aumenta en uno después de cada CCA exitoso. Este valor otorga a cada solicitud en la sesión de control de crédito un número de secuencia único, como se requiere en RFC 4006.

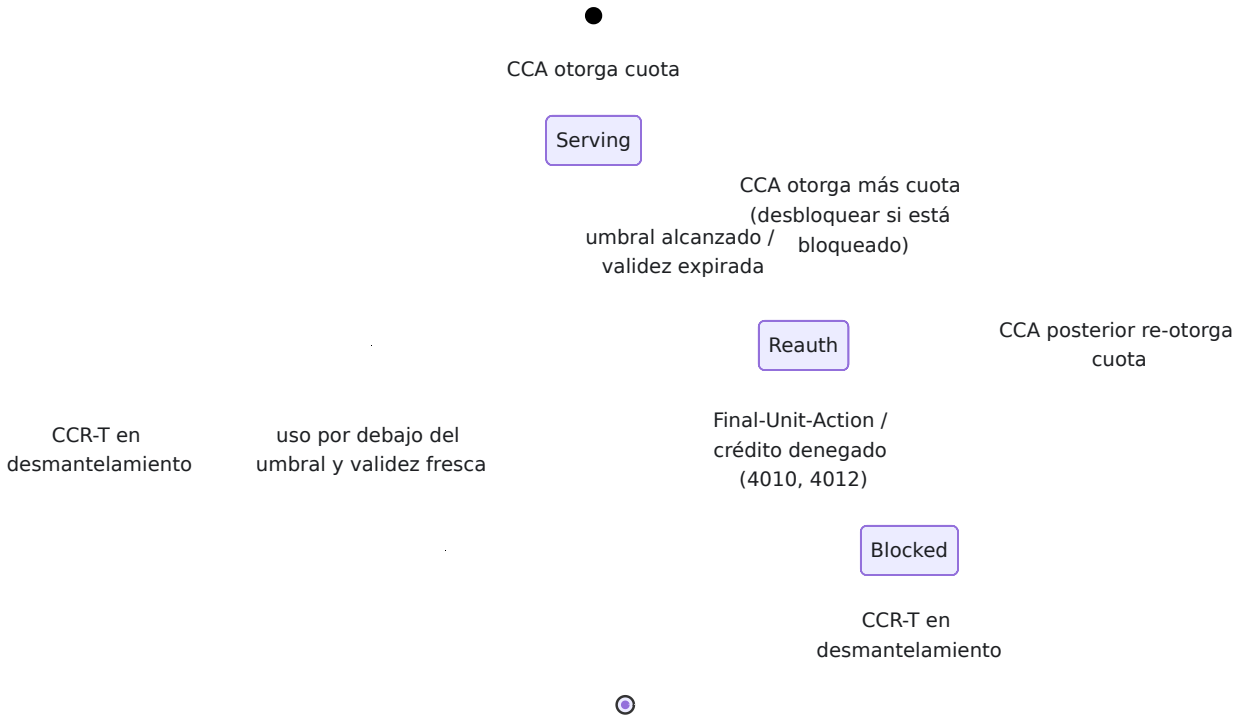
Aplicación de Cuota

OmniTWAG aplica el agotamiento de crédito en modo `:online_charging`. El TWAG no tiene una función de plano de usuario separada, por lo que su palanca de aplicación es la ruta de datos GRE: desregistra al UE de la ruta de datos, lo que elimina el enlace ascendente y descendente para ese UE.

Una respuesta CCR-U activa la aplicación cuando el estado Gy muestra crédito agotado, es decir, cuando el OCS:

- señala cualquier `Final-Unit-Action` (TERMINATE, REDIRECT o RESTRICT_ACCESS), o
- devuelve un `Result-Code` de MSCC de `4010` o `4012`.

Dado que el TWAG no puede honrar REDIRECT o RESTRICT_ACCESS en su ruta de datos, cualquier `Final-Unit-Action` se trata como una detención. Cuando se activa la aplicación, OmniTWAG bloquea la ruta de datos del UE pero mantiene la sesión, el arrendamiento IP y la contabilidad activos. Una reautorización posterior que otorgue cuota utilizable restaura la ruta de datos.



La aplicación es de mejor esfuerzo y falla abierta en errores de transporte: un CCR-U que falla en la capa Diameter nunca bloquea al UE. El bloqueo ocurre solo en una denegación de crédito explícita del OCS o señal de agotamiento. Consulte Comportamiento Operacional y Manejo de Fallas.

Grupo de Calificación y Unidades de Servicio

OmniTWAG utiliza un solo grupo de calificación para toda la sesión. El parámetro `rating_group` establece el valor `Rating-Group` en el MSCC del CCR-I. El valor predeterminado es `1`. OmniTWAG no divide el tráfico en múltiples grupos de calificación o identificadores de servicio. OmniTWAG solicita una concesión de volumen combinada con un `Requested-Service-Unit` vacío, por lo que el OCS decide la concesión.

Configuración

Establezca los parámetros de carga en línea bajo la aplicación `:omnitwag`, en el bloque `online_charging` de `config/runtime.exs`.

```
config :omnitwag,
  online_charging: %{
    # Modo de carga global: :disabled, :authorize_only, o
:online_charging
    mode: :disabled,

    # Dominio Diameter del OCS (Destination-Realm para CCR)
    ocs_realm: "epc.mnc001.mcc001.3gppnetwork.org",

    # Grupo de calificación para el grupo Multiple-Services-
Credit-Control
    rating_group: 1,

    # Gestión de cuota (utilizada en modo :online_charging)
    quota: %{
      # Fracción del volumen otorgado en la que se activa un CCR-
Update
      update_threshold: 0.8,

      # Tiempo de validez de reserva (segundos) cuando el OCS no
envía Validity-Time
      validity_time: 3600
    }
  }
```

Parámetros

Parámetro	Tipo	Requerido	Predeterminado	

`mode`	Átomo	No	`:disabled`	Mo
				glo
				`:d`
				`:a`
				o
				`:o`
				Co
				Ca

`ocs_realm`	Cadena	No	Dominio Diameter	Do de Or es `De` Re CC es Or el Di
`rating_group`	Entero	No	`1`	Va `Gr` Or en CC
`quota.update_threshold`	Flotante	No	`0.8`	Fra de oto se ac

Parámetro	Tipo	Requerido	Predeterminado	
				int Up
`quota.validity_time`	Entero	No	(no establecido)	Va de se pa ter va OC `Va` no se ter res

El `charging_mode` por AP no es parte de este bloque. Vive en cada credencial de AP. Consulte Credenciales RADIUS.

El `Origin-Host` y `Origin-Realm` de Diameter para la sesión Gy provienen de la identidad Diameter en la configuración `:diameter_ex` (`host` y `realm`). El `Destination-Host` no está configurado. OmniTWAG aprende el `Destination-Host` de la `Origin-Host` del primer CCA, y agrega este AVP a los mensajes posteriores CCR-U y CCR-T.

Transporte Diameter y la aplicación Ro

El cliente Gy envía a través de la pila Diameter compartida. Configure el OCS como un par Diameter y habilite la aplicación `:ro`, en la configuración `:diameter_ex`.

```
config :diameter_ex,
  diameter: %{
    service_name: :omnitouch_twag,
    host: "amanaki",
    realm: "epc.mnc001.mcc001.3gppnetwork.org",
    request_timeout: 5000,
    applications: [
      %{
        application_name: :ro,
        application_dictionary: :diameter_gen_3gpp_ro,
        vendor_specific_application_ids: [
          %{vendor_id: 0, auth_application_id: 4,
acct_application_id: nil}
        ]
      }
    ],
    peers: [
      %{
        host: "ocs.epc.mnc001.mcc001.3gppnetwork.org",
        realm: "epc.mnc001.mcc001.3gppnetwork.org",
        ip: "10.179.2.233",
        port: 3868,
        transport: :diameter_tcp,
        initiate_connection: true
      }
    ]
  }
```

Parámetro	Tipo	Requerido	Predetermin
`service_name`	Átomo	Sí	-
`host`	Cadena	Sí	-
`realm`	Cadena	Sí	-
`request_timeout`	Entero	No	5000
`application_name`	Átomo	Sí	-
`application_dictionary`	Átomo	Sí	-

Parámetro	Tipo	Requerido	Predetermin
`vendor_specific_application_ids`	Lista	Sí	-
`peers`	Lista	Sí	-

Parámetros de Pares

Parámetro	Tipo	Requerido	Predeterminado	Des
`host`	Cadena	Sí	-	Ident Diam par (Debe con l ident OCS.
`realm`	Cadena	Sí	-	Dom Diam par. l en el del C
`ip`	Cadena	Sí	-	Direc OCS cone
`port`	Entero	No	3868	Puert Diam OCS. están 3868
`transport`	Átomo	No	`:diameter_tcp`	Proto trans Usar `:dia` para
`initiate_connection`	Booleano	No	falso	Cuar `true` Omn inicia

Parámetro	Tipo	Requerido	Predeterminado	Des
				cone el pa

> **Nota:** El bloque `online_charging` en el `config/runtime.exs` enviado también muestra `ocs_host`, `service_identifier`, un `request_timeout` de nivel superior, y las claves `quota` `initial_request` y `subsequent_request`. El código de carga actual no lee estas claves. Lee `mode`, `ocs_realm`, `rating_group`, `quota.update_threshold`, y `quota.validity_time`. El tiempo de espera de la solicitud Diameter proviene del `request_timeout` de `:diameter_ex`. OmniTWAG solicita cuota con un `Requested-Service-Unit` vacío, por lo que `initial_request` y `subsequent_request` no moldean el CCR hoy. No dependa de las claves no leídas hasta que el código las soporte.

Comportamiento Operacional y Manejo de Fallas

Comportamiento de falla abierta

OmniTWAG trata un fallo de transporte Gy como no fatal. Si una Solicitud de Control de Crédito falla en la capa Diameter, la sesión del suscriptor continúa y OmniTWAG no bloquea el tráfico.

- **OCS inalcanzable o tiempo de espera**: El CCR falla en el transporte Diameter. OmniTWAG registra una advertencia. Para un CCR-I, OmniTWAG aún mantiene el estado Gy, por lo que puede enviar un CCR-T más tarde. Para un CCR-U, OmniTWAG mantiene el estado Gy anterior y no bloquea al UE. Para un CCR-T, OmniTWAG registra la advertencia y completa el desmantelamiento.
- **Código de resultado de error CCA**: El OCS devuelve un `Result-Code` a nivel de comando que no es `2001` (DIAMETER_SUCCESS). OmniTWAG registra el error y trata la solicitud como fallida. OmniTWAG mantiene la sesión activa.

La aplicación es independiente del fallo de transporte. En modo `:online_charging`, OmniTWAG bloquea la ruta de datos del UE cuando el OCS deniega explícitamente el crédito o señala una `Final-Unit-Action` en un CCA MSCC. Consulte Aplicación de Cuota. Nunca bloquea en un fallo de transporte, y nunca deniega el acceso en modo `:authorize_only`. Planifique la integración del OCS con esta separación en mente.

Resumen del ciclo de vida de la sesión de control de crédito

Parse error on line 2: ...arging: modo global :disabled O AP charg... ----------^ Expecting 'SPACE', 'NL', 'HIDE_EMPTY', 'scale', 'COMPOSIT_STATE', 'STRUCT_STOP', 'STATE_DESCR', 'ID', 'FORK', 'JOIN', 'CHOICE', 'CONCURRENT', 'note', 'acc_title', 'acc_descr', 'acc_descr_multiline_value', 'CLICK', 'classDef', 'style', 'class', 'direction_tb', 'direction_bt', 'direction_rl', 'direction_lr', 'EDGE_STATE', got 'DESCR'

Intente de nuevo

Manejo de códigos de resultado

OmniTWAG lee el código de resultado en dos niveles.

- **Código de resultado a nivel de comando.** OmniTWAG trata `2001` como éxito. También trata un `Result-Code` ausente como éxito, porque algunas respuestas de pares llegan como un registro en bruto sin un código de resultado decodificado. Cualquier otro valor a nivel de comando es un error: OmniTWAG lo registra, no lo reintenta, y mantiene la sesión activa.
- **Código de resultado a nivel de MSCC.** Dentro del primer grupo MSCC, un `Result-Code` de `4010` (END_USER_SERVICE_DENIED) o `4012` (CREDIT_LIMIT_REACHED) se lee como una señal de denegación de crédito. En modo `:online_charging`, esto impulsa la aplicación y bloquea la ruta de datos del UE. Consulte Aplicación de Cuota.

Observabilidad

OmniTWAG emite eventos de Telemetría para cada Solicitud y respuesta de Control de Crédito. Los eventos llevan un valor de metadatos `type` de `:initial`, `:update`, o `:terminate`.

Evento de Telemetría	Cuándo	Metadatos
`[:twag, :gy, :ccr, :sent]`	OmniTWAG envía un CCR	`type`
`[:twag, :gy, :cca, :received]`	OmniTWAG recibe un CCA	`type`, `result` (`:success` o `:error`)
`[:twag, :gy, :ccr, :error]`	Un CCR falla en el transporte	`type`
`[:twag, :gy, :quota, :blocked]`	La ruta de datos del UE fue bloqueada por agotamiento de crédito	`identity`
`[:twag, :gy, :quota, :unblocked]`	La ruta de datos del UE fue restaurada después de una re-concesión	`identity`

> **Nota:** Estos eventos de Telemetría Gy no están registrados como m►►tricas de Prometheus en la construcción actual. La Referencia de Métricas aún no las exporta. Para observar el comportamiento de Gy hoy, utilice los registros de la aplicación. El cliente Gy registra cada CCR-I, CCR-U, y CCR-T a nivel INFO, y cada fallo a nivel WARN o ERROR. Las líneas de registro llevan el `Session-Id` de Diameter y, para el CCR-I, el IMSI.

Tablas de Referencia

Valores de CC-Request-Type

Valor	Nombre	Mensaje
1	INITIAL_REQUEST	CCR-Initial
2	UPDATE_REQUEST	CCR-Update
3	TERMINATION_REQUEST	CCR-Termination

Los valores están definidos en RFC 4006 §8.3.

Valores de Reporting-Reason utilizados

Valor	Nombre	Donde OmniTWAG lo utiliza
2	FINAL	CCR-Termination
4	VALIDITY_TIME	CCR-Update

Código de Resultado

Código	Nivel	Nombre	Comportamiento de OmniTWAG
2001	Comando	DIAMETER_SUCCESS	Tratado como éxito. El estado Gy avanza.
(ausente)	Comando	-	Tratado como éxito, para respuestas de registro en bruto.
Cualquier otro	Comando	(error)	Registrado como un error. La sesión continúa.
4010	MSCC	END_USER_SERVICE_DENIED	Leído como crédito denegado. Aplica (bloquea al UE) en modo `:online_charging`.
4012	MSCC	CREDIT_LIMIT_REACHED	Leído como crédito denegado. Aplica (bloquea al UE) en modo `:online_charging`.

Valores de Final-Unit-Action

El MSCC `Final-Unit-Indication` lleva una `Final-Unit-Action`. OmniTWAG no tiene función de plano de usuario, por lo que no puede honrar REDIRECT o RESTRICT_ACCESS en la ruta de datos. Trata cualquier `Final-Unit-Action` como una detención y bloquea al UE.

Valor	Nombre	Comportamiento de OmniTWAG
0	TERMINATE	Bloquea la ruta de datos del UE (sesión retenida).
1	REDIRECT	Bloquea la ruta de datos del UE (redirección no soportada).
2	RESTRICT_ACCESS	Bloquea la ruta de datos del UE (restricción no soportada).

Final-Unit-Indication está definido en 3GPP TS 32.299 y RFC 4006 §5.6.

Aplicación Diameter

ID	Interfaz	Diccionario	Dirección	Referencia
4	Ro / Gy	`diameter_gen_3gpp_ro`	TWAG a OCS	RFC 4006, 3GPP TS 32.299

Referencias de especificación

Tema	Especificación
Aplicación de Control de Crédito Diameter	RFC 4006
Aplicaciones de carga Diameter (Ro)	3GPP TS 32.299
Arquitectura de carga	3GPP TS 32.240
TWAG / acceso no 3GPP (S2a, NSWO)	3GPP TS 23.402
Protocolo Base Diameter	RFC 6733

Páginas relacionadas

- Resumen
- Credenciales RADIUS
- Grupos IP
- Gestión de Direcciones IP
- Sesiones
- S2a GTP-C
- Plano de Usuario

Resumen

Contabilidad por AP

Resumen

OmniTWAG agrega datos de contabilidad RADIUS para cada punto de acceso. El operador ve el tráfico total y el conteo de sesiones para cada punto de acceso. Estos datos son separados de los registros de contabilidad por sesión. OmniTWAG clavea los totales por AP por la `NAS-IP-Address` del punto de acceso. Cuando un punto de acceso omite `NAS-IP-Address` (por ejemplo, algunos modelos de UBNT), OmniTWAG recurre a la IP de origen UDP del paquete RADIUS, por lo que la contabilidad se adjunta a la misma entrada de punto de acceso que la autenticación creó.

Tabla de Contenidos

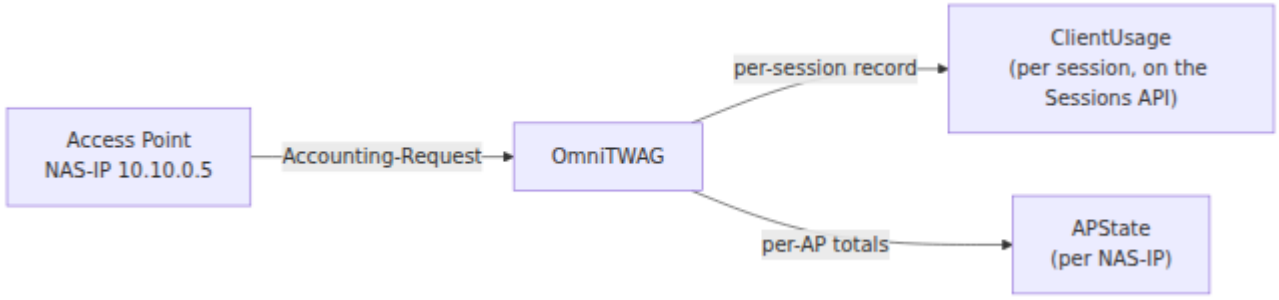
- Conceptos
- Cómo se cuentan los totales
- Campos registrados
- Conteo de clientes activos
- Dónde ver los datos
- Relación con los datos por sesión
- Solución de problemas
- Páginas relacionadas

Conceptos

Un punto de acceso envía paquetes RADIUS Accounting-Request para cada sesión de cliente. Un paquete tiene un `Acct-Status-Type` que indica el evento:

- **Inicio** (`1`): una sesión comenzó.
- **Actualización-Interina** (`3`): una sesión está activa, con los contadores hasta ahora.
- **Detener** (`2`): una sesión se detuvo, con los contadores finales.

OmniTWAG lee los contadores de octetos y paquetes y la `NAS-IP-Address` de cada paquete. OmniTWAG suma estos valores a los totales acumulados para ese punto de acceso. Esto le da al operador una vista por AP del tráfico y las sesiones.



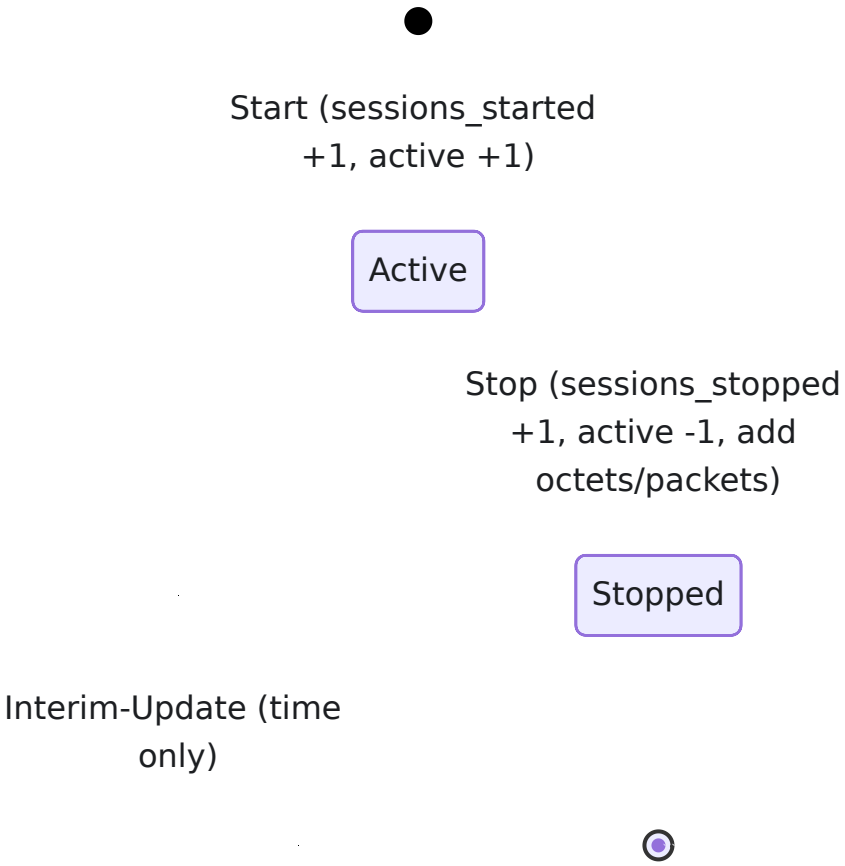
Cómo se cuentan los totales

Los contadores de octetos y paquetes RADIUS son **cumulativos para cada sesión**. Una Actualización-Interina y un Detener reportan el total de la sesión hasta ahora, no el cambio desde el último paquete. Si OmniTWAG sumara los contadores de cada paquete, contaría el mismo tráfico más de una vez.

Para evitar el conteo doble, OmniTWAG suma los contadores de octetos y paquetes a los totales por AP **solo en un Detener**. Un Detener reporta el total final para una sesión completada. Las reglas son:

- En **Inicio**: aumentar `acct_sessions_started` en uno. Aumentar `acct_active_sessions` en uno.
- En **Detener**: aumentar `acct_sessions_stopped` en uno. Disminuir `acct_active_sessions` en uno, pero no por debajo de cero. Agregar los contadores finales de octetos y paquetes a los totales por AP.
- En **Actualización-Interina**: actualizar solo el tiempo de contabilidad. No cambiar los totales.

Si OmniTWAG recibe un Detener para una sesión que no vio comenzar, el conteo de sesiones activas no baja de cero. OmniTWAG aún agrega los octetos y paquetes finales. Si OmniTWAG recibe un Accounting-Request para un punto de acceso que no vio antes, OmniTWAG crea la entrada por AP. Esto significa que un punto de acceso solo de contabilidad aún se rastrea.



Campos registrados

OmniTWAG registra estos campos para cada punto de acceso, además de los metadatos del punto de acceso como SSID y `Called-Station-Id`.

Campo	Tipo	Descripción
`acct_input_octets`	Entero	Total de octetos de entrada, sumados sobre sesiones detenidas.
`acct_output_octets`	Entero	Total de octetos de salida, sumados sobre sesiones detenidas.
`acct_input_packets`	Entero	Total de paquetes de entrada, sumados sobre sesiones detenidas.
`acct_output_packets`	Entero	Total de paquetes de salida, sumados sobre sesiones detenidas.
`acct_sessions_started`	Entero	Número de sesiones que comenzaron en este punto de acceso.
`acct_sessions_stopped`	Entero	Número de sesiones que se detuvieron en este punto de acceso.
`acct_active_sessions`	Entero	Medida de sesiones abiertas derivada de la contabilidad: `started` menos `stopped`, nunca menos que cero. Refleja solo lo que reporta la contabilidad RADIUS, por lo que puede contar en exceso cuando un punto de acceso pierde paquetes de Detener (por ejemplo, después de un reinicio) y permanece obsoleto hasta que lleguen los Detener o la entrada expire. Esta **no** es la cifra de "Clientes Activos" mostrada en la página de Puntos de Acceso — ver Conteo de clientes activos.
`last_accounting_at`	Entero	Tiempo del último paquete de contabilidad, en milisegundos desde la

Campo	Tipo	Descripción
		época Unix.

La dirección de los contadores de octetos y paquetes sigue la definición de RADIUS en RFC 2866. La entrada es tráfico del cliente hacia la red. La salida es tráfico de la red hacia el cliente.

Conteo de clientes activos

La cifra de **Clientes Activos** en la página de Puntos de Acceso, y el campo `client_count` en la API del punto de acceso, reportan cuántos clientes están **conectados en este momento**. Este valor no es el contador de contabilidad `acct_active_sessions`. Se deriva de los procesos de sesión en vivo en memoria que OmniTWAG ejecuta para cada cliente autenticado, agrupados por la `NAS-IP-Address` del punto de acceso.

Cada proceso de sesión monitorea la actividad a partir de las actualizaciones interinas de contabilidad. Cuando un cliente deja de enviar actualizaciones durante un número configurado de intervalos, el proceso se desmantela, ya sea que llegue o no un RADIUS Accounting-Stop. Debido a que el conteo de clientes activos es un recuento de estos procesos en vivo, se reduce a cero por sí mismo cuando un punto de acceso se desconecta — no depende de que el punto de acceso envíe un Detener, y no aumenta después de un reinicio del punto de acceso.

Utiliza `client_count` (la cifra en vivo) como el conteo de clientes actualmente conectados. Utiliza `acct_active_sessions` cuando desees específicamente la medida de sesiones abiertas reportada por la contabilidad, teniendo en cuenta sus advertencias anteriores.

Dónde ver los datos

Los campos por AP son parte del registro del punto de acceso. El operador los lee en dos lugares:

- **API REST**: el endpoint `GET /access_points` devuelve cada punto de acceso con sus campos de contabilidad por AP. El endpoint `GET /access_points/{id}` devuelve un punto de acceso por su `NAS-IP-Address`.
- **Interfaz web**: la página de **Puntos de Acceso** muestra cada punto de acceso. Expande un punto de acceso para ver sus datos completos, incluidos los campos de contabilidad por AP.

Relación con los datos por sesión

Los totales por AP son un agregado. Para cada RADIUS Accounting-Request, OmniTWAG escribe dos registros:

- Los **totales por AP** descritos aquí, mantenidos en `APState` y claveados por NAS-IP.
- Un **registro por sesión**, mantenido en `ClientUsage` y claveado por `Acct-Session-Id`. Esta es la vista de tráfico y contabilidad en vivo por suscriptor, y se expone en la API de Sesiones, no en el registro del punto de acceso. Ver Sesiones.

Los totales por AP cuentan octetos y paquetes solo en un Detener, por lo que se retrasan respecto a la vista en vivo por sesión durante una sesión activa. Utiliza la API de Sesiones cuando necesites el uso actual de un solo suscriptor, y los totales por AP cuando necesites el tráfico de por vida y los conteos de sesiones para un punto de acceso.

La entrada del punto de acceso se crea ya sea por autenticación (el AP se registra desde el Access-Request, ver RADIUS Shared Secrets) o, para puntos de acceso solo de contabilidad, por el primer Accounting-Request. Un punto de acceso nunca aparece si sus paquetes son descartados porque la IP de origen no coincide con ninguna subred de credenciales.

Solución de problemas

Los totales no cambian

Síntomas: Los totales de octetos y paquetes para un punto de acceso permanecen en cero, incluso aunque los clientes usen la red.

Causas posibles:

- El punto de acceso no envía paquetes de Detener. OmniTWAG agrega octetos y paquetes solo en un Detener.
- El punto de acceso no envía contabilidad en absoluto.

Resolución:

1. Confirma que el punto de acceso envía paquetes Accounting-Request a OmniTWAG.
2. Confirma que el punto de acceso envía un Detener al final de una sesión.
3. Confirma que el secreto compartido de contabilidad en el punto de acceso coincide con el credencial en archivo. Ver RADIUS Shared Secrets.

El conteo de sesiones activas parece demasiado alto

Síntomas: El valor de `acct_active_sessions` es mayor que el número real de clientes activos.

Causas posibles:

- El punto de acceso envió paquetes de Inicio pero no envió los paquetes de Detener correspondientes, por ejemplo, después de un reinicio del punto de acceso.

Resolución:

1. Confirma que el punto de acceso envía un Detener para cada sesión.
2. Establece un intervalo de actualización interina en el punto de acceso, para que OmniTWAG vea las sesiones activas.

3. El conteo activo se corrige a sí mismo a medida que llegan nuevos paquetes de Detener. El conteo no baja de cero.

Esto afecta solo a `acct_active_sessions`. La cifra de **Clientes Activos** en la página de Puntos de Acceso (y el campo de API `client_count`) no se ve afectada, porque es un recuento de procesos de sesión en vivo en lugar del contador de contabilidad. Ver Conteo de clientes activos.

Un punto de acceso no aparece

Síntomas: Un punto de acceso que envía contabilidad no aparece en la lista.

Causas posibles:

- OmniTWAG descartó los paquetes porque la IP de origen no coincidía con ninguna subred de credenciales.

Resolución:

1. Verifica la métrica `radius_auth_unauthorized_source_count`. Ver Métricas de referencia y Rechazos de RADIUS.
2. Agrega una credencial para la subred del punto de acceso. Ver RADIUS Shared Secrets.

Páginas relacionadas

- Resumen
- Sesiones
- RADIUS Shared Secrets
- RADIUS Rejections
- Métricas de referencia
- Arquitectura

Resumen

Referencias de rendimiento de OmniTWAG

A través de la autenticación y señalización del plano de control del Trusted WLAN Access Gateway.

Entorno de prueba

CPU	Intel(R) Core(TM) i5-5250U CPU @ 1.60GHz (2 núcleos)
Runtime	Elixir 1.17.0 / Erlang OTP 27 (JIT) (erts 15.0.1)
Medición	1 s de calentamiento + 3 s de medida + 1 s de memoria por operación
Commit	`main` @ `01ec4e8`

El rendimiento se informa como **ips** (iteraciones por segundo: más alto es más rápido). **Avg** y **Mediana** son la latencia por llamada en microsegundos; la mediana es la mejor estimación de la latencia típica en estado estable. **Mem/op** es la memoria en el heap asignada por llamada.

Autenticación completa (de extremo a extremo)

La cifra principal: cada escenario encadena la secuencia completa por adjunto (decodificación de identidad EAP, manejo del vector de autenticación, la jerarquía de claves EAP-AKA', construcción de desafío, MAC, decodificación de respuesta y verificación de MAC), por lo que su rendimiento (**ips**) es **adjuntos**

completos, es decir, sesiones, por segundo por núcleo. *Modo local* calcula el vector de autenticación en la caja (Milenage); *modo STa* lo recibe del HSS a través de SWx/STa.

Escenario	Rendimiento (ips)	Avg (µs)	Mediana (µs)	Mem/op
Adjuntar EAP-AKA' completo, modo STa (vector del HSS)	17,311	57.77	52.90	7,048 B
Adjuntar EAP-AKA' completo, modo local (vector calculado en la caja)	12,493	80.05	73.31	10,560 B

Criptografía

Operaciones criptográficas por autenticación: generación de vector de autenticación SIM (Milenage), la jerarquía de claves EAP-AKA' (CK'/IK', claves de sesión, MAC), claves de re-autenticación rápida ERP y la función de derivación de clave genérica 3GPP.

Operación	Rendimiento (ips)	Avg (µs)	Mediana (µs)	Mem/op
Derivación de clave WLCP (EMSK -> clave WLCP)	262,975	3.80	3.54	72 B
Construir paquete Challenge EAP-AKA' (codificar)	257,991	3.88	0.87	456 B
Derivar CK'/IK' EAP-AKA'	225,191	4.44	3.93	328 B
Calcular MAC EAP-AKA' (HMAC-SHA-256/16)	223,503	4.47	4.11	72 B
KDF genérico 3GPP (HMAC-SHA-256)	198,596	5.04	4.32	248 B
Derivar rMSK ERP (por re-autenticación)	113,269	8.83	8.01	376 B
Re-autenticación rápida completa ERP (rRK -> rIK -> rMSK)	61,496	16.26	14.97	520 B
Calcular todo Milenage (RES, CK, IK, AK, MAC-A)	52,451	19.07	16.11	3,560 B
Derivar_claves EAP-AKA' (MK -> todas las claves de sesión)	31,696	31.55	28.93	1,552 B
Criptografía de autenticación completa	28,380	35.24	32.59	1,808 B

Operación	Rendimiento (ips)	Avg (µs)	Mediana (µs)	Mem/op
EAP-AKA' (CK'/IK' + todas las claves)				

Operación	Qué hace
Construir paquete Challenge EAP-AKA'	Codifica el EAP-Request/AKA'-Challenge (AT_RAND, AT_AUTN, AT_KDF, AT_MAC).
Derivación de clave WLCP	Deriva la clave de protección WLCP del EMSK (modo de conexión múltiple).
KDF genérico 3GPP	Función de derivación de clave HMAC-SHA-256 utilizada en toda la jerarquía de claves.
Calcular MAC EAP-AKA'	HMAC-SHA-256 AT_MAC sobre un paquete EAP, truncado a 16 bytes.
Derivar CK'/IK' EAP-AKA'	Transforma CK/IK en CK'/IK' ligado al nombre de la red de acceso.
Derivar rMSK ERP	Clave de sesión por re-autenticación para EAP Re-autenticación (RFC 6696).
Re-autenticación rápida completa ERP	Cadena completa de derivación rRK → rIK ►►► rMSK para una re-autenticación rápida.
Calcular todo Milenage	Un vector de autenticación AKA: RES, CK, IK, AK, MAC-A.
Derivar_claves EAP-AKA'	Expande la clave maestra en K_encr, K_aut, K_re, MSK, EMSK.
Criptografía de autenticación completa EAP-AKA'	Derivación CK'/IK' más la expansión completa de claves de sesión.

Códecs de paquetes

Operaciones de codificación/decodificación del plano de señalización: análisis de paquetes EAP, manejo de mensajes WLCP (TS 24.244) para modo de conexión múltiple, y el atributo de clave MPPE construido en cada RADIUS Access-Accept.

Operación	Rendimiento (ips)	Avg (µs)	Mediana (µs)	Mem/op
Decodificar WLCP (Solicitud de Conexión PDN)	4,229,558	0.24	0.20	448 B
Codificar WLCP (Solicitud de Conexión PDN)	2,492,441	0.40	0.24	160 B
Codificar + decodificar WLCP ida y vuelta	1,989,407	0.50	0.40	608 B
Decodificar paquete de identidad EAP	336,816	2.97	2.66	1,136 B
Construir VSA MPPE (Send-Key, RC4 + MD5)	135,026	7.41	5.32	1,296 B

Operación	Qué hace
Decodificar WLCP	Analiza una Solicitud de Conexión PDN WLCP desde el cable.
Codificar WLCP	Serializa una Solicitud de Conexión PDN WLCP al cable.
Codificar + decodificar WLCP ida y vuelta	Codifica y luego decodifica un mensaje WLCP.
Decodificar paquete de identidad EAP	Analiza una respuesta de identidad EAP (extracción de NAI).
Construir VSA MPPE	Construye el MS-MPPE-Send-Key RADIUS VSA (RC4 + MD5) que lleva el MSK.

Secretos Compartidos de RADIUS (por grupo de IP de origen)

Resumen

OmniTWAG autentica cada cliente RADIUS (un punto de acceso o NAS) con un secreto compartido. Esta función permite a un operador gestionar **muchos** secretos compartidos, cada uno vinculado a una o más subredes de IP de origen. Un grupo de puntos de acceso que comparten una subred también comparte un secreto. El operador agrega, cambia y elimina estas credenciales bajo demanda a través de la API REST o la interfaz web del panel de control. El almacenamiento es persistente, por lo que las credenciales sobreviven a un reinicio.

Tabla de Contenidos

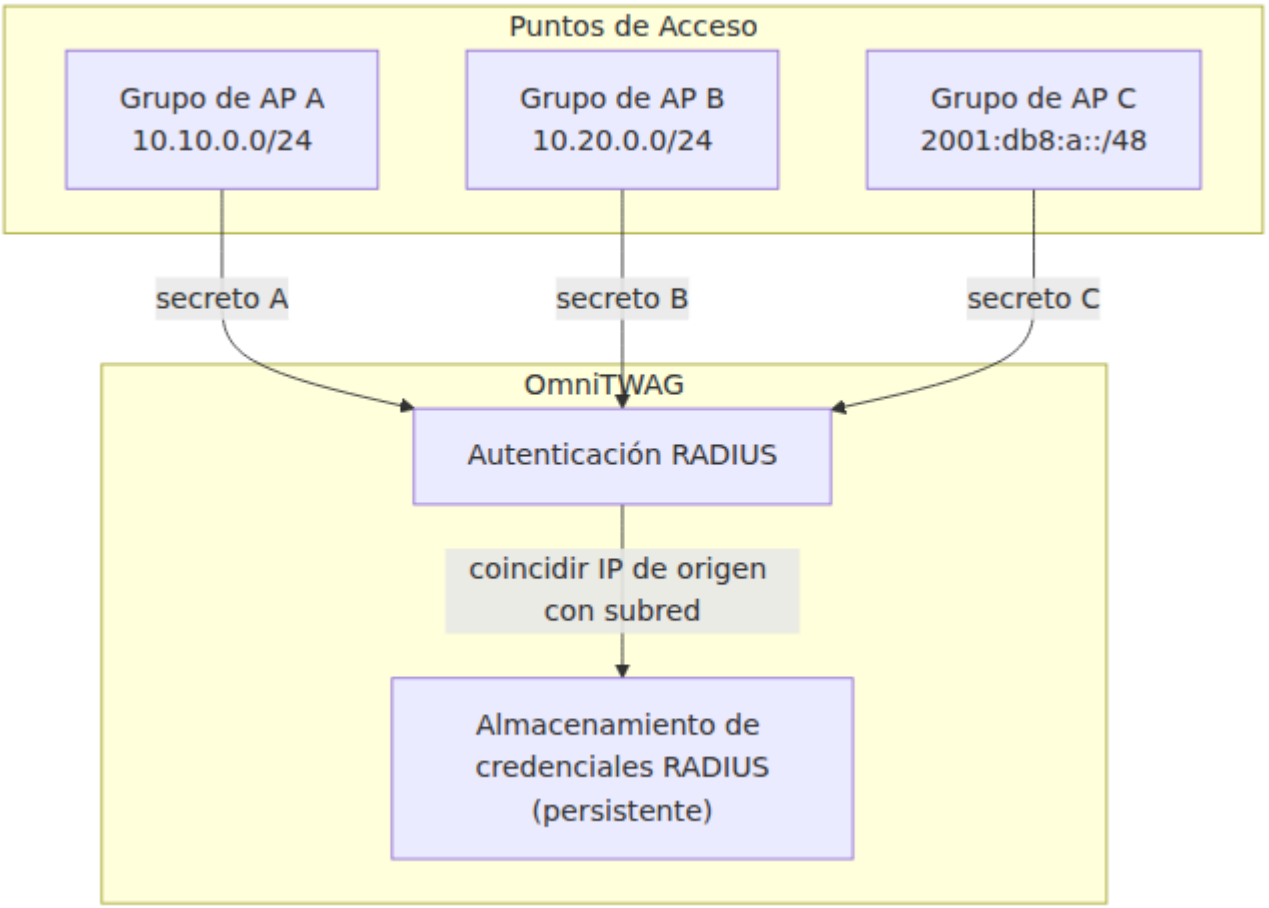
- Conceptos
- Cómo la autenticación utiliza una credencial
- Modelo de credencial
- Desagregación y carga por AP
- API REST
- Interfaz web
- Configuración
- Migración desde un único secreto estático
- Métricas
- Solución de problemas

Conceptos

Una **credencial** vincula un secreto compartido a una o más **subredes de origen**. Una subred de origen es un bloque CIDR. La subred puede ser IPv4, IPv6 o una mezcla de ambas.

La **IP de origen** es la dirección IP del punto de acceso que envía el paquete RADIUS. OmniTWAG lee la IP de origen del paquete UDP recibido. OmniTWAG no utiliza el atributo `NAS-IP-Address` para la selección de secretos, porque un atacante puede establecer ese atributo a cualquier valor.

Una subred de credencial también **autoriza** el punto de acceso. Si la IP de origen coincide con una subred de credencial, el punto de acceso está autorizado. Si la IP de origen no coincide con ninguna subred de credencial, OmniTWAG descarta el paquete. Esto significa que cuando un operador agrega una credencial para un nuevo punto de acceso, el punto de acceso está autorizado al mismo tiempo.



Cómo la autenticación utiliza una credencial

OmniTWAG resuelve el secreto compartido para cada paquete RADIUS entrante. Los pasos son:

1. OmniTWAG lee la IP de origen del paquete.
2. OmniTWAG encuentra la credencial cuya subred coincide con la IP de origen. Si más de una subred coincide, OmniTWAG utiliza la subred con el **prefijo más largo** (la coincidencia más específica).
3. Si ninguna subred coincide, OmniTWAG descarta el paquete. La fuente no está autorizada.
4. Si una subred coincide, OmniTWAG utiliza el secreto de esa credencial para decodificar el paquete, verificar el `Message-Authenticator`, cifrar las claves MPPE y firmar la respuesta.

Si una subred coincide pero el `Message-Authenticator` no se verifica, el punto de acceso tiene un secreto diferente al que está registrado. OmniTWAG registra este evento y aumenta una métrica. OmniTWAG aún procesa el paquete. La respuesta se firma con el secreto registrado, por lo que el punto de acceso rechaza la respuesta y la autenticación falla. Este comportamiento facilita encontrar un secreto mal configurado. No descarta el paquete, por lo que un caso límite válido nunca se descarta por error.

Modelo de credencial

Cada credencial tiene estos campos.

Campo	Tipo	Descripción
`id`	String	Identificador único (UUID). El servidor establece este valor.
`name`	String	Etiqueta legible por humanos, por ejemplo `Puntos de Acceso CBD de Melbourne`.
`secret`	String	Secreto compartido RADIUS.
`subnets`	Lista de String	Una o más subredes CIDR (IPv4, IPv6 o una mezcla), por ejemplo `["10.10.0.0/24", "2001:db8:a::/48"]`.
`breakout_type`	String	Manejo del plano de datos para sesiones de este punto de acceso: `tunneled`, `nswo` o `local`. Por defecto `tunneled`. Ver Desagregación y carga por AP.
`address_source`	String	Para desagregación `nswo`, cómo se asigna la IP del UE: `pool` o `dhcp`. Por defecto `pool`. Ver Desagregación y carga por AP.
`charging_mode`	String	Comportamiento de carga en línea Gy: `charged`, `uncharged` o `auth_only`. Por defecto `charged`. Ver Desagregación y carga por AP.
`ip_pool_id`	String	`id` opcional de un pool de IP que aborda sesiones `nswo` de este punto de acceso. `null` significa sin pool.
`inserted_at`	Integer	Tiempo en que se creó la credencial, en milisegundos desde la época Unix. El servidor establece este valor.

Campo	Tipo	Descripción
`updated_at`	Integer	Tiempo en que se cambió por última vez la credencial, en milisegundos desde la época Unix. El servidor establece este valor.

El operador establece `name`, `secret`, `subnets`, `breakout_type`, `address_source`, `charging_mode` y `ip_pool_id`. El servidor establece `id`, `inserted_at` y `updated_at`.

Desagregación y carga por AP

Cada credencial también controla el plano de datos y la carga para los puntos de acceso en el grupo. Los valores predeterminados preservan el comportamiento estándar: un túnel GTP hacia el PGW con carga en línea.

breakout_type

`breakout_type` establece cómo el tráfico de suscriptor sale del TWAG.

Valor	Descripción
`tunneled`	OmniTWAG establece una sesión GTP-C/GTP-U hacia el PGW y reenvía el tráfico del suscriptor a través del túnel. Este es el valor predeterminado. Ver Interfaz GTP-C S2a.
`nswo`	Descarga de WLAN no continua. OmniTWAG ancla la sesión localmente y no hace túnel hacia el PGW. OmniTWAG asigna la IP del UE desde un pool de IP. El tráfico del suscriptor transita por el TWAG y se desagrega hacia la red local.
`local`	El punto de acceso posee el plano de datos. OmniTWAG autentica al suscriptor y no asigna IP. El UE obtiene su dirección de la red local.

address_source

`address_source` se aplica solo cuando `breakout_type` es `nswo`. Establece cómo se asigna la IP del UE. Ver Pools de IP.

Valor	Descripción
`pool`	OmniTWAG pre-asigna una dirección fija del pool de IP asociado para cada sesión. Este es el valor predeterminado.
`dhcp`	El servidor DHCP de OmniTWAG asigna una dirección del rango del pool cuando el UE envía un DHCP DISCOVER.

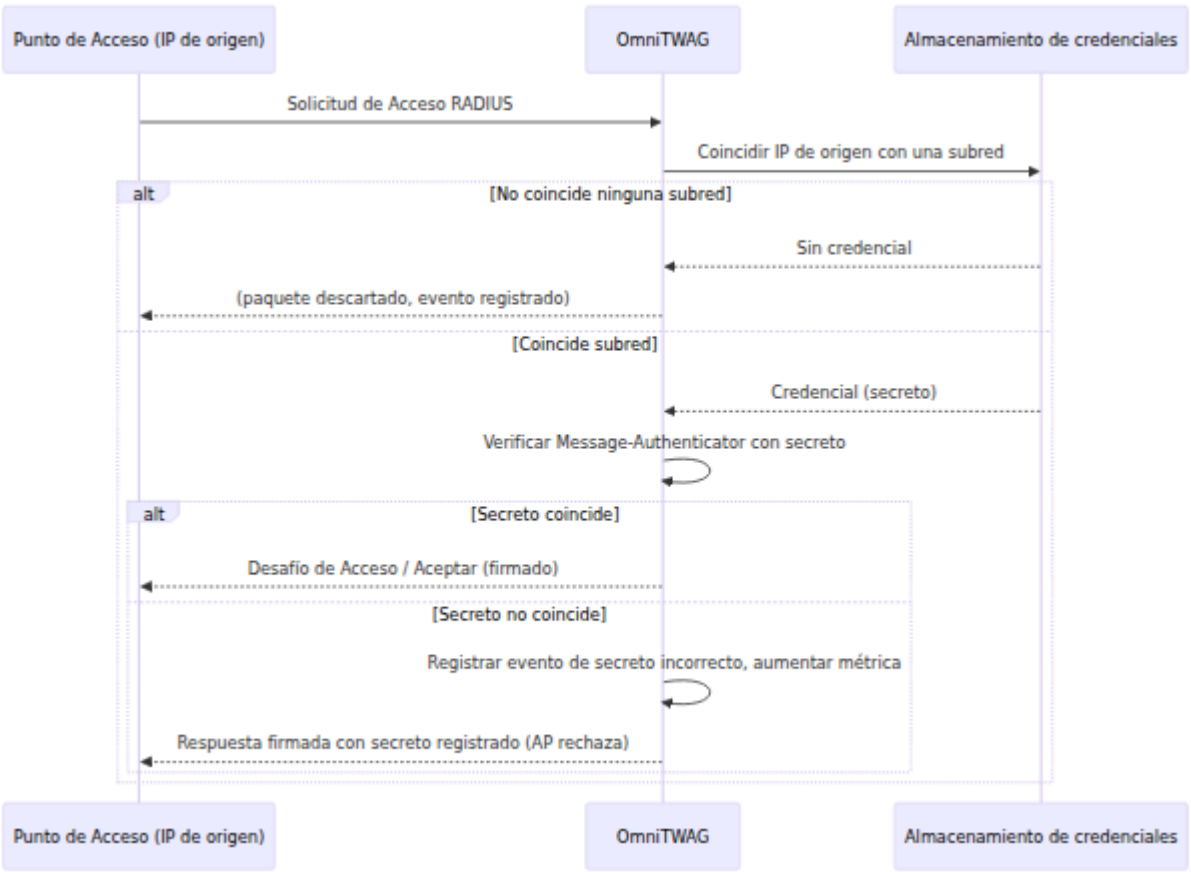
charging_mode

`charging_mode` establece el comportamiento de carga en línea Gy. Ver Carga en Línea.

Valor	Descripción
`charged`	OmniTWAG abre una sesión de carga en línea Gy y reporta el uso al OCS. Este es el valor predeterminado.
`uncharged`	OmniTWAG establece la sesión de datos pero no abre una sesión Gy. Usa este valor para acceso de tarifa plana o no cargado.
`auth_only`	OmniTWAG autentica al suscriptor pero no establece una sesión de datos. Usa este valor cuando otro sistema proporciona conectividad.

Cómo interactúan los campos

`charging_mode: auth_only` tiene prioridad. Cuando una credencial utiliza `auth_only`, OmniTWAG no establece una sesión de datos, por lo que `breakout_type` y `address_source` no tienen efecto.



API REST

La API REST de OmniTWAG sirve estos endpoints a través de HTTPS en el puerto `api_ex` configurado (`8444` por defecto). La ruta base es `/radius_credentials`.

Método	Ruta	Acción	Descripción
GET	`/radius_credentials`	index	Listar todas las credenciales.
GET	`/radius_credentials/{id}`	show	Obtener una credencial por `id`.
POST	`/radius_credentials`	create	Crear una credencial.
PUT	`/radius_credentials/{id}`	update	Cambiar una credencial.
DELETE	`/radius_credentials/{id}`	delete	Eliminar una credencial.

Cuerpo de la solicitud

`POST` y `PUT` aceptan un cuerpo JSON con estos campos.

Campo	Tipo	Requerido	Descripción
`name`	String	No	Etiqueta legible por humanos.
`secret`	String	Sí	Secreto compartido RADIUS. No debe estar vacío.
`subnets`	Lista de String	Sí	Una o más subredes CIDR válidas. No debe estar vacío.
`breakout_type`	String	No	`tunneled`, `nswo` o `local`. Por defecto `tunneled`. Ver Desagregación y carga por AP.
`address_source`	String	No	`pool` o `dhcp` (para `nswo`). Por defecto `pool`.
`charging_mode`	String	No	`charged`, `uncharged` o `auth_only`. Por defecto `charged`.
`ip_pool_id`	String	No	`id` de un pool de IP para direccionamiento `nswo`. Una cadena vacía significa sin pool.

`PUT` solo cambia los campos en el cuerpo. Los campos que no están en el cuerpo mantienen su valor actual.

Respuestas

Estado	Significado
`200 OK`	La solicitud fue exitosa (index, show, update, delete).
`201 Created`	La credencial fue creada.
`404 Not Found`	Ninguna credencial tiene el `id` dado.
`422 Unprocessable Entity`	El `secret` está vacío, una subred no es un bloque CIDR válido, o `breakout_type` / `address_source` / `charging_mode` no es un valor válido.

Ejemplo: listar credenciales

Solicitud:

```
GET /radius_credentials
```

Respuesta:

```
{
  "count": 1,
  "radius_credentials": [
    {
      "id": "0d1e2f3a-4b5c-6d7e-8f90-1a2b3c4d5e6f",
      "name": "Puntos de Acceso CBD de Melbourne",
      "secret": "s3cr3t",
      "subnets": ["10.10.0.0/24", "2001:db8:a::/48"],
      "inserted_at": 1734512400000,
      "updated_at": 1734512400000
    }
  ]
}
```

Ejemplo: crear una credencial

Solicitud:

```
POST /radius_credentials
```

```
{
  "name": "Puntos de Acceso CBD de Melbourne",
  "secret": "s3cr3t",
  "subnets": ["10.10.0.0/24", "2001:db8:a::/48"]
}
```

Una solicitud exitosa devuelve `201 Created` con la nueva credencial, incluyendo su `id`.

Ejemplo: una subred no válida

Solicitud:

```
{
  "secret": "s3cr3t",
  "subnets": ["10.10.0.0/24", "not-a-subnet"]
}
```

Respuesta (`422 Unprocessable Entity`):

```
{
  "error": "subred CIDR no válida(s): not-a-subnet"
}
```

Interfaz web

El panel de control tiene una página de **Secretos RADIUS** en `/radius_credentials`. La página lista cada credencial con su nombre, subredes, un secreto enmascarado y la hora de la última modificación. La

página tiene un formulario para agregar una credencial y controles para editar o eliminar una credencial. Ingrese una o más subredes en el campo de subredes. Separe las subredes con una coma o una nueva línea.

Configuración

El almacenamiento de credenciales RADIUS mantiene sus datos en un archivo DETS. Establezca la ruta del archivo en el bloque `radius_config`.

```
config :omnitwag,
  radius_config: %{
    # Ruta al almacenamiento persistente de credenciales RADIUS.
    credentials_dets_path: "priv/radius_credentials.dets",
    # Campos de secreto único heredados (ver "Migración" a
continuación).
    allowed_source_subnets: [],
    secret: "123456"
  }
```

Parámetro	Tipo	Requerido	Predeterminado
`credentials_dets_path`	String	No	`priv/radius_credentials`
`secret`	String	No	`123456`
`allowed_source_subnets`	Lista de String	No	`[]`

Migración desde un único secreto estático

Versiones anteriores usaban un único `secret` estático y una lista de permitidos `allowed_source_subnets` opcional. En el primer inicio, si el almacenamiento de credenciales está vacío, OmniTWAG inicializa una credencial a partir de estos campos heredados. La inicialización se ejecuta solo cuando ambas de estas condiciones son verdaderas:

- El almacenamiento no tiene credenciales.
- El `secret` está establecido y `allowed_source_subnets` no está vacío.

La credencial inicializada utiliza el `secret` heredado y las subredes heredadas. Esto mantiene un despliegue existente funcionando después de una actualización. OmniTWAG registra un mensaje cuando inicializa una credencial.

Si `allowed_source_subnets` está vacío, OmniTWAG no inicializa una credencial. En ese caso, OmniTWAG rechaza todas las solicitudes RADIUS hasta que el operador agregue una credencial. Agregue credenciales a través de la API REST o la interfaz web.

La autenticación por paquete no lee `secret` ni `allowed_source_subnets`. Estos campos sirven solo como la inicialización del primer inicio. Administre credenciales activas a través de la API o la interfaz web.

Métricas

OmniTWAG expone estos contadores en el endpoint de Prometheus. Consulte la Referencia de Métricas para el endpoint y ejemplos de consultas.

radius_auth_unauthorized_source_count

Tipo: Contador
Descripción: Número de paquetes RADIUS descartados porque la IP de origen no coincidió con ninguna subred de credencial.

Etiqueta	Descripción
`source`	IP de origen del paquete descartado.

radius_auth_bad_secret_count

Tipo: Contador
Descripción: Número de paquetes RADIUS cuya IP de origen coincidió con una credencial, pero cuyo `Message-Authenticator` no se verificó. El punto de acceso tiene un secreto diferente al que está registrado.

Etiqueta	Descripción
`source`	IP de origen del punto de acceso.
`credential_id`	`id` de la credencial coincidente.

Solución de problemas

Un punto de acceso no recibe respuesta

Síntomas: El punto de acceso envía solicitudes RADIUS pero no recibe respuesta. La métrica `radius_auth_unauthorized_source_count` aumenta.

Causas posibles:

- Ninguna subred de credencial contiene la IP de origen del punto de acceso.
- El punto de acceso envía desde una IP diferente a la esperada, por ejemplo, una dirección NAT.

Resolución:

1. Encuentre la IP de origen del punto de acceso en los registros de OmniTWAG.
2. Confirme que una subred de credencial contenga esa IP.

3. Si ninguna credencial cubre la IP, agregue una credencial para la subred correcta.

Un punto de acceso está autorizado pero la autenticación falla

Síntomas: La métrica `radius_auth_bad_secret_count` aumenta para el punto de acceso. Los clientes no pueden autenticarse.

Causas posibles:

- El punto de acceso tiene un secreto compartido diferente al de la credencial registrada.

Resolución:

1. Lea la etiqueta `credential_id` en la métrica, o encuentre el mensaje de secreto incorrecto en los registros.
2. Compare el secreto en el punto de acceso con el secreto en la credencial.
3. Corrija el secreto en el punto de acceso o actualice la credencial para que coincida.

Las credenciales se pierden después de un reinicio

Síntomas: La lista de credenciales está vacía después de un reinicio. OmniTWAG rechaza todas las solicitudes.

Causas posibles:

- La `credentials_dets_path` apunta a un directorio que no es escribible, o a una ruta que se borra al reiniciar.

Resolución:

1. Establezca `credentials_dets_path` en un directorio de datos persistente y escribible.
2. Confirme que el proceso de OmniTWAG pueda escribir en esa ruta.

3. Agregue las credenciales nuevamente a través de la API o la interfaz web.

Rechazos de RADIUS

Resumen

OmniTWAG rastrea las fuentes de RADIUS cuyos paquetes fueron rechazados. Esto permite a un operador identificar un punto de acceso desconocido o mal configurado de un vistazo. La función es la imagen espejo de Secretos Compartidos de RADIUS: el almacén de credenciales registra las fuentes que OmniTWAG **acepta**, y el rastreador de rechazos registra las fuentes que OmniTWAG **rechaza**.

El rastreador mantiene sus datos en memoria. Los datos no son persistentes. El rastreador se reinicia al reiniciar. Esto es intencional. Ver Solución de problemas.

Tabla de Contenidos

- Conceptos
- Cómo un paquete llega al rastreador
- Razones de rechazo
- Modelo de entrada
- Comportamiento del almacén
- API REST
- Métricas
- Solución de problemas

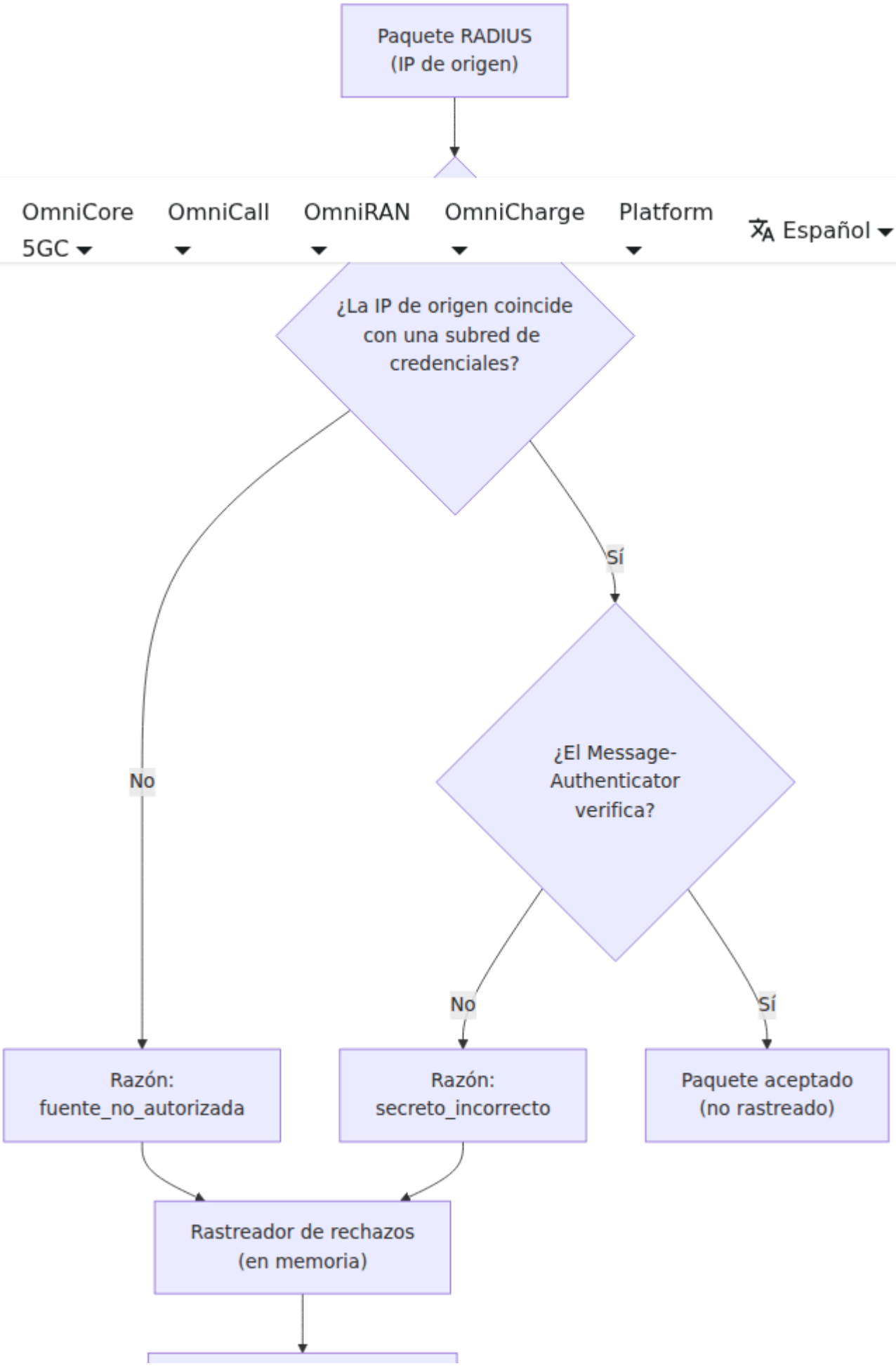
Conceptos

Una **fuente rechazada** es una dirección IP que envió un paquete RADIUS que OmniTWAG no aceptó. El rastreador registra una entrada por cada par único de **fuente** y **razón**. Si la misma fuente es rechazada muchas veces por la misma razón, OmniTWAG aumenta un contador en la entrada existente. OmniTWAG no añade una nueva fila.

La **IP de origen** es la dirección IP que envió el paquete RADIUS. OmniTWAG lee la IP de origen del paquete UDP recibido. OmniTWAG no utiliza el atributo `NAS-IP-Address`, porque un atacante puede establecer ese atributo a cualquier valor.

Cómo un paquete llega al rastreador

OmniTWAG compara cada paquete RADIUS entrante con las subredes de credenciales. Un paquete es rechazado cuando falla una de dos verificaciones. El rastreador registra la fuente y la razón para cada paquete rechazado.


Paquete RADIUS (IP de origen)
¿La IP de origen coincide con una subred de credenciales?
No
Sí
¿El Message-Authenticator verifica?
No
Sí
Razón: fuente_no_autorizada
Razón: secreto_incorrecto
Paquete aceptado (no rastreado)
Rastreador de rechazos (en memoria)

Coalescer por {fuente, razón}
aumentar cuenta

Razones de rechazo

Una fuente es rechazada por una de dos razones.

Razón	Significado
`fuente_no_autorizada`	La IP de origen no coincidió con ninguna subred de credenciales. OmniTWAG descartó el paquete. Este es el caso de punto de acceso desconocido.
`secreto_incorrecto`	Una subred de credenciales coincidió, pero el `Message-Authenticator` del paquete no verificó. El punto de acceso tiene un secreto compartido diferente al que está registrado.

Para una entrada `secreto_incorrecto`, el rastreador también registra la credencial que coincidió. Esto indica al operador qué credencial verificar.

Modelo de entrada

Cada entrada tiene estos campos.

Campo	Tipo	Descripción
`source`	Cadena	IP de origen del paquete rechazado. Este valor es el identificador de la entrada.
`reason`	Cadena	Razón del rechazo: `fuente_no_autorizada` o `secreto_incorrecto`.
`count`	Entero	Número de veces que OmniTWAG rechazó esta fuente por esta razón.
`first_seen`	Entero	Tiempo en que la fuente fue rechazada por primera vez, en milisegundos desde la época Unix.
`last_seen`	Entero	Tiempo en que la fuente fue rechazada por última vez, en milisegundos desde la época Unix.
`credential_id`	Cadena	`id` de la credencial que coincidió. Se establece solo cuando `reason` es `secreto_incorrecto`.
`credential_name`	Cadena	`name` de la credencial que coincidió. Se establece solo cuando `reason` es `secreto_incorrecto`.

Los campos `credential_id` y `credential_name` están vacíos para una entrada `fuente_no_autorizada`, porque no coincidió ninguna credencial.

Comportamiento del almacén

El rastreador es un almacén en memoria. Tiene estas propiedades.

- **No persistente.** El almacén se reinicia al reiniciar. No sobrevive a un reinicio de proceso.

- **Coalescido.** OmniTWAG registra una entrada por cada par de `source` y `reason`. Un rechazo repetido aumenta `count` y actualiza `last_seen`. No añade una fila.
- **Limitado.** El almacén mantiene un número máximo de fuentes. El máximo predeterminado es 256.
- **Desalojo por menos recientemente visto.** Cuando el almacén está lleno, OmniTWAG elimina la entrada con el valor `last_seen` más antiguo para hacer espacio para una nueva fuente. El almacén está limitado porque una IP de origen puede ser suplantada. Un flujo de IPs de origen suplantadas no puede hacer crecer el almacén sin límite.

API REST

La API REST de OmniTWAG sirve estos puntos finales en el puerto API configurado (`8444` por defecto). La ruta base es `/radius_rejections`.

Método	Ruta	Acción	Descripción
GET	`/radius_rejections`	index	Listar fuentes rechazadas. La lista está paginada. Buscar por `source` o `reason`.
DELETE	`/radius_rejections/{id}`	delete	Olvidar una fuente. `{id}` es la IP de origen.

Utilice `DELETE` para olvidar una fuente después de resolverla, por ejemplo, después de agregar una credencial que cubra la IP de origen.

Respuestas

Estado	Significado
`200 OK`	La solicitud fue exitosa.
`404 Not Found`	Ninguna entrada tiene la IP de origen dada.

Ejemplo: listar fuentes rechazadas

Solicitud:

```
GET /radius_rejections
```

Respuesta:

```
{
  "count": 2,
  "radius_rejections": [
    {
      "source": "203.0.113.7",
      "reason": "fuente_no_autorizada",
      "count": 42,
      "first_seen": 1734512400000,
      "last_seen": 1734512999000,
      "credential_id": null,
      "credential_name": null
    },
    {
      "source": "10.10.0.15",
      "reason": "secreto_incorrecto",
      "count": 3,
      "first_seen": 1734512500000,
      "last_seen": 1734512890000,
      "credential_id": "0d1e2f3a-4b5c-6d7e-8f90-1a2b3c4d5e6f",
      "credential_name": "APs del CBD de Melbourne"
    }
  ]
}
```

Ejemplo: olvidar una fuente

Solicitud:

```
DELETE /radius_rejections/203.0.113.7
```

Respuesta:

```
{
  "deleted": "203.0.113.7"
}
```

Métricas

El rastreador de rechazos complementa dos contadores de Prometheus que documenta la función de credenciales. Cada contador aumenta al mismo tiempo que OmniTWAG registra un rechazo.

Contador	Razón de rechazo
`radius_auth_fuente_no_autorizada_count`	`fuente_no_autorizada`
`radius_auth_secreto_incorrecto_count`	`secreto_incorrecto`

Los contadores proporcionan totales a lo largo del tiempo. El rastreador ofrece una vista en vivo, por fuente. Ver Secretos Compartidos de RADIUS para las definiciones de los contadores, y la Referencia de Métricas para el punto final de Prometheus y ejemplos de consulta.

Solución de problemas

Un punto de acceso desconocido aparece en la lista

Síntomas: Una fuente aparece con `reason` configurado como `fuente_no_autorizada`. El métrico `radius_auth_fuente_no_autorizada_count` aumenta para esa fuente.

Causas posibles:

- Ninguna subred de credenciales contiene la IP de origen.
- El punto de acceso está detrás de NAT. La IP de origen RADIUS que OmniTWAG ve difiere de la dirección del propio punto de acceso.

Resolución:

1. Lea el campo `source` de la entrada.
2. Confirme que este es un punto de acceso conocido.

3. Agregue una subred de credenciales que cubra la IP de origen que OmniTWAG realmente ve. Para un caso de NAT, use la dirección traducida, no la dirección del punto de acceso. Ver Secretos Compartidos de RADIUS.
4. Elimine la entrada con `DELETE /radius_rejections/{id}` para olvidar la fuente.

Un punto de acceso conocido aparece como secreto_incorrecto

Síntomas: Una fuente conocida aparece con `reason` configurado como `secreto_incorrecto`. El métrico `radius_auth_secreto_incorrecto_count` aumenta para esa fuente.

Causas posibles:

- El punto de acceso tiene un secreto compartido diferente al de la credencial registrada.

Resolución:

1. Lea los campos `credential_id` y `credential_name` de la entrada. Estos nombran la credencial que coincidió.
2. Compare el secreto en el punto de acceso con el secreto en esa credencial.
3. Corrija el secreto en el punto de acceso, o actualice la credencial para que coincida. Ver Secretos Compartidos de RADIUS.
4. Elimine la entrada con `DELETE /radius_rejections/{id}` para olvidar la fuente.

La lista está vacía después de un reinicio

Síntomas: La lista de rechazos está vacía después de un reinicio, aunque las fuentes estaban listadas antes del reinicio.

Causa:

- Este es un comportamiento esperado. El rastreador mantiene sus datos en memoria. Los datos no son persistentes. El almacén se reinicia al reiniciar.

Resolución:

- No se necesita ninguna acción. Nuevos rechazos repoblarán la lista. Utilice los métricos `radius_auth_fuente_no_autorizada_count` y `radius_auth_secreto_incorrecto_count` para un historial que sobreviva a un reinicio.

Resumen

S2a / Interfaz GTP-C al PGW

Descripción general

La interfaz S2a conecta el OmniTWAG a la Puerta de Enlace de Datos por Paquete (PGW). Transporta conectividad PDN basada en GTP para una Red de Acceso WLAN de Confianza (TWAN). OmniTWAG actúa como el punto final GTP-C de TWAN. Este rol es similar al de un ePDG en la interfaz S2b, pero utiliza tipos de interfaz S2a y un Identificador de TWAN.

La interfaz S2a tiene dos planos:

- **GTP-C (plano de control)**: crea, modifica y elimina sesiones PDN. También gestiona la ruta. Este documento cubre GTP-C.
- **GTP-U (plano de usuario)**: transporta datos de UE en túneles GTP. Consulte la guía de Plano de Usuario.

GTP-C en S2a sigue 3GPP TS 29.274 (GTPv2-C). El TWAG utiliza tipos de mensajes GTPv2-C y elementos de información (IEs) con tipos de interfaz específicos de S2a.

Tabla de Contenidos

- Dónde encaja S2a: Los Tres Modos de Salida
- Rol en el Ciclo de Vida de la Sesión
- Tipos de Interfaz S2a
- Establecimiento de Sesión
  - Disparador
  - Solicitud de Creación de Sesión
  - Portadora por Defecto
  - Respuesta a la Solicitud de Creación de Sesión
  - Sesiones de Emergencia

- Gestión de Portadoras
- Desmantelamiento de Sesión
- Gestión de Rutas
- Configuración
- Tablas de Referencia
- Observabilidad
- Solución de Problemas
- Referencias 3GPP

Dónde encaja S2a: Los Tres Modos de Salida

OmniTWAG resuelve un modo de plano de datos por punto de acceso (AP), a partir de las credenciales RADIUS que coinciden con el AP. El valor `breakout_type` en la credencial selecciona uno de los tres modos. Consulte Credenciales RADIUS para los ajustes de `breakout_type`, `address_source` y `ip_pool_id`.

Modo (`breakout_type`)	Quién asigna la IP de UE	Plano de datos	S2a / GTP utilizado
`tunneled`	PGW (en el IE de Asignación de Dirección PDN)	GRE desde el AP al TWAG, luego GTP-U en S2a al PGW, luego hacia Internet por el PGW	Sí. Este documento.
`nswo`	TWAG, desde un grupo de IP	GRE desde el AP al TWAG, luego salida local a Internet (NAT) en el TWAG	No. No hay sesión PGW.
`local`	La red local en el AP	El AP posee el plano de datos. El TWAG no transporta tráfico.	No. No hay sesión PGW, no IP.

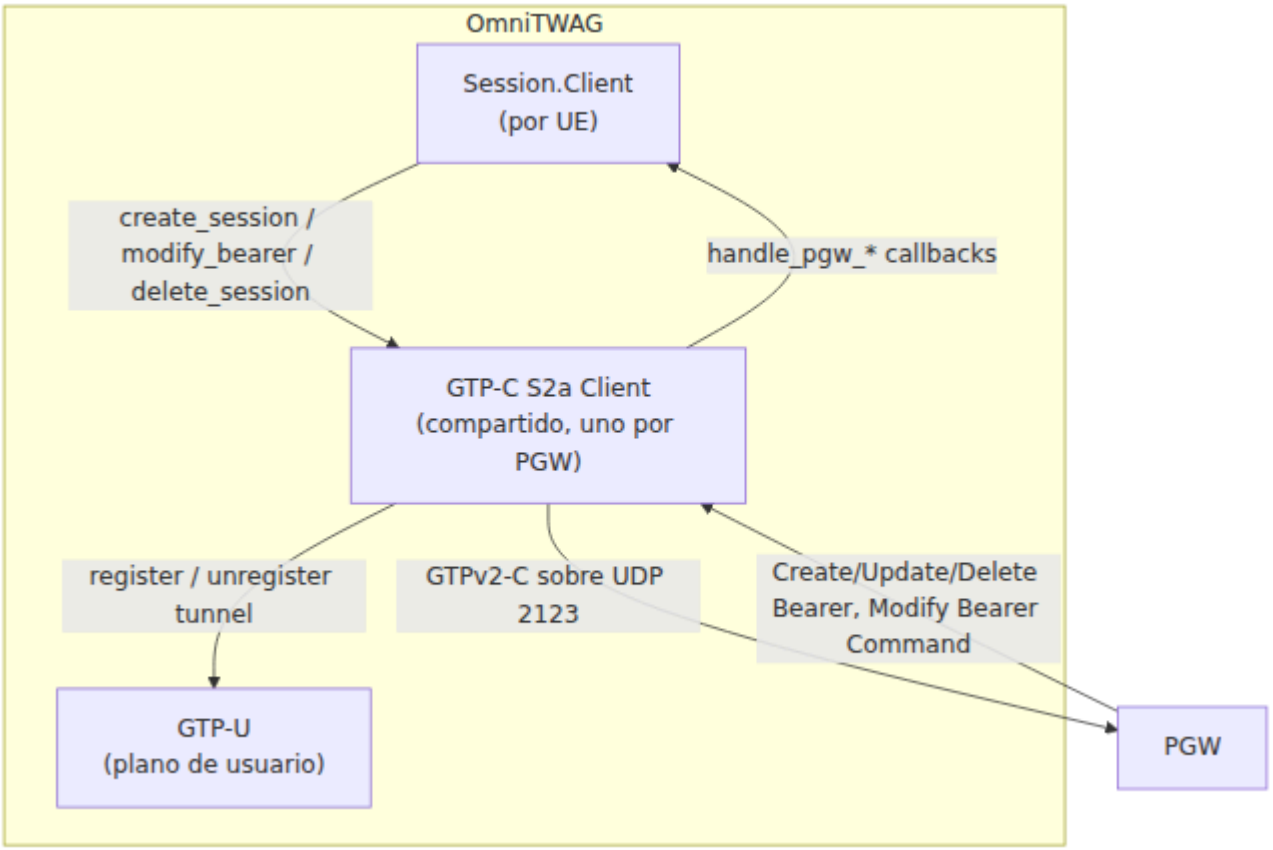
GTP-C en S2a se aplica solo al modo `tunneled`. En el modo `tunneled`, el PGW asigna la dirección IP de UE y aplica políticas y cargos. En los modos `nswo` y `local`, el TWAG no crea una sesión PDN, por lo que no se envía ningún mensaje S2a.

El cobro en modo `tunneled` lo realiza el PGW, no el TWAG. El cobro en línea del lado del TWAG en Cobro en Línea se aplica a los modos de salida local.

> Nota: Un controlador global separado de Descarga WLAN No Sincronizada (NSWO) (`Twag.NSWO.Handler`, habilitado con `nswo.enabled: true`) proporciona salida local para cada sesión cuando no se utiliza GTP. Consulte la sección de Salida Local de la guía de Plano de Usuario. El tipo de salida `nswo` por AP descrito anteriormente es el modelo impulsado por credenciales y es preferido.

Rol en el Ciclo de Vida de la Sesión

El cliente GTP-C es un único proceso compartido (`Twag.Gtp.GtpcS2a`). Maneja todas las sesiones contra un PGW configurado. Cada sesión de UE `tunneled` autenticada impulsa un intercambio de Creación de Sesión. El proceso `Session.Client` por sesión llama al cliente GTP-C. No envía mensajes GTPv2-C por sí mismo.



Tipos de Interfaz S2a

El TWAG utiliza tipos de interfaz F-TEID S2a. Estos son diferentes de S2b (ePDG), que utiliza los tipos 30 y 31. Los valores siguen TS 29.274 Tabla 8.22-1.

Tipo de Interfaz	Valor	Dirección	Uso
`s2a_twan_gtp_c`	35	Control TWAG	F-TEID TWAG en Solicitud de Creación de Sesión
`s2a_twan_gtp_u`	34	Usuario TWAG	F-TEID de portadora TWAG (GTP-U)
`s2a_pgw_gtp_c`	36	Control PGW	F-TEID PGW en Respuesta a la Solicitud de Creación de Sesión
`s2a_pgw_gtp_u`	37	Usuario PGW	F-TEID de portadora PGW (GTP-U)

El Tipo RAT es siempre WLAN (valor 3).

Establecimiento de Sesión

Disparador

Una autenticación EAP exitosa inicia la sesión. El AP envía un Accounting-Start, que activa el procesamiento posterior a la aceptación. El `Session.Client` resuelve el modo de salida del AP. Para el modo `tunneled`, llama al cliente GTP-C para crear la sesión PDN. Si el parámetro `gtp.enabled` es `false`, el TWAG salta S2a. Si la sesión no tiene IMSI, el TWAG no crea la sesión. Para los modos `nswo` y `local`, el TWAG no envía ningún mensaje S2a. Consulte Sesiones para el ciclo de vida completo de la sesión.

Solicitud de Creación de Sesión

El TWAG envía una Solicitud de Creación de Sesión (tipo de mensaje según TS 29.274 §7.2.1) al PGW. El TWAG genera un nuevo TEID de plano de control no predecible y un TEID de plano de usuario para la solicitud. Construye la solicitud con estos IEs:

Elemento de Información	Origen / Valor	Notas
IMSI	Suscripción de UE	Omitido si no está presente.
Red de Servicio (MCC/MNC)	`gtp.mcc` / `gtp.mnc`, o primeros dígitos del IMSI	Omitido si falta alguno de los valores.
Tipo RAT	WLAN (3)	Fijo para S2a.
F-TEID (control)	Tipo de interfaz 35, TEID de control TWAG, IP local TWAG	Identifica el punto final GTP-C del TWAG.
Nombre de Punto de Acceso (APN)	`gtp.apn`, por defecto `internet`. APN de emergencia si se solicita.	Consulte Sesiones de Emergencia.
Modo de Selección	2 (proporcionado por la red, suscripción no verificada)	El APN siempre es proporcionado por la red en TWAN.
Asignación de Dirección PDN (PAA)	Tipo solicitado con una dirección cero	El TWAG solicita una dirección. El PGW la asigna.
Tasa Máxima Agregada de Bits (APN-AMBR)	Por defecto 256000 uplink / 256000 downlink	Sobrescrito si se proporciona una tasa.
Contexto de Portadora a Crear	EBI por defecto 5, F-TEID de portadora (tipo 34), QoS	Consulte Portadora por Defecto.

Elemento de Información	Origen / Valor	Notas
Restricción de APN	2 (normal), o 0 para emergencia	
Identidad del ME (IMEISV)	Identidad del equipo de UE	Omitido si no está presente.
Recuperación	Contador de recuperación del TWAG	Usado para detección de reinicio.
Características de Cobro	`0x0800`	Valor fijo.
Identificador de TWAN	SSID de `twan_ssid`, BSSID opcional de `twan_bssid`	Identifica la WLAN.
Zona Horaria de UE	Derivada del desfase horario local del host	
Indicación	Bandera de transferencia	La bandera de transferencia se establece para una transferencia entrante.

El tipo de PDN por defecto es IPv4. Puede configurarse a `:ipv4`, `:ipv6` o `:ipv4v6` a través del parámetro `gtp.pdn_type`. La dirección PAA solicitada es siempre cero, por lo que el PGW asigna la dirección real.

> Nota: El IE de Opciones de Configuración de Protocolo Adicional (APCO) no está incluido. El TWAG no solicita direcciones DNS o P-CSCF a través de APCO en este momento. Lee DNS del PCO o del IE de PCO Adicional en la respuesta si el PGW proporciona uno.

Portadora por Defecto

El TWAG crea una portadora por defecto por sesión. La EBI por defecto es 5. La portadora no tiene filtros TFT, por lo que transporta todo el tráfico. La QoS a nivel de portadora en la solicitud utiliza estos valores fijos:

Parámetro QoS	Valor
QCI	9
Nivel de Prioridad	10
Capacidad de Pre-empción	deshabilitada
Vulnerabilidad a Pre-empción	habilitada
GBR / MBR (uplink y downlink)	0

El TWAG no admite portadoras dedicadas. Consulte Gestión de Portadoras.

Respuesta a la Solicitud de Creación de Sesión

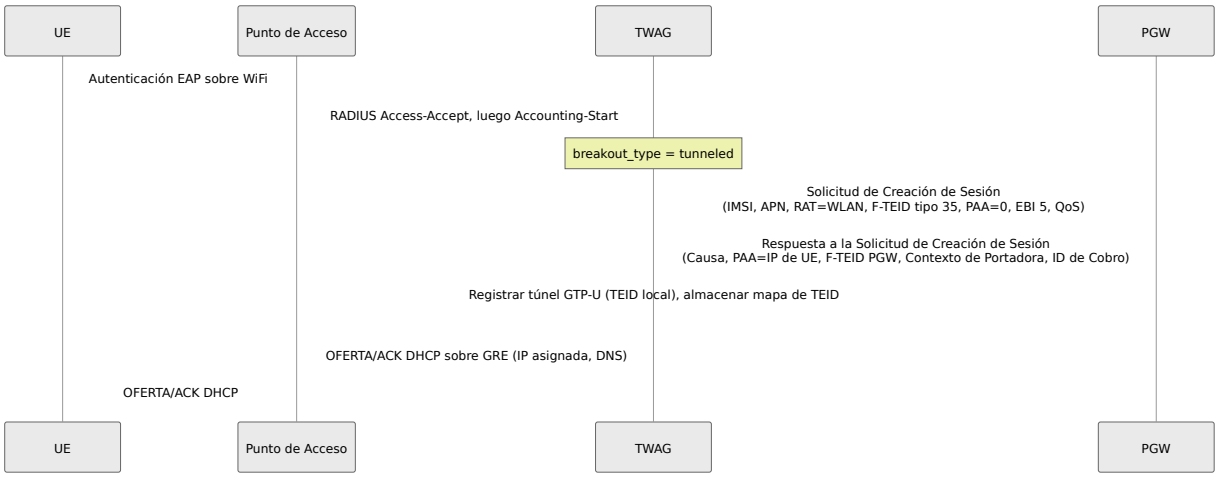
El TWAG empareja la respuesta con la solicitud pendiente por número de secuencia. Verifica el IE de Causa. La solicitud es un éxito si la causa es `request_accepted`, `new_pdp_type_due_to_network_preference`, o `new_pdp_type_due_to_single_address_bearer_only`. Cualquier otra causa devuelve un `gtpc_error` con ese valor de causa.

En caso de éxito, el TWAG lee estos valores de la respuesta:

Elemento de Respuesta	Valor Extraído	Uso
Asignación de Dirección PDN (PAA)	Dirección IP de UE asignada	Entregada a la UE por DHCP sobre GRE. Consulte Gestión de Direcciones IP.
F-TEID (control PGW)	TEID de control PGW y IP PGW	Usado como destino para mensajes de control posteriores.
Contexto de Portadora	F-TEID de portadora (TEID y IP GTP-U PGW), EBI, ID de Cobro	Usado para registrar el túnel GTP-U.
APN-AMBR	Límites de tasa de sesión	Almacenados en la sesión.
PCO / PCO Adicional	Direcciones de servidores DNS	Entregadas a la UE por DHCP.

El TWAG almacena un mapeo de su TEID de control local al TEID de control PGW y la identidad de UE. Este mapeo enruta los mensajes iniciados por PGW posteriores a la sesión correcta. Si no hay F-TEID de portadora presente, el TWAG recurre al F-TEID de PGW para el túnel de plano de usuario.

Después de un éxito, el `Session.Client` registra el túnel GTP-U (que genera el TEID de plano de usuario local del lado del TWAG), pre-asigna el arrendamiento DHCP con la IP y DNS asignadas, y registra la UE para el enrutamiento de bajada GRE. Consulte Plano de Usuario para el túnel y la ruta de bajada.



Sesiones de Emergencia

Si la sesión solicita un servicio de emergencia, el TWAG utiliza el APN de emergencia. El parámetro `emergency_apn` establece este APN. El valor por defecto es `sos`. Para una sesión de emergencia, la Restricción de APN se establece en 0 (sin restricción).

Gestión de Portadoras

Solo Portadora por Defecto

OmniTWAG admite solo la portadora por defecto. No admite portadoras dedicadas. Esto sigue el perfil de Modo de Conexión Única Transparente (TSCM) de TWAN. El gestor de portadoras puede rastrear una tabla de portadoras indexada por EBI, pero S2a establece solo la portadora por defecto.

Modificar Portadora (Iniciada por TWAG)

El TWAG puede enviar una Solicitud de Modificación de Portadora (TS 29.274 §7.2.7). Esta solicitud transporta un contexto de portadora con la EBI por defecto, el F-TEID de plano de usuario del TWAG (tipo 34), y la IP local del TWAG. El TEID de destino es el TEID de control PGW. El TWAG empareja la Respuesta de Modificación de Portadora por número de secuencia y verifica el IE de Causa para el éxito. Una transferencia entrante reutiliza la ruta de

Creación de Sesión con la bandera de transferencia establecida, no una Modificación de Portadora separada.

Procedimientos Iniciados por PGW

El PGW puede iniciar procedimientos de portadora. El TWAG recibe estos en su socket GTP-C. El TEID de destino en el mensaje es el TEID de control local del TWAG. El TWAG utiliza el mapa de TEID almacenado para encontrar el TEID de control PGW y la identidad de UE. Luego notifica al proceso `Session.Client` correspondiente.

Mensaje PGW	Respuesta TWAG	Comportamiento
Solicitud de Creación de Portadora	Respuesta de Creación de Portadora, causa `request_rejected`	El TWAG rechaza todas las portadoras dedicadas (TSCM). No se crea ninguna portadora.
Comando de Modificación de Portadora	Solicitud de Modificación de Portadora de vuelta al PGW	El TWAG reconoce los nuevos parámetros (APN-AMBR, F-TEID). Si el envío falla, envía una Indicación de Fallo de Modificación de Portadora con causa `system_failure`.
Solicitud de Actualización de Portadora	Respuesta de Actualización de Portadora, causa `request_accepted` por portadora	El TWAG acepta la actualización de QoS y notifica la sesión.
Solicitud de Eliminación de Portadora	Respuesta de Eliminación de Portadora, causa `request_accepted`	Consulte Desmantelamiento de Sesión.

Para un Comando de Modificación de Portadora, el TWAG lee el APN-AMBR y el contexto de portadora (EBI, nuevo F-TEID, QoS). Notifica al proceso de sesión,

luego envía una Solicitud de Modificación de Portadora al PGW para reconocer. Si el contexto de portadora lleva un nuevo F-TEID GTP-U, la sesión vuelve a registrar el túnel GTP-U con los nuevos valores remotos. Para una Solicitud de Actualización de Portadora, el TWAG lee el APN-AMBR y cada contexto de portadora (EBI, QoS), notifica la sesión, luego acepta cada portadora en la respuesta.

Desmantelamiento de Sesión

Eliminar Sesión (Iniciada por TWAG)

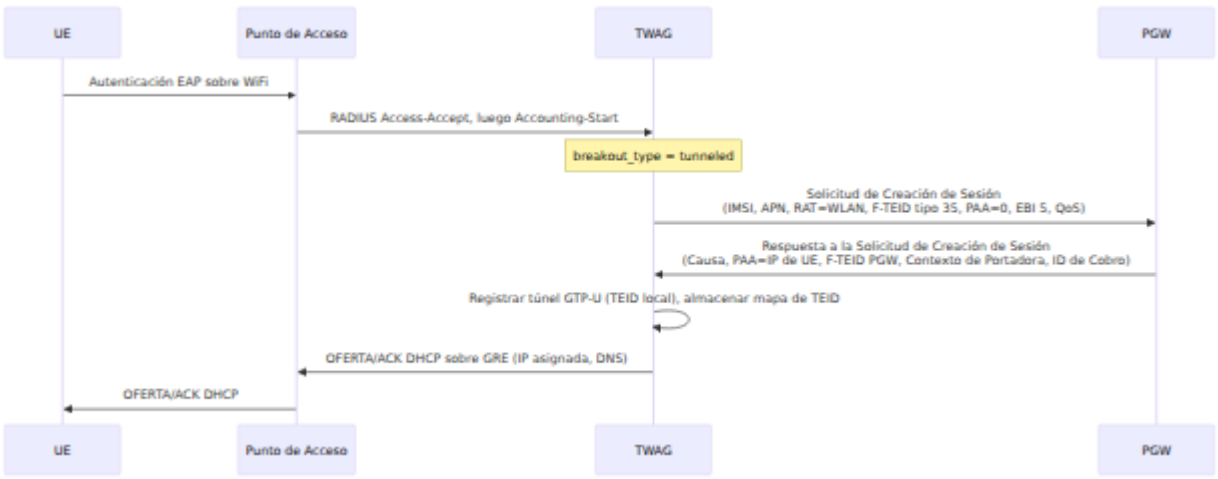
El TWAG elimina la sesión PDN cuando el UE se desconecta. El `Session.Client` llama al cliente GTP-C para el desmantelamiento. El TWAG envía una Solicitud de Eliminación de Sesión (TS 29.274 §7.2.9) al TEID de control PGW. La solicitud lleva:

- ID de Portadora EPS (la EBI por defecto, por defecto 5)
- Causa (por defecto `local_detach`)

El TWAG empareja la Respuesta de Eliminación de Sesión por número de secuencia y verifica el IE de Causa. Luego elimina las entradas del mapa de TEID para ese TEID de control PGW. Si la sesión no tiene TEID de control PGW (por ejemplo, una sesión `nswo` o `local`), el TWAG salta la eliminación GTP-C. Después de la eliminación, el `Session.Client` desregistra el túnel GTP-U, libera el arrendamiento DHCP y cualquier dirección del grupo de IP, desregistra la UE de GRE, y limpia la sesión.

Eliminar Portadora (Iniciada por PGW)

El PGW puede iniciar el desmantelamiento con una Solicitud de Eliminación de Portadora. El TWAG lee el EBI vinculado, la lista de EBI y la causa. Notifica al proceso `Session.Client` correspondiente, que realiza una limpieza solo local. El TWAG no envía una Solicitud de Eliminación de Sesión en este caso, porque el PGW ya eliminó la portadora. Luego el TWAG envía una Respuesta de Eliminación de Portadora con causa `request_accepted`. Si el EBI vinculado es la portadora por defecto, toda la conexión PDN se desmantela.



Gestión de Rutas

Eco

El TWAG ejecuta la gestión de rutas GTPv2-C con Solicitud de Eco y Respuesta de Eco (TS 29.274 §6). El parámetro `echo_enabled` activa esto. Está activado por defecto. El TWAG envía una Solicitud de Eco al PGW cada 60 segundos. Cada Solicitud de Eco lleva el número de secuencia del TWAG. Cada Respuesta de Eco lleva un contador de Recuperación.

Cuando el PGW envía una Solicitud de Eco, el TWAG responde con una Respuesta de Eco que lleva el contador de Recuperación del TWAG.

Estado de la Ruta

El TWAG rastrea el estado de la ruta al PGW. Una Respuesta de Eco exitosa marca la ruta como ACTIVA y restablece el contador de fallos. Fallos consecutivos de Eco incrementan un contador. Después de 3 fallos consecutivos, el TWAG marca la ruta como INACTIVA.

Detección de Reinicio de PGW

El TWAG almacena el contador de Recuperación del PGW de cada Respuesta de Eco. Si el contador cambia, el TWAG detecta un reinicio del PGW. Luego limpia todas las sesiones afectadas. Notifica a cada proceso de sesión con una causa `pgw_restart` y borra el mapa de TEID.

Tiempo de Espera de Solicitud y Retransmisión

El TWAG utiliza un temporizador T3-RESPONSE y reintentos N3-REQUESTS para solicitudes de control.

Temporizador / Contador	Valor	Significado
T3-RESPONSE	3000 ms	Tiempo de espera por una respuesta antes de una retransmisión.
N3-REQUESTS	3	Máximo de retransmisiones antes de que la solicitud falle.

Si una solicitud se agota, el TWAG retransmite el mismo mensaje con el mismo número de secuencia. Reinicia el temporizador. Después de 3 reintentos sin respuesta, el TWAG falla la solicitud. Un fallo de Creación de Sesión devuelve `timeout` al llamador. Las retransmisiones de Eco siguen los mismos límites de temporizador y reintentos. Un fallo de Eco cuenta para el umbral de caída de ruta.

El TWAG abre el socket UDP GTP-C al inicio. Si el socket no puede abrirse, el TWAG reintenta cada 5 segundos.

Configuración

S2a se configura bajo el mapa `gtp`. Este mapa es parte de la configuración de la aplicación `omnitwag` en `runtime.exs`.

```
config :omnitwag,
  gtp: %{
    # Habilitar la interfaz S2a (necesaria para la salida
tunelada)
    enabled: true,

    # Punto final GTP-C PGW
    pgw_ip: "10.5.198.1",
    pgw_port: 2123,

    # IP local del TWAG para F-TEIDs GTP-C y GTP-U
    local_ip: "10.5.198.200",

    # Puerto de plano de usuario GTP-U
    gtpu_port: 2152,

    # Gestión de rutas (Eco)
    echo_enabled: true,

    # Valores por defecto para la Solicitud de Creación de Sesión
    apn: "internet",
    pdn_type: :ipv4
  }
```

Parámetros GTP

Parámetro	Tipo	Requerido	Por defecto	Descripci
`enabled`	Booleano	No	`false`	Habilitar la interfaz S2a. Establecer er `true` para sa tunelada. Cua es `false`, el TWAG no inic cliente GTP-C salta S2a par cada sesión.
`pgw_ip`	Cadena	Sí (si está habilitado)	`127.0.0.1`	Dirección IP C C del PGW. El TWAG envía todos los mensajes GT C a esta dirección.
`pgw_port`	Entero	No	`2123`	Puerto UDP G C del PGW. El puerto estánc GTP-C es 212
`local_ip`	Cadena	No	Detectado automáticamente	IP local del TWAG. Usado F-TEIDs y con fuente para G C y GTP-U. Si se establece, TWAG detect automáticam la IP de salida

Parámetro	Tipo	Requerido	Por defecto	Descripci
`gtpu_port`	Entero	No	`2152`	Puerto UDP G U para el plar de usuario. Consulte Plar de Usuario.
`echo_enabled`	Booleano	No	`true`	Habilitar la gestión de ru Echo GTPv2-(hacia el PGW.
`apn`	Cadena	No	`internet`	Nombre del Punto de Acc para la sesión PDN.
`pdn_type`	Átomo	No	`:ipv4`	Tipo de PDN: `:ipv4`, `:ipv6` `:ipv4v6`.
`mcc`	Cadena	No	Primeros 3 dígitos del IMSI	Código de Pa Móvil para el de Red de Servicio.
`mnc`	Cadena	No	Dígitos 4-5 del IMSI	Código de Re Móvil para el de Red de Servicio.

Parámetros de Emergencia y TWAN

Los siguientes parámetros se leen del entorno de la aplicación `omnitwag`, no del mapa `gtp`.

Parámetro	Tipo	Por defecto	Descripción
`emergency_apn`	Cadena	`sos`	APN utilizado para una sesión de emergencia. La Restricción de APN se establece en 0 para estas sesiones.
`twan_ssid`	Cadena	`TWAG`	SSID colocado en el IE de Identificador de TWAN.
`twan_bssid`	Binario	(no establecido)	BSSID opcional de 6 bytes para el IE de Identificador de TWAN. Omitido si no es un valor válido de 6 bytes.

> Nota: El cliente GTP-C se conecta a un único PGW. No hay selección por APN o multi-PGW en este momento. Para cambiar el PGW, actualice `pgw_ip`.

Tablas de Referencia

Tipos de Mensajes GTPv2-C (S2a)

Mensaje	Dirección	Propósito
Solicitud / Respuesta de Eco	Ambos	Gestión de rutas.
Solicitud / Respuesta de Creación de Sesión	TWAG a PGW	Establecer la sesión PDN.
Solicitud / Respuesta de Modificación de Portadora	TWAG a PGW	Actualizar el F-TEID de la portadora o QoS.
Solicitud / Respuesta de Eliminación de Sesión	TWAG a PGW	Desmantelar la sesión PDN.
Solicitud / Respuesta de Creación de Portadora	PGW a TWAG	Portadora dedicada. Rechazada por el TWAG.
Solicitud / Respuesta de Actualización de Portadora	PGW a TWAG	Actualizar QoS. Aceptada por el TWAG.
Comando de Modificación de Portadora	PGW a TWAG	PGW pide al TWAG que actualice los parámetros de la portadora.
Indicación de Fallo de Modificación de Portadora	TWAG a PGW	Enviado cuando no se puede honrar un Comando de Modificación de Portadora.

Mensaje	Dirección	Propósito
Solicitud / Respuesta de Eliminación de Portadora	PGW a TWAG	Desmantelamiento iniciado por PGW.

Códigos de Causa GTPv2-C

El TWAG utiliza estos valores de causa. Los códigos siguen TS 29.274 §8.4.

Código	Nombre	Significado
2	Desconexión Local	Causa por defecto para una Eliminación de Sesión iniciada por TWAG.
13	Fallo de Red	Fallo de red informado por el par.
16	Solicitud Aceptada	Éxito.
64	Contexto No Encontrado	El par no tiene contexto coincidente.
72	Fallo del Sistema	Usado en una Indicación de Fallo de Modificación de Portadora.
92	Fallo de Autenticación de Usuario	Autenticación rechazada.
93	Acceso Denegado al APN	El APN no está permitido.
94	Solicitud Rechazada	Usado cuando el TWAG rechaza una Solicitud de Creación de Portadora.
96	IMSI/IMEI No Conocido	El suscriptor no es conocido.

Observabilidad

El TWAG emite eventos de telemetría para la actividad de portadoras y desmantelamiento. Estos eventos impulsan las métricas en la Referencia de Métricas.

Evento	Cuándo
`[:twag, :bearer, :modify_command]`	El TWAG recibe un Comando de Modificación de Portadora del PGW.
`[:twag, :bearer, :update]`	El TWAG recibe una Solicitud de Actualización de Portadora del PGW.
`[:twag, :detach, :pgw_delete]`	El TWAG recibe una Solicitud de Eliminación de Portadora del PGW.
`[:twag, :session, :teardown]`	La sesión se desmantela.
`[:twag, :detach, :cleanup_start]`	Comienza el desmantelamiento de la sesión.
`[:twag, :detach, :cleanup_complete]`	Se completa el desmantelamiento de la sesión.

Estado del Cliente

El cliente GTP-C informa un estado instantáneo. Este estado muestra la salud de la ruta S2a.

Campo	Significado
`connected`	El socket UDP GTP-C está abierto.
`pgw_ip` / `pgw_port`	El punto final PGW configurado.
`pending_requests`	Conteo de solicitudes GTP-C en vuelo.
`path_up`	El estado de la ruta Echo hacia el PGW.
`echo_failures`	Conteo de fallos consecutivos de Eco.
`recovery_counter`	El contador de Recuperación del TWAG.

Registro

El cliente GTP-C registra cada procedimiento. Las líneas clave del registro incluyen:

- Envíos y respuestas de Creación de Sesión, Eliminación de Sesión y Modificación de Portadora.
- Cambios en el estado de la ruta: `La ruta hacia PGW ... está ACTIVA` y `La ruta hacia PGW ... está INACTIVA`.
- Detección de reinicio de PGW con los contadores de Recuperación antiguos y nuevos.
- Mensajes de Creación, Actualización, Modificación y Eliminación de Portadora iniciados por PGW.

Solución de Problemas

Fallos en la Creación de Sesión

Síntomas: Las sesiones no se establecen. El registro muestra `GTP Create Session failed`.

Causas posibles:

- El PGW rechaza la solicitud con una causa no exitosa.
- La ruta hacia el PGW está inactiva.
- El APN no está provisionado en el PGW.
- El IMSI no es conocido por el núcleo.

Resolución:

1. Verifique el código de causa devuelto en la tabla de Códigos de Causa.
2. Verifique el estado de la ruta con el estado del cliente (`path_up`).
3. Confirme que el APN está provisionado en el PGW.
4. Confirme que el suscriptor está provisionado en el HSS.

La Ruta hacia el PGW está Inactiva

Síntomas: El registro muestra `La ruta hacia PGW ... está INACTIVA`. Las nuevas sesiones fallan.

Causas posibles:

- El PGW es inalcanzable en UDP 2123.
- Un firewall bloquea GTP-C.
- El PGW no está en ejecución.

Resolución:

1. Confirme que los valores de `pgw_ip` y `pgw_port` son correctos.
2. Confirme que la ruta de red permite UDP 2123 en ambas direcciones.
3. Confirme que el servicio GTP-C del PGW está en ejecución.
4. Espere el próximo Eco. Una Respuesta de Eco exitosa marca la ruta como ACTIVA.

Las Sesiones Se Caen Después de un Reinicio del PGW

Síntomas: Todas las sesiones se desmantelan juntas. El registro muestra `Reinicio de PGW detectado`.

Causas posibles:

- El PGW se reinició y cambió su contador de Recuperación.

Resolución:

1. Este es un comportamiento esperado. El TWAG limpia las sesiones obsoletas después de un reinicio del PGW.
2. Confirme que el PGW es estable. Los UEs se re-autenticaran y restablecerán sesiones.

Las Sesiones Se Establecen Pero No Fluyen Datos

Síntomas: El UE recibe una dirección IP, pero el tráfico no pasa.

Causas posibles:

- El túnel GTP-U o la ruta GRE no están activos.

Resolución:

1. Confirme que `gtp.enabled` y `gre.enabled` son ambos `true` para la salida tunelada.
2. Verifique las estadísticas de GTP-U y GRE en la guía de Plano de Usuario.

Referencias 3GPP

Especificación	Título
TS 29.274	Protocolo de Túnel GPRS Evolucionado para el plano de control (GTPv2-C).
TS 23.402	Mejoras de arquitectura para accesos no 3GPP (TWAN, S2a).
TS 24.302	Acceso al EPC a través de redes de acceso no 3GPP.

Descripción general

Sesiones

Descripción general

Una **sesión** representa un suscriptor autenticado en OmniTWAG. OmniTWAG identifica una sesión por la identidad del suscriptor (el IMSI o nombre de usuario). OmniTWAG mantiene las sesiones en memoria. La API de Sesiones proporciona el estado de sesión en vivo.

La vista de Sesiones también transporta el tráfico del suscriptor. OmniTWAG fusiona el registro de contabilidad RADIUS en la sesión por `Acct-Session-Id`. Por lo tanto, una sesión es la única fuente tanto para el estado de un suscriptor como para el tráfico del suscriptor. No hay un punto final de contabilidad separado. La contabilidad es parte de la sesión.

Tabla de Contenidos

- Conceptos
- Ciclo de vida del estado de autenticación
- Modelo de sesión
- Referencia de estado de autenticación
- API REST
- Acciones por sesión
- Ejemplos de JSON
- Interfaz web
- Resolución de problemas

Conceptos

Cada suscriptor autenticado tiene una sesión. OmniTWAG crea la sesión cuando una solicitud de acceso RADIUS inicia una autenticación EAP. OmniTWAG actualiza la sesión a medida que avanza la autenticación y se establece la ruta de datos.

Una sesión rastrea la identidad del suscriptor, el punto de acceso, la MAC del cliente, la dirección IP del UE asignada, el método y fase EAP, y el estado de GTP y DNS. Cuando llega la contabilidad RADIUS para el suscriptor, OmniTWAG fusiona los contadores de tráfico en la sesión.

La forma de la sesión es abierta. Las claves que están presentes dependen del método de autenticación y de cuán avanzada esté la sesión. Una sesión que solo ha comenzado EAP tiene menos claves que una sesión que ha alcanzado RADIUS Accept y ha establecido una ruta de datos GTP y DHCP.

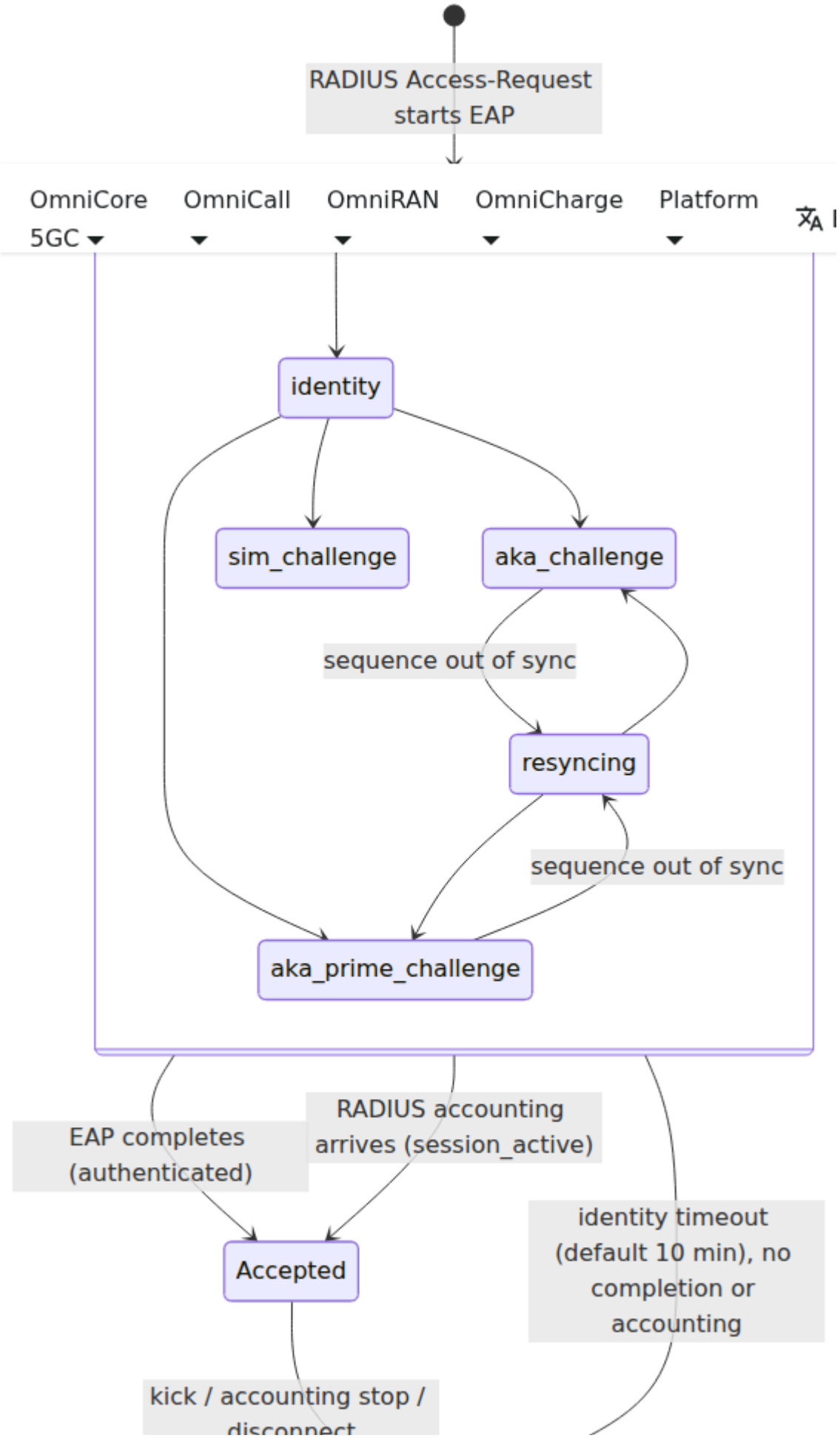
Ciclo de vida del estado de autenticación

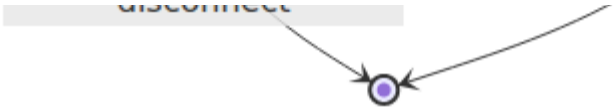
Una sesión tiene un estado de autenticación. El estado se deriva de la fase EAP de la sesión (`eap_phase`).

- Una sesión está **Autenticando** mientras `eap_phase` es una fase de autenticación en progreso. Las fases en progreso son `identity`, `aka_challenge`, `aka_prime_challenge`, `sim_challenge` y `resyncing`.
- Una sesión está **Aceptada** una vez que está autenticada, o una vez que está activa (`session_active`).

Una sesión que recibe contabilidad RADIUS se promueve a activa. La contabilidad fluye solo para un suscriptor que ya está autenticado. La llegada de la contabilidad es por lo tanto prueba de que el suscriptor está activo, por lo que OmniTWAG marca la sesión como Aceptada.

Una sesión que permanece en una fase de autenticación en progreso es eliminada. Si una sesión comenzó la autenticación, nunca se completó y nunca recibió contabilidad, OmniTWAG destruye la sesión después de un tiempo de espera. Esto evita que un intento de autenticación a medio terminar permanezca como Autenticando para siempre. El tiempo de espera predeterminado es de 10 minutos.


RADIUS Access-Request starts EAP
OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
identity
sim_challenge
aka_challenge
sequence out of sync
resyncing
sequence out of sync
aka_prime_challenge
EAP completes (authenticated)
RADIUS accounting arrives (session_active)
identity timeout (default 10 min), no completion or accounting
Accepted
kick / accounting stop /



Modelo de sesión

Una sesión tiene los siguientes campos. La forma es abierta, por lo que una sesión también puede llevar otras claves de GTP o de carga. Cada campo a continuación se representa como una cadena imprimible, un entero o una lista. Los valores binarios se representan como cadenas imprimibles o como hexadecimales. Las tuplas IP se representan como cadenas de punto-cuádruple.

Campo	Tipo	Descripción
`identity`	String	Identidad de la sesión. Este es el IMSI o el nombre de usuario. OmniTWAG identifica la sesión por este valor.
`imsi`	String	IMSI del suscriptor.
`mac_address`	String	Dirección MAC del UE.
`ue_ip`	String	Dirección IP del UE asignada, como una cadena de punto-cuádruple.
`nas_ip`	String	IP de origen del AP (RADIUS `NAS-IP-Address`).
`called_station_id`	String	Identificador del punto de acceso (RADIUS `Called-Station-Id`).
`calling_station_id`	String	MAC del cliente (RADIUS `Calling-Station-Id`).
`acct_session_id`	String	ID de sesión de contabilidad RADIUS. OmniTWAG fusiona el registro de contabilidad por esta clave.
`eap_phase`	String	Fase EAP actual. Este valor determina el estado de autenticación. Ver Referencia de estado de autenticación.
`eap_type`	String	Método EAP en uso, por ejemplo `aka` o `aka_prime`.
`pgw_ip`	String	Dirección IP del PGW, como una cadena de punto-cuádruple.
`bearer_id`	Integer	ID de portadora EPS.

Campo	Tipo	Descripción
`dns_servers`	List of String	Servidores DNS asignados al UE.
`created_at`	Integer	Hora de creación de la sesión, en milisegundos desde la época Unix.
`last_activity`	Integer	Hora de la última actividad, en milisegundos desde la época Unix.
`established_at`	Integer	Hora de establecimiento de la sesión, en milisegundos desde la época Unix.
`acct_input_octets`	Integer	Bytes recibidos del UE. Fusionados desde la contabilidad RADIUS.
`acct_output_octets`	Integer	Bytes enviados al UE. Fusionados desde la contabilidad RADIUS.
`acct_input_packets`	Integer	Paquetes recibidos del UE. Fusionados desde la contabilidad RADIUS.
`acct_output_packets`	Integer	Paquetes enviados al UE. Fusionados desde la contabilidad RADIUS.
`acct_session_time`	Integer	Duración de la sesión de contabilidad, en segundos. Fusionados desde la contabilidad RADIUS.
`acct_status_type`	Integer	––Último estado de contabilidad. `1` es Inicio, `2` es Detener, `3` es Interino.
`acct_last_update`	Integer	Última hora de actualización de contabilidad, en milisegundos desde la época Unix.

Los campos `acct_*` están presentes solo después de que el registro de contabilidad del suscriptor ha sido fusionado en la sesión.

Referencia de estado de autenticación

OmniTWAG mapea el valor de `eap_phase` a un estado mostrado. La tabla a continuación enumera el mapeo.

`eap_phase`	Estado mostrado	Estado de autenticación
`identity`	Identidad	Autenticando
`aka_challenge`	Desafío AKA	Autenticando
`aka_prime_challenge`	Desafío AKA'	Autenticando
`sim_challenge`	Desafío SIM	Autenticando
`resyncing`	Re-sincronizando	Autenticando
`authenticated`	Aceptado	Aceptado
`session_active`	Aceptado	Aceptado
`accepted`	Aceptado	Aceptado
`rejected`	Rechazado	Rechazado

Cualquier otro valor de `eap_phase` cuenta como Autenticando.

API REST

La API REST de OmniTWAG proporciona estos puntos finales a través de HTTPS en el puerto de API configurado (`8444` por defecto). La ruta base es `/sessions`.

Método	Ruta	Acción	Descripción
GET	`/sessions`	index	Lista las sesiones autenticadas. La respuesta está paginada y es buscable.
GET	`/sessions/{id}`	show	Obtiene una sesión por `identity`.
DELETE	`/sessions/{id}`	delete	Elimina la sesión y desconecta al suscriptor.
PATCH	`/sessions/{id}`	action	Realiza una acción en la sesión. La acción está en el cuerpo JSON.
PUT	`/sessions/{id}`	action	Realiza una acción en la sesión. La acción está en el cuerpo JSON.

El parámetro de ruta `{id}` es la `identity` de la sesión (el IMSI o nombre de usuario).

Búsqueda y paginación

El punto final de índice acepta parámetros de paginación y búsqueda.

Parámetro	Tipo	Descripción
`offset`	Integer	Número de sesiones a omitir antes de la página.
`limit`	Integer	Número máximo de sesiones en la página.
`q`	String	Término de búsqueda. OmniTWAG compara el término con los campos de búsqueda a continuación.

El término `q` se compara con estos campos de sesión.

Campo de búsqueda	Coincidencias
`identity`	Identidad de la sesión (IMSI o nombre de usuario).
`imsi`	IMSI del suscriptor.
`nas_ip`	IP de origen del AP (`NAS-IP-Address`).
`called_station_id`	Identificador del punto de acceso.
`calling_station_id`	MAC del cliente.
`mac_address`	Dirección MAC del UE.
`ue_ip`	Dirección IP del UE asignada.

Cuerpo de solicitud de acción

`PATCH` y `PUT` aceptan un cuerpo JSON con un campo.

Campo	Tipo	Requerido	Descripción
`action`	String	Sí	La acción a realizar. Debe ser `kick`, `start_accounting` o `stop_accounting`.

Ver Acciones por sesión para lo que hace cada acción.

Respuestas

Estado	Significado
`200 OK`	La solicitud fue exitosa (index, show, delete, action).
`404 Not Found`	No hay sesión con la `identity` dada.
`422 Unprocessable Entity`	La acción no es una de `kick`, `start_accounting` o `stop_accounting`.

Una acción exitosa devuelve un cuerpo con un campo `status` y la `identity`. Una eliminación devuelve `{"status": "deleted", "identity": ...}`.

Acciones por sesión

Una acción ejecuta una operación administrativa contra una sesión en vivo. Envía la acción en el cuerpo JSON de una solicitud `PATCH` o `PUT` a `/sessions/{id}`.

Acción	Efecto	Resultado `status`
`kick`	Envía un RADIUS CoA-Disconnect al punto de acceso y destruye la sesión.	`kicked`
`start_accounting`	Marca la sesión como activa y comienza la contabilidad. Esto promueve la sesión a Aceptada.	`accounting_started`
`stop_accounting`	Termina la contabilidad, como si el punto de acceso enviara Accounting-Stop. Esto destruye la sesión.	`accounting_stopped`

Tanto `kick` como `stop_accounting` destruyen la sesión. Usa `kick` para desconectar activamente al suscriptor en el punto de acceso. Usa `stop_accounting` para terminar la sesión como lo haría la red.

Ejemplos de JSON

Una sesión con tráfico

Solicitud:

```
GET /sessions/001010000000001
```

Respuesta:

```
{
  "identity": "001010000000001",
  "imsi": "001010000000001",
  "mac_address": "AA:BB:CC:DD:EE:FF",
  "ue_ip": "10.45.0.2",
  "nas_ip": "192.0.2.10",
  "called_station_id": "AA-BB-CC-DD-EE-FF:WiFi-Calling",
  "calling_station_id": "11-22-33-44-55-66",
  "acct_session_id": "5A2E1F0C",
  "eap_phase": "session_active",
  "eap_type": "aka_prime",
  "pgw_ip": "192.0.2.20",
  "bearer_id": 5,
  "dns_servers": ["8.8.8.8", "8.8.4.4"],
  "created_at": 1712000000000,
  "last_activity": 1712000600000,
  "established_at": 1712000004000,
  "acct_input_octets": 1048576,
  "acct_output_octets": 5242880,
  "acct_input_packets": 1200,
  "acct_output_packets": 3400,
  "acct_session_time": 600,
  "acct_status_type": 3,
  "acct_last_update": 1712000600000
}
```

Una acción de kick

Solicitud:

```
PATCH /sessions/001010000000001
```

```
{
  "action": "kick"
}
```

Respuesta (`200 OK`):

```
{
  "status": "kicked",
  "identity": "001010000000001"
}
```

Una acción desconocida

Solicitud:

```
{
  "action": "reboot"
}
```

Respuesta (`422 Unprocessable Entity`):

```
{
  "error": "Unknown action \"reboot\" (expected
kick|start_accounting|stop_accounting)",
  "identity": "001010000000001"
}
```

Interfaz web

El panel de control tiene una página de **Clientes RADIUS**. La página lista cada sesión con su estado de autenticación. La página muestra el conteo de sesiones Aceptadas, Autenticando y Rechazadas. Usa la página para revisar sesiones en vivo de un vistazo.

Resolución de problemas

Una sesión está atascada en "Autenticando"

Síntomas: Una sesión muestra el estado Autenticando. El estado no cambia a Aceptada. La sesión permanece en una fase EAP como `identity`,

`aka_challenge`, `aka_prime_challenge`, `sim_challenge` o `resyncing`.

Causas posibles:

- El suscriptor comenzó la autenticación EAP pero nunca la completó.
- El punto de acceso dejó de enviar respuestas EAP.
- El suscriptor falló en la autenticación y no intentó de nuevo.

Cómo se resuelve:

1. OmniTWAG activa un tiempo de espera cuando la sesión entra en una fase de autenticación en progreso.
2. Si la sesión nunca completa la autenticación y nunca recibe contabilidad, OmniTWAG destruye la sesión cuando se activa el tiempo de espera. El tiempo de espera predeterminado es de 10 minutos.
3. Una sesión que completa la autenticación se mueve a Aceptada. Una sesión que recibe contabilidad RADIUS se promueve a activa y también se mueve a Aceptada.

Para eliminar una sesión atascada de inmediato, envía una acción `kick` o una solicitud `DELETE` a `/sessions/{id}`.

Una sesión no muestra tráfico

Síntomas: Una sesión está Aceptada, pero los campos `acct_*` están ausentes o en cero.

Causas posibles:

- No se ha recibido ningún registro de contabilidad RADIUS para el suscriptor aún.
- El registro de contabilidad tiene un `Acct-Session-Id` diferente al de la sesión.

Resolución:

1. Confirma que el punto de acceso envía contabilidad RADIUS a OmniTWAG.
2. Confirma que el `Acct-Session-Id` de la contabilidad coincide con el `acct_session_id` de la sesión. OmniTWAG fusiona el tráfico por esta clave.

3. Espera la próxima actualización interina de contabilidad, luego lee la sesión nuevamente.

Descripción general

Plano de Usuario y Ruta de Datos

Descripción general

OmniTWAG separa el plano de control del plano de usuario. El plano de control ejecuta la autenticación y la señalización de sesión. El plano de usuario transporta el tráfico real del UE. Esta página describe el plano de usuario. Explica cómo OmniTWAG mueve los paquetes del UE, cómo construye y desmantela el túnel GTP-U, y cómo reenvía los paquetes de bajada de nuevo al UE.

Para la señalización de sesión GTP-C que establece la ruta de datos en túnel, consulte S2a / GTP-C. Para la vista completa del sistema, consulte Arquitectura.

Tabla de Contenidos

- Plano de Control y Plano de Usuario
- Modos de Ruta de Datos
- Ruta de Datos GRE
- Túnel GTP-U
- Reenvío de Bajada
- Portadores y Plantillas de Flujo de Tráfico
- Manejo de Errores
- MTU y Fragmentación
- Configuración
- Observabilidad

Plano de Control y Plano de Usuario

El plano de control y el plano de usuario funcionan como procesos separados.

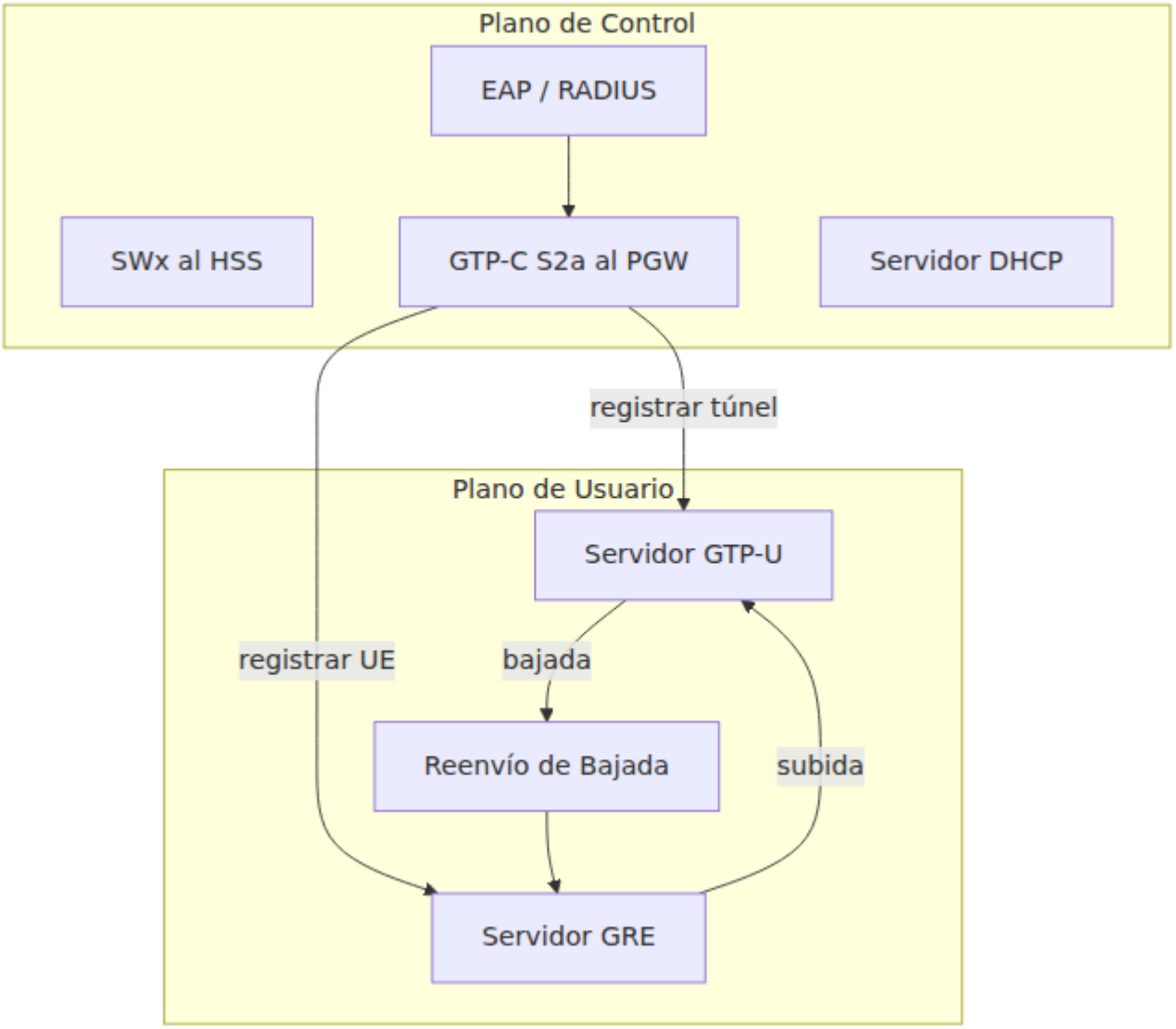
El **plano de control** realiza el siguiente trabajo:

- Autenticación EAP a través de RADIUS con el punto de acceso.
- Recuperación del vector de autenticación del HSS a través de Diameter SWx.
- Señalización de sesión GTP-C al PGW en la interfaz S2a. Esto crea y elimina la conexión PDN y los portadores. Consulte S2a / GTP-C.
- Entrega de dirección DHCP al UE. Consulte Gestión de Direcciones IP.

El **plano de usuario** realiza el siguiente trabajo:

- Túnel GRE hacia el punto de acceso.
- Túnel GTP-U hacia el PGW en la interfaz S2a-U, según 3GPP TS 29.281 (modo en túnel solamente).
- Reenvío de paquetes de bajada hacia el UE.
- Clasificación de paquetes de subida en portadores.

El plano de control establece el estado del plano de usuario. Después de la autenticación, el proceso de sesión crea la sesión GTP-C (en modo en túnel), registra el túnel GTP-U y registra el UE con el servidor GRE. Luego, el plano de usuario transporta el tráfico.



Modos de Ruta de Datos

OmniTWAG resuelve un modo de plano de datos por punto de acceso (AP) a partir de las credenciales RADIUS que coinciden con el AP. El valor de `breakout_type` en las credenciales selecciona uno de los tres modos. Consulte Credenciales RADIUS para los ajustes de `breakout_type`, `address_source` y `ip_pool_id`.

Modo (`breakout_type`)	IP del UE asignada por	TWAG transporta tráfico	Ruta
`tunneled`	PGW	Sí	GRE desde el AP hasta TWAG, luego GTP-U en S2a hacia el PGW.
`nswo`	TWAG, desde un grupo de IP	Sí	GRE desde el AP hasta TWAG, luego salida local a internet (NAT) en el TWAG.
`local`	Red local en el AP	No	El AP posee el plano de datos. El TWAG no transporta paquetes ni asigna IP.

La ruta de datos GRE se aplica tanto a `tunneled` como a `nswo`. En ambos modos, el AP envía tramas UE al TWAG en GRE, y el TWAG envía paquetes de bajada de nuevo al AP en GRE. La diferencia está en la subida después de GRE:

- **`tunneled`**: el TWAG encapsula el paquete de subida en GTP-U y lo envía al PGW. El PGW asigna la IP del UE y aplica políticas y cargos.
- **`nswo`**: el TWAG rompe la subida localmente hacia internet con una regla de enmascaramiento NAT. El TWAG asigna la IP del UE desde un grupo de IP. No hay PGW ni sesión GTP-C. Se aplica el Gy Cobro en Línea del lado del TWAG.

En modo `local`, el TWAG ni asigna una IP ni transporta tráfico, por lo que ni la ruta GRE ni la GTP-U se utilizan para ese AP.

Modo en Túnel (GTP-U al PGW)

En modo en túnel, OmniTWAG transporta el tráfico del UE al PGW. El PGW asigna la dirección IP del UE, aplica la política y el cobro, y proporciona acceso a internet. Esta es la interfaz S2a según 3GPP TS 23.402 Sección 16.

La ruta de subida es:

```
UE --WiFi--> AP --GRE--> TWAG --GTP-U S2a--> PGW --> Internet
```

La ruta de bajada es la inversa:

```
Internet --> PGW --GTP-U S2a--> TWAG --GRE--> AP --WiFi--> UE
```

Para usar este modo, establezca `gtp.enabled` en `true` y `gre.enabled` en `true`, y establezca el `breakout_type` de las credenciales del AP en `tunneled`. Consulte Configuración.

Salida Local (NSWO)

La Salida WLAN No Sincronizada (NSWO) es una salida local. El tráfico del UE sale directamente del TWAG a internet. El tráfico no va al PGW. El TWAG asigna la IP del UE y aplica enmascaramiento NAT. Esto es según 3GPP TS 23.402 Sección 16.2.

OmniTWAG tiene dos formas de alcanzar NSWO:

1. **Salida por AP (preferida)**: establezca el `breakout_type` de las credenciales del AP en `nswo`. La dirección del UE proviene del grupo de IP nombrado por `ip_pool_id`. El `address_source` de la credencial selecciona cómo:
    - `pool` pre-asigna una dirección fija del grupo y registra el UE para la bajada GRE de inmediato.
    - `dhcp` vincula la MAC del UE al grupo y permite que el servidor DHCP del TWAG asigne una dirección en el DHCP DISCOVER. La bajada GRE se registra cuando se conoce la dirección.

2. **Manejador NSWO Global**: el `Twag.NSWO.Handler` proporciona salida local para cada sesión cuando no se utiliza GTP. Se selecciona con `nswo.enabled: true` (y se aplica cuando `gtp.enabled: false`). El manejador gestiona un único grupo de IP local y configura una regla de enmascaramiento NAT `iptables`. Al establecer la sesión, el TWAG asigna una dirección del grupo local, la entrega al UE a través de DHCP, y registra el UE para la bajada GRE, sin PGW / sesión GTP-C. La asignación se libera al desmantelar la sesión.

Para ambos caminos, la subida se enmascara NAT hacia internet y la bajada regresa a través de la ruta GRE. Consulte Gestión de Direcciones IP y Grupos de IP para la asignación de direcciones.

El manejador NSWO global lee estos parámetros del mapa `nswo`:

Parámetro	Tipo	Predeterminado	Descripción
`enabled`	Booleano	`false`	Iniciar el manejador NSWO global y enrutar sesiones a través de salida local.
`local_pool`	Cadena (CIDR)	`10.100.0.0/16`	Grupo del que se asigna la dirección del UE.
`nat_interface`	Cadena	`eth0`	Interfaz de salida para la regla de enmascaramiento NAT.
`dns_servers`	Lista	`["8.8.8.8", "8.8.4.4"]`	Servidores DNS anunciados al UE. La salida local no tiene PCO de PGW para transportarlos.

Ruta de Datos GRE

El servidor GRE (`Twag.Gre.Server`) es el punto final del plano de usuario hacia el punto de acceso. El punto de acceso envía las tramas del UE al TWAG en GRE, según RFC 2784 y RFC 2890. El servidor GRE también envía los paquetes de bajada y las respuestas DHCP de vuelta al punto de acceso. GRE es la ruta que enfrenta al AP para ambos `tunneled` y `nswo`.

El servidor GRE utiliza un socket en bruto para el protocolo IP 47 (GRE), abierto con `:socket.open/3`. El host debe otorgar la capacidad `CAP_NET_RAW`. Si el socket no se abre, el servidor registra un error y vuelve a intentar cada 10 segundos. GRE está desactivado por defecto. Para activarlo, establezca `gre.enabled` en `true`.

Clasificación GRE Entrante

El servidor GRE lee el encabezado IP externo, lo elimina y decodifica el encabezado GRE. El servidor admite dos tipos de carga útil GRE:

Protocolo GRE	Significado	Manejo
`0x6558`	Puenteo Ethernet Transparente	Extraer el marco Ethernet interno, luego clasificar por etype.
`0x0800`	IPv4	Tratar como un paquete IPv4 de subida.

Para un marco de Puenteo Ethernet Transparente, el servidor lee el etype interno:

Ethertype	Significado	Manejo
`0x0800`	IPv4	Si el paquete es DHCP, páselo al servidor DHCP. Si no, reenvíelo a la ruta de subida.
`0x0806`	ARP	Manejar localmente. No reenviar.
`0x86DD`	IPv6	Reenviar a la ruta de subida.

El servidor mantiene un mapa de la dirección MAC del UE a la IP del punto de acceso. El servidor aprende este mapa de la MAC de origen de cada marco entrante. Este mapa permite al servidor enrutar los marcos de bajada de regreso al punto de acceso correcto.

Manejo de Subida

Para un paquete IP de subida, el servidor GRE lee la IP de origen. Esta es la IP del UE. El servidor busca la IP del UE en la tabla de UE registrados para encontrar la identidad de la sesión. Encuentra el proceso de sesión en el registro y lee el TEID local de la sesión. Luego envía el paquete de forma asíncrona al servidor GTP-U con ese TEID local (en modo en túnel). El envío asíncrono evita un interbloqueo entre los procesos GRE y GTP-U.

Si no hay sesión para la IP del UE, o la sesión aún no tiene túnel GTP-U, el servidor descarta el paquete. Un paquete IP pequeño (menos de 20 bytes) también se descarta.

> Nota: Para la salida `nswo`, el grupo de IP local y el enmascaramiento NAT llevan la subida a internet. El servidor GRE registra el UE para el enrutamiento de bajada de la misma manera que en modo en túnel.

Registro del UE para Bajada

Cuando la sesión se vuelve activa, el proceso de sesión registra el UE con el servidor GRE. El registro contiene la IP del UE, la MAC del UE, la IP del punto de acceso, y la identidad de la sesión. El servidor GRE utiliza estos datos para

enrutar los paquetes de bajada. Al desmantelar la sesión, el proceso de sesión desregistra el UE.

Túnel GTP-U

El servidor GTP-U (`Twag.Gtp.GtpU`) es el punto final del plano de usuario hacia el PGW en modo en túnel. Ejecuta un único socket UDP en el puerto `2152` y mantiene una tabla de túneles. El servidor implementa GTPv1-U según 3GPP TS 29.281.

Manejo de TEID

Cada túnel tiene un TEID local y un TEID remoto. El TEID remoto es el TEID del PGW. El TWAG utiliza el TEID remoto en el encabezado de los paquetes que envía al PGW. El PGW utiliza el TEID local en el encabezado de los paquetes que envía al TWAG. El TWAG busca el TEID local en la tabla de túneles para encontrar el túnel.

Cuando el proceso de sesión recibe la Respuesta de Creación de Sesión del PGW, registra el túnel. El servidor GTP-U genera el TEID local. El TWAG utiliza un valor aleatorio criptográficamente seguro de 32 bits para el TEID local, y nunca utiliza cero. Esto sigue el requisito de TEID no predecible en 3GPP TS 29.274.

Un túnel registrado contiene estos campos:

Campo	Descripción
`local_teid`	El TEID del lado del TWAG. El PGW envía paquetes de bajada con este valor.
`remote_teid`	El TEID del lado del PGW. El TWAG envía paquetes de subida con este valor.
`remote_ip`	La dirección IP GTP-U del PGW.
`remote_port`	El puerto UDP GTP-U del PGW. Predeterminado `2152`.
`session_id`	La identidad de la sesión para el túnel.

Encapsulación de Subida

Para cada paquete UE de subida, el servidor GTP-U construye un encabezado GTPv1-U G-PDU y coloca el encabezado delante del paquete IP interno. El encabezado lleva la versión GTP (1), el tipo de protocolo (GTP), el tipo de mensaje (G-PDU, 255), la longitud de la carga útil, y el TEID remoto. El servidor envía el paquete a la IP y puerto del PGW a través del socket UDP. El servidor GRE llama al servidor GTP-U en la ruta de envío asíncrono, lo que evita un interbloqueo entre los dos procesos.

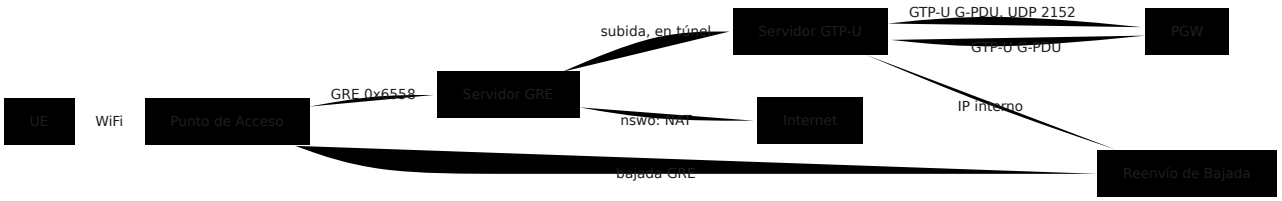
Desencapsulación de Bajada

Para cada paquete que llega al socket GTP-U, el servidor lee el encabezado GTP-U. El servidor actúa según el tipo de mensaje:

Mensaje	Tipo	Acción
G-PDU	255	Buscar el túnel por el TEID local, luego reenviar el paquete IP interno al reenvío de bajada.
Echo Request	1	Responder con un Echo Response.
Echo Response	2	Aceptar e ignorar.
Error Indication	26	Aceptar e ignorar.
End Marker	254	Aceptar e ignorar.

Para un G-PDU, el servidor elimina el encabezado GTP-U y cualquier campo de encabezado opcional (número de secuencia, número N-PDU y tipo de siguiente encabezado de extensión). Luego, el servidor pasa el paquete IP interno a `Twag.Downlink`. Consulte Reenvío de Bajada.

Si un G-PDU llega para un TEID que no tiene túnel, el servidor descarta el paquete y envía una Indicación de Error GTP-U de vuelta a la fuente. Consulte Manejo de Errores.



Reenvío de Bajada

El reenvío de bajada (`Twag.Downlink`) recibe los paquetes IP internos de la desencapsulación GTP-U. El reenvío envía cada paquete hacia el UE. El reenvío tiene dos rutas y prefiere GRE.

Ruta GRE (Preferida)

Si el servidor GRE está en funcionamiento, el reenvío envía el paquete a través de GRE. El reenvío lee la IP de destino del paquete interno. Esta es la IP del UE. El reenvío llama al servidor GRE. El servidor GRE busca la IP del UE en la tabla de UE registrados, envuelve el paquete en un marco GRE de Puenteo Ethernet Transparente, y lo envía a la IP del punto de acceso. El punto de acceso entrega el marco al UE a través de WiFi.

Para el marco Ethernet de bajada, el servidor GRE utiliza la MAC del UE como la MAC de destino. El servidor utiliza una MAC administrada localmente fija (`02:00:00:00:00:01`) como la MAC de origen.

Ruta de Socket en Bruto (Fallback)

Si el servidor GRE no está en funcionamiento, o el envío GRE falla, el reenvío utiliza un socket en bruto. El reenvío inyecta el paquete IP en bruto. El kernel luego enruta el paquete al UE desde la tabla de enrutamiento local.

El reenvío abre dos sockets en bruto: uno para IPv4 y uno para IPv6. El reenvío establece la opción `IP_HDRINCL` (y `IPV6_HDRINCL` para IPv6). Esta opción significa que el reenvío proporciona el encabezado IP completo. En algunos kernels, el socket aún funciona sin esta opción, por lo que un fallo al establecer la opción no es fatal.

El host debe otorgar la capacidad `CAP_NET_RAW` para los sockets en bruto. Si un socket no se abre, el reenvío registra una advertencia y desactiva esa familia de direcciones. Si ni GRE ni un socket en bruto están disponibles, el reenvío cuenta y descarta los paquetes, y registra una advertencia una vez.

Selección de Ruta

El parámetro `prefer_gre` establece el orden de selección de la ruta.

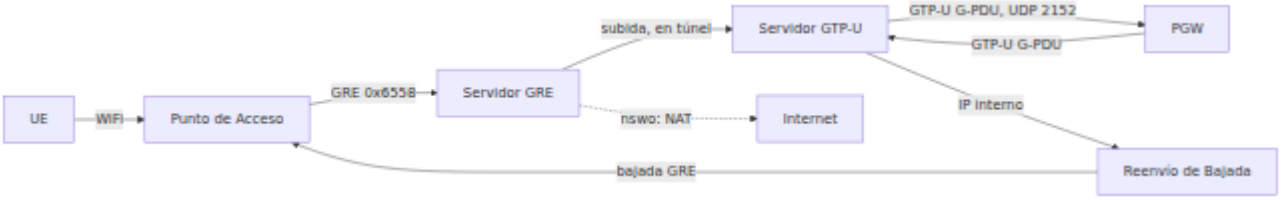
- `prefer_gre: true` (predeterminado): el reenvío intenta el túnel GRE primero y recurre al socket en bruto si GRE no está disponible o el envío GRE falla.

- `prefer_gre: false`: el reenvío intenta el socket en bruto para la familia de direcciones del paquete primero y utiliza GRE solo cuando ese socket en bruto no está disponible.

Comprobaciones de Paquete

El reenvío descarta un paquete en estos casos:

- El reenvío está desactivado por configuración.
- No hay socket disponible, y GRE no está disponible.
- El paquete es pequeño (menos de 20 bytes).
- El reenvío no puede leer la IP de destino.



Portadores y Plantillas de Flujo de Tráfico

El administrador de portadores (`Twag.Bearer.Manager`) mantiene los portadores para una sesión. Cada portador se mapea a una Identidad de Portador EPS (EBI). Un portador contiene los TEIDs GTP-U, los filtros de paquetes de la Plantilla de Flujo de Tráfico (TFT), y la clase QoS (QCI). Un portador es el portador predeterminado. Esto sigue 3GPP TS 23.402 Sección 16.1.6.1.

Portadores Predeterminados y Dedicados

Tipo de portador	Filtros TFT	Propósito
Predeterminado	Ninguno	Transporta todo el tráfico que no coincide con un portador dedicado.
Dedicado	Uno o más	Transporta el tráfico que coincide con sus filtros de paquetes TFT.

Si se elimina el portador predeterminado, el administrador de portadores señala un desmantelamiento completo de la sesión. La eliminación de un portador dedicado elimina solo ese portador.

Filtros de Paquetes TFT

Un TFT es una Plantilla de Flujo de Tráfico según 3GPP TS 24.008 Sección 10.5.6.12. Un TFT contiene una operación y una lista de filtros de paquetes. Cada filtro de paquete contiene una dirección, una precedencia y una lista de componentes. El analizador TFT admite estos tipos de componentes:

Componente	Coincidencia
Dirección y máscara remota IPv4	La dirección de destino del paquete.
Dirección y máscara local IPv4	La dirección de origen del paquete.
Dirección y prefijo remotos/locales IPv6	La dirección IPv6.
Identificador de protocolo	El número de protocolo IP.
Puerto local único / rango de puertos locales	El puerto de origen del UE.
Puerto remoto único / rango de puertos remotos	El puerto de destino.
Tipo de servicio / clase de tráfico	El byte TOS, con una máscara.
Índice de parámetro de seguridad	No coincidente en la subida TWAN.
Etiqueta de flujo	No coincidente (solo IPv6).

El analizador admite las operaciones TFT: crear, eliminar, agregar filtros, reemplazar filtros, eliminar filtros y ninguna operación. El administrador de portadores aplica estas operaciones cuando el PGW envía una Solicitud de Actualización de Portador.

Clasificación de Subida

El clasificador de subida vincula cada paquete de subida a un portador. Las reglas son:

1. Leer la IP de origen, la IP de destino, el protocolo, los puertos y el TOS del paquete.

2. Probar el paquete contra los filtros TFT de los portadores dedicados.
3. Un filtro coincide solo si todos sus componentes coinciden. Esta es la lógica AND.
4. Si más de un filtro coincide, seleccionar el filtro con el número de precedencia más bajo. Un número de precedencia más bajo es una prioridad más alta.
5. Si ningún portador dedicado coincide, seleccionar el portador predeterminado.

Un filtro vacío (un filtro sin componentes) coincide con todos los paquetes. Esto es un comodín.

> **Nota:** El administrador de portadores y el clasificador TFT están presentes para el soporte de portadores dedicados. La ruta GRE de subida reenvía el tráfico en el TEID local de la sesión. Un completo desglose de múltiples portadores en subida depende del PGW para crear portadores dedicados para la sesión, que el TWAG rechaza bajo TSCM (ver S2a / GTP-C).

Manejo de Errores

Eco GTP-U

El PGW y el TWAG utilizan la Solicitud de Eco GTP-U y la Respuesta de Eco para probar la ruta del túnel. Cuando el servidor GTP-U recibe una Solicitud de Eco, responde con una Respuesta de Eco. El proceso GTP-C S2a ejecuta su propio eco cuando `gtp.echo_enabled` es `true`. Consulte S2a / GTP-C.

Indicación de Error GTP-U

Si un G-PDU llega para un TEID que no tiene túnel, el servidor GTP-U envía una Indicación de Error GTP-U de vuelta a la fuente. La Indicación de Error lleva el TEID desconocido y la dirección del par GTP-U del TWAG. Esto le indica al PGW que deje de enviar tráfico en el túnel muerto. El servidor también descarta el paquete y aumenta el conteo de errores.

Marcador de Fin GTP-U

El servidor GTP-U puede enviar un Marcador de Fin en un túnel antes de desmantelar el túnel. Esto es para una transferencia. El Marcador de Fin le dice al par que no se enviarán más paquetes por la antigua ruta.

Errores de Decodificación

Si el servidor GTP-U no puede decodificar un paquete, registra una advertencia y aumenta el conteo de errores. Si el servidor GRE no puede decodificar un paquete, o recibe un protocolo o etype desconocido, cuenta el evento y descarta el paquete.

MTU y Fragmentación

El servidor GRE construye el encabezado IPv4 externo para cada paquete GRE de bajada. El número de protocolo GRE es 47.

La encapsulación GTP-U y GRE añade bytes de encabezado al paquete del UE. Para evitar la fragmentación, el punto de acceso debe reducir el TCP MSS y el MTU del túnel. La configuración de referencia del punto de acceso Cisco establece `mss 1410` y `mtu 1500` en el túnel GRE. Consulte Arquitectura para la configuración del punto de acceso.

Configuración

Los procesos del plano de usuario leen la configuración de la aplicación `:omnitwag`. Los bloques relevantes son `gtp`, `gre`, y `downlink`. El modo de salida del AP proviene de las credenciales RADIUS, no de estos bloques. Consulte Credenciales RADIUS.

```
config :omnitwag,
  # GTP-U y GTP-C hacia el PGW (S2a) - modo en túnel
  gtp: %{
    enabled: false,
    pgw_ip: "10.5.198.1",
    pgw_port: 2123,
    local_ip: "10.5.198.200",
    gtpu_port: 2152,
    echo_enabled: true
  },

  # GRE hacia el punto de acceso
  gre: %{
    enabled: false,
    local_ip: "10.5.198.200"
  },

  # Reenvío de bajada
  downlink: %{
    enabled: true,
    ip_hdr_incl: true,
    prefer_gre: true
  }
```

Parámetros GTP

Parámetro	Tipo	Requerido	Predeterminado	Descripción
`enabled`	Booleano	No	`false`	Iniciar los servidores GTP-C y GTP-U. Establecer `true` para modo en túnel.
`pgw_ip`	Cadena	No	`"10.5.198.1"`	La dirección IP del PGW para GTP-C y GTP-U.
`pgw_port`	Entero	No	`2123`	El puerto UDP GTP-C del PGW.
`local_ip`	Cadena	No	Detectado automáticamente	La IP local del TWAG para GTP. Utilizada como origen para los paquetes GTP-U y para la dirección del par de Indicación de Error.
`gtpu_port`	Entero	No	`2152`	El puerto UDP local

Parámetro	Tipo	Requerido	Predeterminado	Descripción
				para el servidor GTP-U. Este es el puerto GTP-U estándar.
`echo_enabled`	Booleano	No	`true`	Ejecutar el eco GTP-C hacia el PGW. Consulte S2a / GTP-C.

Parámetros GRE

Parámetro	Tipo	Requerido	Predeterminado	Descripción
`enabled`	Booleano	No	`false`	Iniciar el servidor GRE. Establecer `true` para recibir tramas UE del punto de acceso a través de GRE. Necesario para ambas salidas `tunneled` y `nswo`.
`local_ip`	Cadena	No	`"0.0.0.0"`	La IP local para vincular el socket en bruto GRE. Esta es la IP de destino a la que el punto de acceso envía el tráfico GRE.

Parámetros de Bajada

Parámetro	Tipo	Requerido	Predeterminado	Descripción
`enabled`	Booleano	No	`true`	Iniciar el reenvío de bajada. Cuando `false`, el reenvío descarta todos los paquetes de bajada.
`ip_hdr_incl`	Booleano	No	`true`	Establecer las opciones `IP_HDRINCL` y `IPV6_HDRINCL` en los sockets en bruto para que el reenvío proporcione e encabezado IP completo. Cuando `false`, la opción no se establece. Dejar `true` a menos que el kernel rechace `IP_HDRINCL`.
`prefer_gre`	Booleano	No	`true`	Orden de selección de

Parámetro	Tipo	Requerido	Predeterminado	Descripción
				ruta. Cuando `true`, el reenvío intenta GRE primero y recurre al socket en bruto. Cuando `false`, intenta el socket en bruto primero y utiliza GRE solo cuando el socket en bruto no está disponible.

Capacidad del Host

El servidor GRE y la ruta de socket en bruto de bajada utilizan sockets en bruto. El host debe otorgar la capacidad `CAP_NET_RAW`. Sin esta capacidad, los sockets en bruto no se abren. El servidor GRE vuelve a intentar el socket cada 10 segundos. El reenvío de bajada recurre o descarta paquetes, como se describe en Reenvío de Bajada.

Observabilidad

Los procesos del plano de usuario mantienen estadísticas internas. Cada proceso informa su propio estado, cubriendo el punto final del túnel GTP-U, el servidor GRE y el reenvío de bajada. Para las métricas de Prometheus que OmniTWAG exporta, consulte la Referencia de Métricas.

Estado GTP-U

El estado GTP-U informa `socket_open`, `local_ip`, `tunnel_count`, y un mapa `stats` anidado:

Estadística	Descripción
`packets_in`	El conteo de paquetes recibidos en el socket GTP-U.
`packets_out`	El conteo de paquetes enviados en el socket GTP-U.
`bytes_in`	El total de bytes recibidos.
`bytes_out`	El total de bytes enviados.
`errors`	El conteo de errores de decodificación y eventos de túnel desconocido.
`tunnel_count`	El número de túneles registrados (campo de nivel superior).

Estado GRE

El estado GRE informa `running`, `socket_open`, `local_ip`, `ap_tunnels`, `registered_ues`, y un mapa `stats` anidado:

Estadística	Descripción
`packets_in`	El conteo de paquetes GRE recibidos.
`packets_out`	El conteo de paquetes GRE enviados.
`bytes_in` / `bytes_out`	El total de bytes entrantes y salientes.
`dhcp_in`	El conteo de paquetes DHCP recibidos del punto de acceso.
`uplink_forwarded`	El conteo de paquetes de subida reenviados a GTP-U.
`downlink_forwarded`	El conteo de paquetes de bajada enviados al punto de acceso.
`unknown_protocol`	El conteo de paquetes con un protocolo o etype desconocido.
`errors`	El conteo de errores de decodificación y envío.
`registered_ues`	El número de UEs registrados (campo de nivel superior).

Estadísticas de Bajada

Estadística	Descripción
`forwarded`	El conteo de paquetes IPv4 reenviados (GRE y socket en bruto IPv4).
`forwarded_v6`	El conteo de paquetes IPv6 reenviados.
`dropped`	El conteo de paquetes descartados.
`errors`	El conteo de errores de envío.
`bytes`	El total de bytes reenviados.
`enabled`	El estado habilitado del reenvío.
`socket_open`	El estado del socket en bruto IPv4.

Descripción general