Guia de Operações do OmniPGW

OmniPGW - Plano de Controle do Gateway de Pacote (PGW-C)

por Omnitouch Network Services

Índice

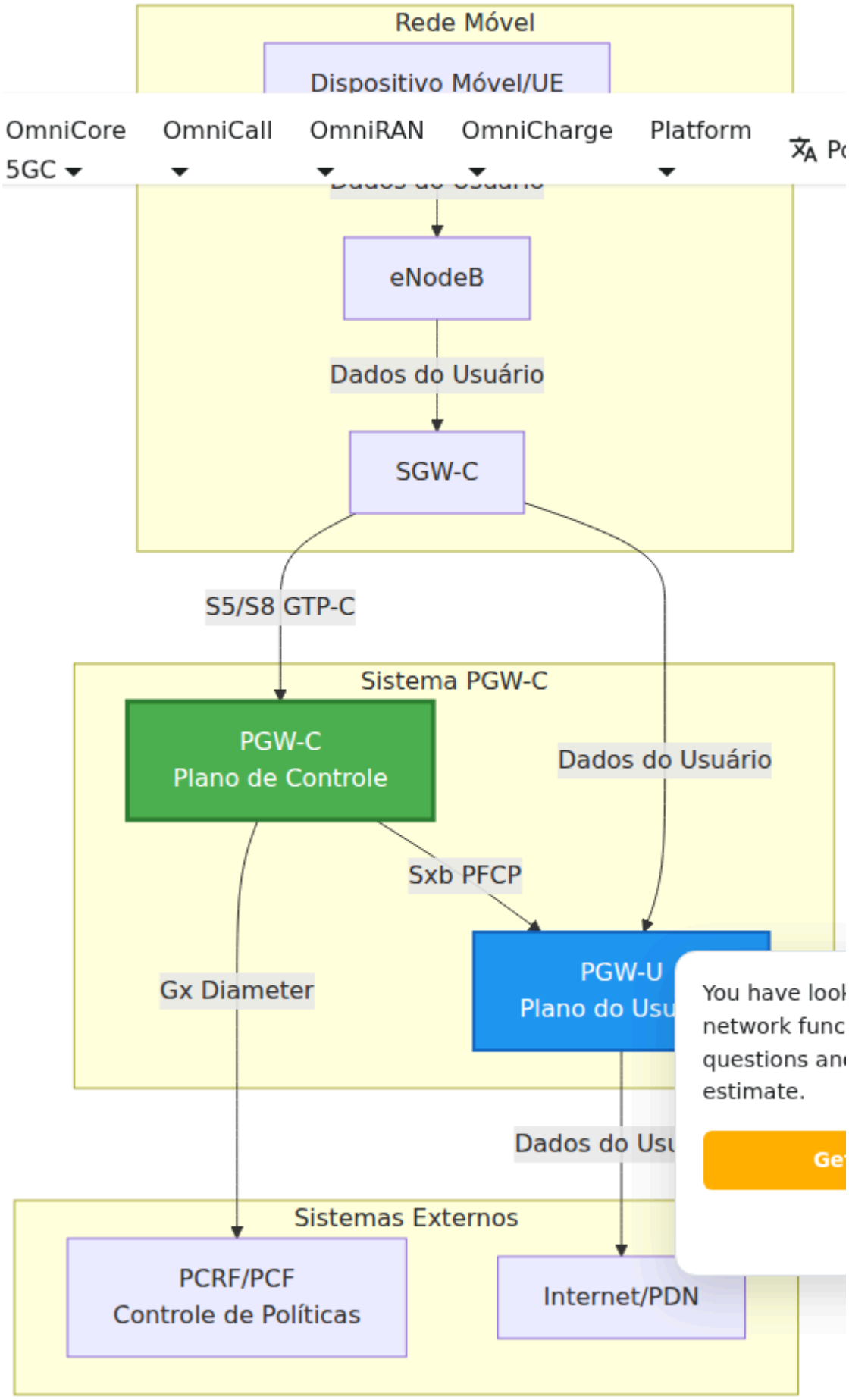
1. Visão Geral
2. Arquitetura
3. Interfaces de Rede
4. Conceitos Chave
5. Introdução
6. Configuração
7. Monitoramento de Operações em Tempo Real (API OAM)
8. Monitoramento & Métricas
9. Documentação Detalhada
10. Recursos Adicionais

Visão Geral

OmniPGW é uma implementação de alto desempenho do Plano de Controle do Gateway de Pacote (PGW-C) para redes 3GPP LTE Evolved Packet Core (EPC), desenvolvida pela Omnitouch Network Services. Ele gerencia as funções do plano de controle para sessões de dados, incluindo:

- **Gerenciamento de Sessão** - Criar, modificar e encerrar sessões de dados do UE (Equipamento do Usuário)

- **Alocação de Endereço IP** - Atribuindo endereços IP a dispositivos móveis a partir de pools configurados
- **Controle de Políticas e Cobrança** - Interagindo com o PCRF para aplicação de políticas e cobrança
- **Coordenação do Plano do Usuário** - Controlando o PGW-U (Plano do Usuário) para encaminhamento de pacotes


Rede Móvel
Dispositivo Móvel/UE
OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
eNodeB
Dados do Usuário
SGW-C
S5/S8 GTP-C
Sistema PGW-C
PGW-C
Plano de Controle
Dados do Usuário
Sxb PFCP
PGW-U
Gx Diameter
Sistemas Externos
PCRF/PCF
Controle de Políticas
Internet/PDN

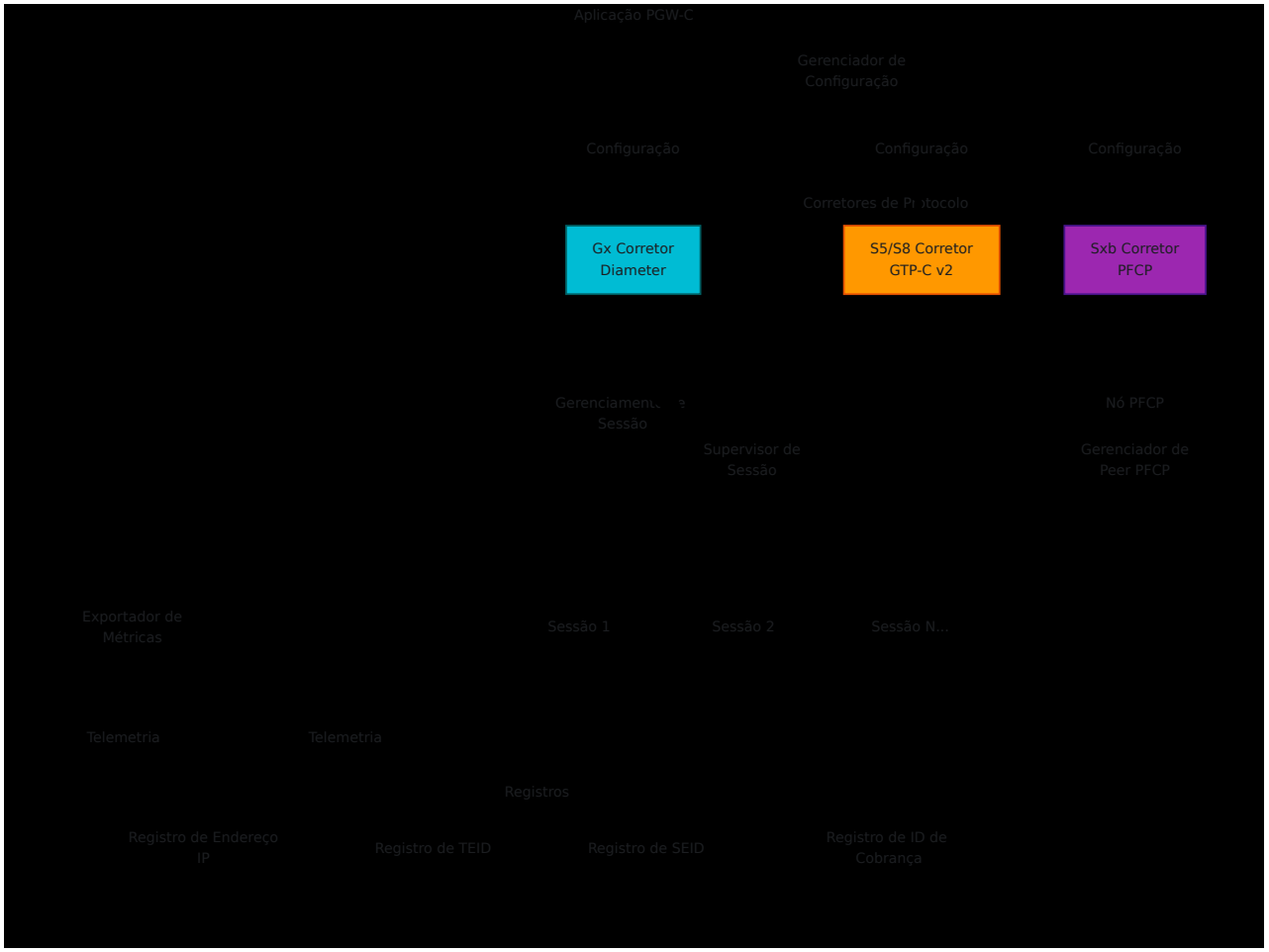
O que o PGW-C Faz

- **Aceita solicitações de sessão** do SGW-C via interface S5/S8 (GTP-C)
- **Aloca endereços IP do UE** a partir de pools de sub-rede configurados
- **Solicita decisões de políticas** do PCRF via interface Gx (Diameter)
- **Programa regras de encaminhamento** no PGW-U via interface Sxb (PFCP)
- **Gerencia a aplicação de QoS** através de contextos de portadora e regras de QoS
- **Rastreia informações de cobrança** para sistemas de faturamento

---

Arquitetura

Visão Geral dos Componentes



Arquitetura de Processos

O PGW-C é construído sobre Elixir/OTP e utiliza uma arquitetura de processos supervisionados:

- **Supervisor da Aplicação** - Supervisor de nível superior gerenciando todos os componentes
- **Corretores de Protocolo** - Lidam com mensagens de protocolo de entrada/saída
- **Processos de Sessão** - Um GenServer por conexão PDN ativa
- **Registros** - Rastreiam recursos alocados (IPs, TEIDs, SEIDs, etc.)
- **Gerenciador de Nó PFCP** - Mantém associações PFCP com peers PGW-U

Cada componente é supervisionado e será reiniciado automaticamente em caso de falha, garantindo a confiabilidade do sistema.

Interfaces de Rede

O PGW-C implementa três interfaces principais 3GPP:

Interface S5/S8 (GTP-C v2)

Propósito: Sinalização do plano de controle entre SGW-C e PGW-C

Protocolo: GTP-C Versão 2 sobre UDP

Mensagens Chave:

- Solicitação/Resposta de Criação de Sessão
- Solicitação/Resposta de Exclusão de Sessão
- Solicitação/Resposta de Criação de Portadora
- Solicitação/Resposta de Exclusão de Portadora

Configuração: Veja Configuração S5/S8

Interface Sxb (PFCP)

Propósito: Sinalização do plano de controle entre PGW-C e PGW-U

Protocolo: PFCP (Protocolo de Controle de Encaminhamento de Pacotes) sobre UDP

Mensagens Chave:

- Solicitação/Resposta de Configuração de Associação
- Solicitação/Resposta de Estabelecimento de Sessão
- Solicitação/Resposta de Modificação de Sessão
- Solicitação/Resposta de Exclusão de Sessão
- Solicitação/Resposta de Heartbeat

Configuração: Veja Documentação da Interface PFCP/Sxb

Interface Gx (Diameter)

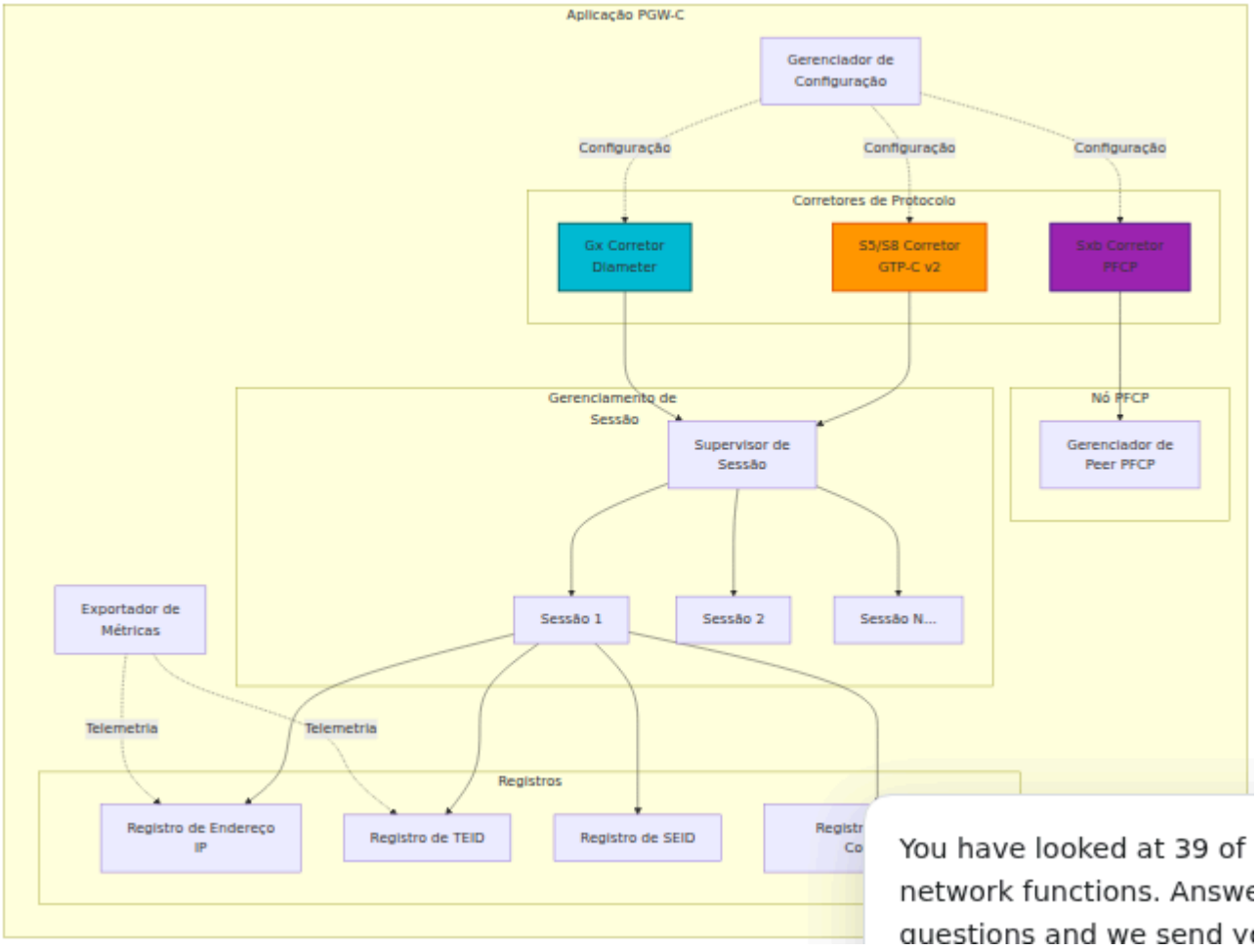
Propósito: Interface da Função de Regras de Políticas e Cobrança (PCRF)

Protocolo: Diameter (IETF RFC 6733)

Mensagens Chave:

- Solicitação/Resposta de Controle de Crédito Inicial (CCR-I/CCA-I)
- Solicitação/Resposta de Controle de Crédito de Término (CCR-T/CCA-T)

Configuração: Veja Documentação da Interface Diameter Gx



Conceitos Chave

Sessão PDN

Uma Sessão PDN (Rede de Dados de Pacote) representa a conexão de dados de um UE a uma rede externa (como a Internet). Cada sessão tem:

- **Endereço IP do UE** - Alocado a partir de um pool de sub-rede configurado
- **APN** (Nome do Ponto de Acesso) - Identifica a rede externa
- **Contexto de Portadora** - Contém parâmetros de QoS e informações de túnel
- **ID de Cobrança** - Identificador único para faturamento
- **TEID** (ID de Ponto de Extremidade do Túnel) - Identificador de túnel da interface S5/S8

- **SEID** (ID de Ponto de Extremidade da Sessão) - Identificador de sessão da interface Sxb

Contexto de Portadora

Uma portadora representa um fluxo de tráfego com características específicas de QoS:

- **Portadora Padrão** - Criada com cada sessão PDN
- **Portadoras Dedicadas** - Portadoras adicionais para necessidades específicas de QoS
- **EBI** (ID de Portadora EPS) - Identificador único para cada portadora
- **Parâmetros de QoS** - QCI, ARP, taxas de bits (MBR, GBR)

Regras PFCP

O PGW-C programa o PGW-U com regras de processamento de pacotes:

- **PDR** (Regra de Detecção de Pacotes) - Combina pacotes (uplink/downlink)
- **FAR** (Regra de Ação de Encaminhamento) - Especifica o comportamento de encaminhamento
- **QER** (Regra de Aplicação de QoS) - Aplica limites de taxa de bits
- **BAR** (Regra de Ação de Bufferização) - Controla a bufferização de pacotes

Veja Documentação da Interface PFCP para detalhes.

Alocação de Endereço IP

Os endereços IP do UE são alocados a partir de pools de sub-rede configurados:

- **Seleção baseada em APN** - Diferentes APNs podem usar sub-redes diferentes
- **Alocação dinâmica** - Seleção aleatória de IPs do intervalo disponível
- **Alocação estática** - Suporte para endereços IP solicitados pelo UE
- **Detecção de colisão** - Garante a atribuição única de IP

Veja Alocação de Pool de IP do UE para configuração, e IPv6 / Dual-Stack para operação PDN IPv6 e IPv4v6.

Introdução

Pré-requisitos

- Elixir ~1.16
- Erlang/OTP 26+
- Conectividade de rede com SGW-C, PGW-U e PCRF
- Compreensão da arquitetura EPC LTE

Iniciando o OmniPGW

1. **Configure as definições de tempo de execução** em `config/runtime.exs`
2. **Compile a aplicação:**

```
mix deps.get
mix compile
```

3. **Inicie a aplicação:**

```
mix run --no-halt
```

Verificando a Operação

Verifique os logs para um início bem-sucedido:

```
[info] Iniciando OmniPGW...
[info] Iniciando Exportador de Métricas em 127.0.0.42:42069
[info] Iniciando Corretor S5/S8 em 127.0.0.10
[info] Iniciando Corretor Sxb em 127.0.0.20
[info] Iniciando Corretor Gx
[info] Iniciando Gerenciador de Nó PFCP
[info] OmniPGW iniciado com sucesso
```

Acesse as métricas em `http://127.0.0.42:42069/metrics` (endereço configurado).

Configuração

Toda a configuração em tempo de execução é definida em `config/runtime.exs`. Este arquivo é avaliado na inicialização da aplicação. Um exemplo completo segue; cada seção é descrita na tabela de referência abaixo e na Referência de Configuração.

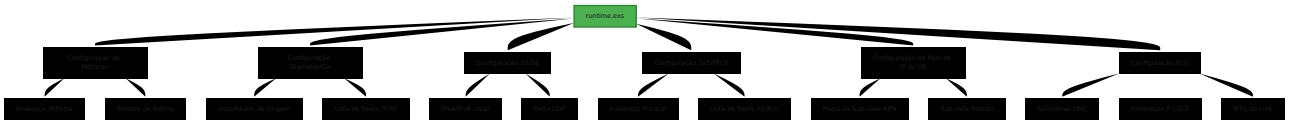
```
# config/runtime.exs
import Config

# Configuração do Logger
config :logger, level: :info

config :pgw_c,
  # Métricas (Prometheus)
  metrics: %{
    enabled: true,
    ip_address: "10.0.0.20",  # Rede de gerenciamento
    port: 9090,
    registry_poll_period_ms: 5_000
  },

  # Diameter/Gx (interface PCRF)
  diameter: %{
    listen_ip: "0.0.0.0",
    host: "omnipgw.epc.mnc001.mcc001.3gppnetwork.org",
    realm: "epc.mnc001.mcc001.3gppnetwork.org",
    peer_list: [
      %{
        host: "pcrf-primary.epc.mnc001.mcc001.3gppnetwork.org",
        realm: "epc.mnc001.mcc001.3gppnetwork.org",
        ip: "10.0.1.30",
        initiate_connection: true
      },
      %{
        host: "pcrf-backup.epc.mnc001.mcc001.3gppnetwork.org",
        realm: "epc.mnc001.mcc001.3gppnetwork.org",
        ip: "10.0.2.30",
        initiate_connection: true
      }
    ]
  },

  # S5/S8 (interface SGW-C)
  s5s8: %{
    local_ipv4_address: "10.0.0.20"
  },

  # Sxb/PFCP (interface PGW-U)
  sxb: %{
```

```
    local_ip_address: "10.0.0.20"
  },

  # Seleção de UPF
  upf_selection: %{
    rules: [
      %{
        name: "Internet Padrão",
        priority: 1,
        match_field: :apn,
        match_regex: ~r/^internet/,
        upf_pool: [
          %{remote_ip_address: "10.0.0.21", remote_port: 8805,
weight: 100},
          %{remote_ip_address: "10.0.0.22", remote_port: 8805,
weight: 0}  # Standby
        ]
      }
    ],
    fallback_pool: [
      %{remote_ip_address: "10.0.0.21", remote_port: 8805, weight:
100}
    ]
  },

  # Pools de IP do UE
  ue: %{
    subnet_map: %{
      "internet" => [
        "100.64.0.0/18"  # 16.382 IPs
      ],
      "ims" => [
        "100.64.64.0/22"  # 1.022 IPs
      ],
      "enterprise.corp" => [
        "10.100.0.0/16"  # 65.534 IPs
      ],
      default: [
        "100.64.127.0/24"  # 254 IPs
      ]
    }
  },

  # Opções de Configuração de Protocolo
```

```
  pco: %{
    primary_dns_server_address: "8.8.8.8",
    secondary_dns_server_address: "8.8.4.4",
    p_cscf_ipv4_address_list: ["10.0.0.50", "10.0.0.51"],
    ipv4_link_mtu_size: 1400
  }
```

A configuração é estruturada em várias seções:

Visão Geral da Configuração



Referência Rápida de Configuração

Seção	Propósito	Documentação
metrics	Exportador de métricas Prometheus	Guia de Monitoramento
diameter	Interface Gx para PCRF	Configuração Diameter Gx
s5s8	Interface GTP-C para SGW-C	Configuração S5/S8
sxb	Interface PFCP para PGW-U	Configuração PFCP
ue	Pools de endereços IP do UE	Configuração de Pool de IP
pco	Opções de Configuração de Protocolo	Configuração PCO
CDR	Cobrança offline e relatórios de uso	Formato CDR
charging	Modo de cobrança online/offline, sobreposição de quebra de vidro	Sobrescrição de Cobrança

Veja o Guia Completo de Configuração para informações detalhadas.

Monitoramento de Operações em Tempo Real (API OAM)

O OmniPGW expõe uma **API OAM REST** para monitoramento e operações em tempo real, fornecendo visibilidade instantânea sobre o status do sistema. Consulte esses endpoints de leitura sob demanda (ou em um intervalo de polling / a partir de um script) para uma visão operacional ao vivo - sem necessidade de acesso à linha de comando ao host ou consultas de métricas brutas.

Acessando a API OAM

A API é servida sobre HTTPS/TLS. Todas as rotas são servidas sob o prefixo `/api`.

```
https://<omnipgw-ip>:8443/
```

A documentação interativa da API (Swagger UI) está disponível em:

```
https://<omnipgw-ip>:8443/api/docs
```

Exemplo de verificação de saúde (use `-k` para um certificado de laboratório autoassinado):

```
curl -k https://localhost:8443/api/status
{"result":"ok"}
```

Endpoints Disponíveis:

Propósito	Endpoint	Substitui (página antiga)
Mergulho profundo em um assinante específico (IMSI/MSISDN/IP)	`POST /api/ue_search`	Pesquisa de UE
Listar / pesquisar sessões PDN ativas	`GET /api/sessions[?search=<imsi|msisdn|ip>]`	Sessões PGW
Uma sessão por IMSI	`GET /api/sessions/<imsi>`	Sessões PGW (detalhe)
Histórico de sessões / eventos de auditoria	`GET /api/session_history[?type=&search=&page=&page_size=]`	Histórico de Sessão
Visão geral da topologia da rede	`GET /api/topology`	Topologia da Rede
Utilização do pool de endereços IP do UE	`GET /api/ip_pools`, `GET /api/ip_pools/<name>`	Pools de IP
Status do UPF / peer PFCP	`GET /api/upf`, `GET /api/upf/<ip>`	Sessões PFCP / Status do UPF
Regras de seleção de UPF, saúde, configuração PCO	`GET /api/upf_selection`	Seleção de UPF

Propósito	Endpoint	Substitui (página antiga)
Conectividade de peers Diameter (Gx/Gy)	`GET /api/diameter[?status=connected|disconnected]`, `GET /api/diameter/<origin-host>`	Peers Diameter
Status de cobrança online (Gy) por sessão	`GET /api/charging[?search=<imsi>]`, `GET /api/charging/<imsi>`	Status Gy
Quebra de vidro: desativar cobrança online em toda a rede	`PUT /api/charging/override`, `GET /api/charging/override`, `DELETE /api/charging/override`	-
Status de descoberta de DNS P-CSCF	`GET /api/pcscf_monitor`	Monitor P-CSCF
Verificação de saúde	`GET /api/status`	-

Para uma visão estilo dashboard ao vivo, faça polling nos endpoints de coleção em um intervalo, por exemplo:

```
watch -n 2 'curl -sk https://localhost:8443/api/sessions'
watch -n 1 'curl -sk https://localhost:8443/api/diameter?status=disconnected'
```

> Nota: a antiga página **Logs** (streaming de logs ao vivo) e a página **Gy Simulator** não têm equivalente na API OAM. Transmita logs do host / seu agregador de logs e use `GET /api/charging` para inspecionar o estado ao vivo de Gy/cobrança online.

Recursos Chave

Dados Sob Demanda e Polling:

- Cada endpoint retorna o estado atual ao vivo dos processos do OmniPGW
- Faça polling em um intervalo com `watch`/cron/um script para uma visão que se atualiza automaticamente
- Campos de status (por exemplo, peer `connected`/`associated`) são retornados explicitamente no JSON

Pesquisa & Filtro:

- Pesquise sessões por IMSI, IP, MSISDN ou APN via `GET /api/sessions?search=...`
- Filtre peers Diameter por `?status=connected|disconnected`
- Filtre o estado de cobrança por `?search=<imsi>`

Registros de Detalhes Completos:

- Endpoints por entidade (`/api/sessions/<imsi>`, `/api/upf/<ip>`, `/api/diameter/<origin-host>`, `/api/charging/<imsi>`) retornam estado completo como JSON
- Inspecione a sessão completa, configuração de peer e capacidades

Controle de Acesso:

- Servido sobre HTTPS/TLS na porta 8443
- Vincule apenas ao IP de gerenciamento e utilize seus controles de acesso
- Destinado à equipe de NOC/operações e uso de automação

Fluxos de Trabalho Operacionais

Solução de Problemas de Sessão (Mergulho Profundo):

```
1. Usuário relata problema de conexão
2. POST /api/ue_search com um corpo JSON contendo o IMSI, MSISDN
ou IP
3. Revise os detalhes abrangentes da sessão na resposta:
   a) Sessões Ativas - Verifique se a sessão existe com os
parâmetros corretos
   b) Localização Atual - Verifique TAC, ID da Célula, rede
servidora
   c) Informações da Portadora - Verifique portadoras padrão e
dedicadas
      - QCI, MBR/GBR, Nomes de Regras de Cobrança
      - Limites APN-AMBR
   d) Informações de Cobrança - ID da sessão Gy, status de cota
(também GET /api/charging/<imsi>)
   e) Informações de Política - Sessão Gx, regras PCC instaladas
   f) Eventos Recentes - GET /api/session_history?search=<imsi>
4. Se a sessão não for encontrada → GET /api/diameter?
status=disconnected para conectividade PCRF
```

Pesquisa Rápida de Sessão:

```
1. Usuário relata problema
2. GET /api/sessions?search=<imsi|msisdn> (ou GET
/api/sessions/<imsi>)
3. Verifique se a sessão existe com detalhes básicos:
   - Endereço IP do UE alocado
   - Parâmetros de QoS
   - Pontos de extremidade de túnel estabelecidos
4. Para análise detalhada → POST /api/ue_search
```

Verificação de Saúde do Sistema:

```
1. GET /api/upf              → Verifique se todos os peers PGW-U
estão "Associados"
2. GET /api/diameter?status=connected → Verifique se todos os
peers PCRF estão "Conectados"
3. GET /api/sessions         → Verifique a contagem de sessões
ativas em relação à capacidade
4. GET /api/status           → Verificação de saúde geral
({"result":"ok"})
```

Monitoramento de Capacidade:

- Conte as entradas retornadas por `GET /api/sessions`
- Compare com a capacidade licenciada/esperada
- Identifique horários de pico de uso
- Monitore a distribuição entre APNs; use `GET /api/ip_pools` para utilização por pool

API vs. Métricas

Use a API OAM para:

- Solução de problemas de assinantes em profundidade (`POST /api/ue_search`)
- Detalhes e inspeção de estado de sessão individuais (`GET /api/sessions/<imsi>`)
- Status de peers em tempo real (`GET /api/upf`, `GET /api/diameter`)
- Verificações rápidas de saúde em todas as interfaces (`GET /api/status`)
- Solução de problemas de usuários específicos por IMSI/MSISDN/IP
- Análise de QoS de portadora (MBR, GBR, QCI)
- Inspeção de regras de política e cobrança (`GET /api/charging`)
- Histórico de sessões e trilhas de auditoria (`GET /api/session_history`)
- Monitoramento de capacidade de pool de IP (`GET /api/ip_pools`)
- Verificação de configuração e regras (`GET /api/upf_selection`)

Use Métricas Prometheus para:

- Tendências históricas
- Alertas e notificações
- Gráficos de planejamento de capacidade
- Análise de desempenho
- Monitoramento de longo prazo

Melhor Prática: Use ambos juntos - a API OAM para operações imediatas, Prometheus para tendências e alertas.

Monitoramento & Métricas

Além da API OAM, o OmniPGW expõe métricas compatíveis com Prometheus para monitoramento:

Métricas Disponíveis

- **Métricas de Sessão**
  - `teid_registry_count` - Sessões S5/S8 ativas
  - `seid_registry_count` - Sessões PFCP ativas
  - `session_id_registry_count` - Sessões Gx ativas
  - `address_registry_count` - Endereços IP do UE alocados
  - `charging_id_registry_count` - IDs de cobrança ativos
- **Métricas de Mensagens**
  - `s5s8_inbound_messages_total` - Mensagens GTP-C recebidas
  - `sxb_inbound_messages_total` - Mensagens PFCP recebidas
  - `gx_inbound_messages_total` - Mensagens Diameter recebidas
  - Distribuições da duração do manuseio de mensagens
- **Métricas de Erros**
  - `s5s8_inbound_errors_total` - Erros de protocolo S5/S8
  - `sxb_inbound_errors_total` - Erros de protocolo PFCP

- `gx_inbound_errors_total` - Erros de Diameter

- **Métricas do Ciclo de Vida da Sessão**
  - `pgw_session_create_total{result,cause,pdn_type,rat_type}` - Tentativas de criação de sessão PDN, divididas por resultado (`success`/`failure`), a causa GTP retornada, tipo de PDN (`ipv4`/`ipv6`/`ipv4v6`) e tipo de RAT
  - `pgw_session_delete_total{pdn_type}` - Exclusões de sessão PDN por tipo de PDN
  - `pgw_session_modify_total` - Solicitações de Modificação de Bearer tratadas

- **Métricas de Pool de IP de UE**
  - `ue_pool_addresses_allocated{pool,ip_version}` - Endereços de UE alocados por pool (padrão APN) e versão de IP
  - `ue_pool_addresses_total{pool,ip_version}` - Capacidade total do pool
  - `ue_pool_addresses_utilization_ratio{pool,ip_version}` - Fração alocada (0.0-1.0); útil para alertas de exaustão do pool. Observe que pools que compartilham um intervalo de endereços contam cada uma das alocações compartilhadas
  - `ue_ip_allocation_failures_total{ip_version,reason}` - Falhas de alocação (`pool_exhausted`, `already_registered`)

- **Métricas de UPF / PFCP por Par**
  - `upf_peer_associated{peer_ip}` - Estado de associação PFCP 1/0 por par
  - `upf_peer_healthy{peer_ip}` - Saúde 1/0 por par
  - `upf_peer_missed_heartbeats{peer_ip}` - Batimentos cardíacos consecutivos perdidos por par
  - `upf_heartbeat_rtt` - Distribuição do tempo de ida e volta do batimento PFCP, por `peer_ip`

- **Métricas de Bearer & QoS**

    - `pgw_bearer_create_total{type,qci}` - Bearers criados por tipo (`default`/`dedicated`) e QCI
    - `pgw_bearer_delete_total{type}` - Bearers excluídos por tipo

- **Métricas de Par Diameter**

    - `diameter_peers_connected{application}` - Pares Diameter conectados por aplicação (`gx`/`gy`/`all`)

Acessando Métricas

As métricas são expostas via HTTP no endpoint configurado:

```
curl http://127.0.0.42:42069/metrics
```

Veja Guia de Monitoramento & Métricas para configuração de painel e alertas.

Documentação Detalhada

Esta seção fornece uma visão abrangente de toda a documentação do OmniPGW. Os documentos estão organizados por tópico e caso de uso.

Estrutura da Documentação

```
Documentação OmniPGW
├── OPERATIONS.md (Este Guia)
│
└── docs/
    ├── Configuração & Configuração
    │   ├── configuration.md          Referência completa do
runtime.exs
    │   ├── ue-ip-allocation.md       Configuração do pool de IP
    │   └── pco-configuration.md      Configurações de DNS, P-
CSCF, MTU
    │
    ├── Interfaces de Rede
    │   ├── pfcp-interface.md         Sxb/PFCP (comunicação PGW-U)
    │   ├── diameter-gx.md            Gx (comunicação PCRF)
    │   ├── diameter-gy.md            Gy/Ro (comunicação OCS)
    │   └── s5s8-interface.md         S5/S8 (comunicação SGW-C)
    │
    └── Operações
        ├── session-management.md     Ciclo de vida da sessão PDN
        ├── charging-override.md      Sobrescrição de cobrança
online
        └── monitoring.md             Métricas Prometheus &
alertas
```

Documentação por Tópico

🚀 Começando

Documento	Descrição	Propósito
OPERATIONS.md	Guia principal de operações (este documento)	Visão geral e início rápido

⚙️ Configuração

Documento	Descrição	Linhas
configuration.md	Referência completa da configuração runtime.exs	1,600+
ue-ip-allocation.md	Gerenciamento e alocação do pool de IP de UE	943
ipv6-dual-stack.md	Operação PDN IPv6 / IPv4v6 entre PGW-C e UPF	-
pco-configuration.md	Opções de Configuração de Protocolo (DNS, P-CSCF, MTU)	344

Interfaces de Rede

Documento	Descrição	Linhas
pfcp-interface.md	Interface PFCP/Sxb para PGW-U	1,355
diameter-gx.md	Interface Diameter Gx para PCRF (Controle de Política)	941
diameter-gy.md	Interface Diameter Gy/Ro para OCS (Cobrança Online)	1,100+
charging-override.md	Sobrescrição de quebra de vidro para desabilitar cobrança online durante um incidente OCS	-
s5s8-interface.md	Interface GTP-C S5/S8 para SGW-C	456

Operações & Monitoramento

Documento	Descrição	Linhas
session-management.md	Ciclo de vida e operações da sessão PDN	435
monitoring.md	Métricas Prometheus, painéis Grafana, alertas	807
data-cdr-format.md	Formato de arquivo CDR, configuração URR, cobrança offline	847
qos-bearers.md	Gerenciamento de QoS & bearer, controle de políticas	448
troubleshooting.md	Procedimentos de solução de problemas e questões comuns	687

Recursos Avançados

Documento	Descrição	Linhas
pcscf-monitoring.md	Descoberta e monitoramento de saúde do P-CSCF	894

Recursos da Documentação

Diagramas Mermaid

Todos os documentos incluem **gráficos Mermaid** para compreensão visual:

- Diagramas de arquitetura
- Diagramas de sequência (fluxos de mensagens)
- Máquinas de estado
- Topologia de rede

Exemplos Práticos

Cada documento inclui:

- Exemplos de configuração do mundo real
- Configurações prontas para copiar e colar
- Casos de uso comuns

🔧 Solução de Problemas

Cada documento de interface inclui:

- Questões comuns e soluções
- Comandos de depuração
- Métricas para diagnóstico

🔗 Referências Cruzadas

Os documentos são amplamente interligados para fácil navegação.

Caminhos de Leitura

Para Operadores de Rede

1. OPERATIONS.md - Visão geral (este documento)
2. configuration.md - Configuração
3. monitoring.md - Monitoramento
4. session-management.md - Operações do dia a dia

Para Engenheiros de Rede

1. OPERATIONS.md - Visão geral da arquitetura (este documento)
2. pfcp-interface.md - Controle do plano de usuário
3. diameter-gx.md - Controle de políticas
4. diameter-gy.md - Cobrança online
5. s5s8-interface.md - Gerenciamento de sessão
6. ue-ip-allocation.md - Gerenciamento de IP

Para Configuração & Implantação

1. configuration.md - Referência completa
2. ue-ip-allocation.md - Pools de IP
3. pco-configuration.md - Parâmetros de rede
4. monitoring.md - Configurar monitoramento

Estatísticas do Documento

- **Total de Documentos:** 14
- **Total de Linhas:** ~10,900+
- **Tamanho Total:** ~265 KB
- **Diagramas Mermaid:** 75+
- **Exemplos de Código:** 150+

Conceitos Chave Abordados

Arquitetura

- □ Separação do plano de controle/plano de usuário
- □ Arquitetura OTP/Elixir
- □ Supervisão de processos
- □ Sessões baseadas em GenServer

Protocolos

- □ PFCP (Protocolo de Controle de Encaminhamento de Pacotes)
- □ GTP-C v2 (Protocolo de Tunelamento GPRS)
- □ Diameter (RFC 6733)

Interfaces 3GPP

- □ Sxb (PGW-C ↔ PGW-U)
- □ Gx (PGW-C ↔ PCRF)
- □ Gy/Ro (PGW-C ↔ OCS)
- □ S5/S8 (SGW-C ↔ PGW-C)

Operações

- □ Gerenciamento de sessão
- □ Estratégias de alocação de IP
- □ Aplicação de QoS
- □ Integração de cobrança
- □ Monitoramento & alertas

Recursos Adicionais

Especificações 3GPP

Especificação	Título
TS 29.274	GTP-C v2 (interface S5/S8)
TS 29.244	PFCP (interface Sxb)
TS 29.212	Interface Diameter Gx (Controle de Política)
TS 32.299	Aplicações de Cobrança Diameter (Gy/Ro)
TS 32.251	Cobrança de domínio comutado por pacotes
TS 23.401	Arquitetura EPC

Documentação Relacionada

- Referência de configuração: configuration.md - Referência completa do `config/runtime.exs`

Sobrescrição de Cobrança Online (Break-Glass)

A sobrescrição de cobrança online é um interruptor a nível de nó. Ela retira a cobrança online (Gy/OCS) do caminho imediatamente, sem reinicialização. Use-a durante um incidente de OCS, quando o OCS está indisponível e os assinantes estão afetados.

Uma reinicialização não é uma maneira aceitável de desativar a cobrança online. Uma reinicialização encerra todas as sessões PDN ativas no nó. A sobrescrição altera o comportamento para sessões que já estão ativas e para sessões que se conectam enquanto está ativa.

A cobrança offline não é afetada. O PGW-C continua a medir o uso e a escrever CDRs enquanto a sobrescrição está ativa. Veja Formato de CDR.

Índice

1. Quando Usar
2. O que a Sobrescrição Faz
3. Referência da API
4. Procedimento Operacional
5. Registro de Auditoria
6. Configuração
7. Métricas
8. Solução de Problemas

Quando Usar

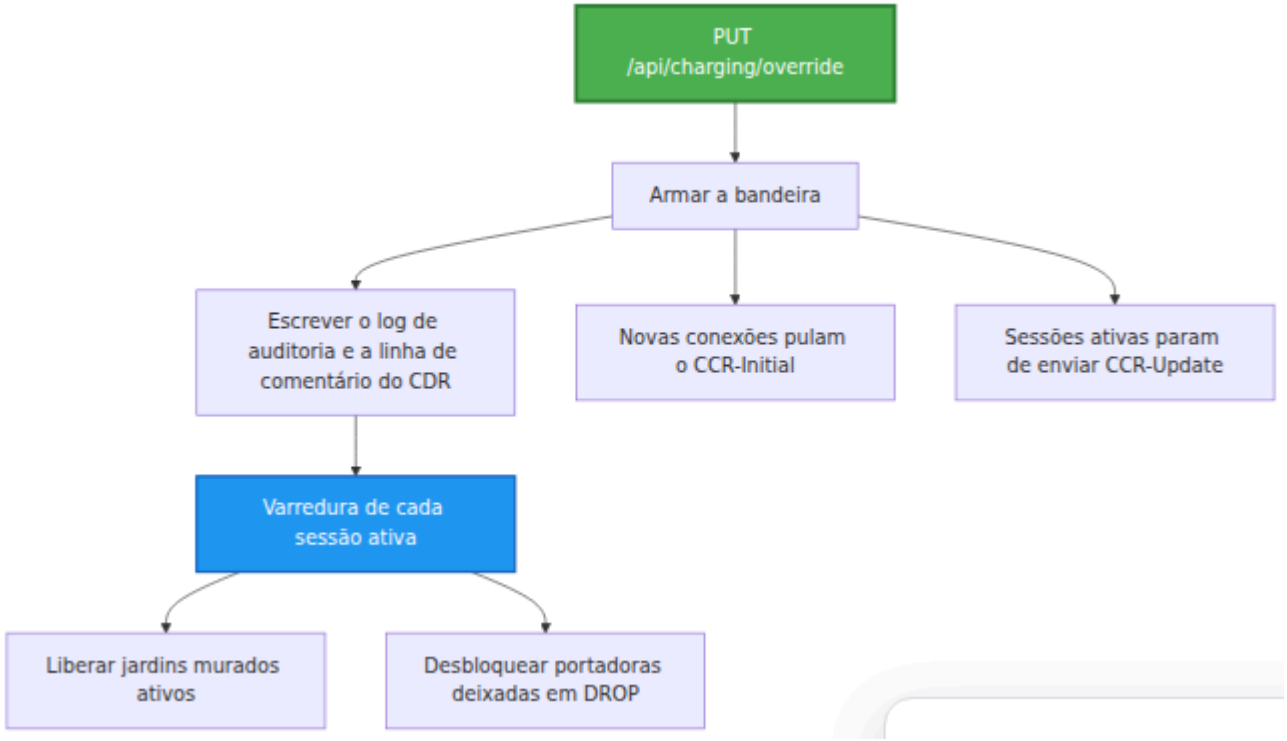
Use a sobrescrição quando o OCS causar impacto nos assinantes que você não consegue corrigir no OCS. Três modos de falha são comuns.

Modo de falha	O que os assinantes veem	O que a sobrescrição faz
O OCS rejeita ou não responde ao CCR-Initial	Novas conexões entram restritas ou vão para o jardim murado	Novas conexões não enviam CCR-Initial e entram com encaminhamento de tráfego
O OCS responde, mas sua contabilidade está errada	Assinantes são desconectados com saldo restante	Jardins murados são liberados e portadoras bloqueadas são desbloqueadas
O OCS deixa reservas penduradas e esgota pacotes	O saldo desaparece sem uso	Nenhum CCR-Update adicional é enviado, então a janela não é reportada e não é tarifada

Não use a sobrescrição para um único assinante. Para mover uma sessão para dentro ou fora do jardim murado, use `PUT /api/charging/<imsi>`. Veja Interface Diameter Gy.

O escopo da sobrescrição é um nó. Se você executar mais de um PGW-C, aplique a sobrescrição a cada nó. Uma aplicação parcial é sobrevivível. Alguns assinantes se recuperam, e o restante permanece como estava.

O que a Sobrescrição Faz



Novas sessões

Enquanto a sobrescrição está ativa, o PGW-C trata a cobrança online como desativada. Este é o mesmo comportamento que o valor de configuração `charging.online: :off`. O PGW-C não envia CCR-Initial. A conexão é concluída e a portadora padrão encaminha tráfego.

A sobrescrição tem prioridade sobre a configuração local e o PCRF. Um Gx CCA que define `Online` como habilitado não reabilita a cobrança enquanto a sobrescrição está ativa.

Sessões existentes

Uma bandeira de configuração sozinha ajuda apenas as sessões que se conectam após ela. Os assinantes que já estão bloqueados são os que o incidente está prejudicando. Quando você arma a sobrescrição, o PGW-C varre cada sessão ativa e faz duas coisas.

1. **Liberar o jardim murado.** Se a sessão está em um jardim murado, o PGW-C envia uma Modificação de Sessão PFCP que remove o

redirecionamento. Este é o mesmo caminho que é executado quando o OCS restaura o crédito.

2. **Desbloquear portadoras bloqueadas.** Se uma portadora tem Regras de Ação de Encaminhamento definidas para aplicar a ação `DROP`, o PGW-C envia uma Modificação de Sessão PFCP que as redefine para `FORW`. Isso cobre as sessões bloqueadas pelo ramo CCR-Initial `DIAMETER_USER_UNKNOWN`.

A varredura é um esforço de melhor tentativa. Uma sessão que não responde é contada como falha e nomeada no log. A varredura continua com as sessões restantes. O PGW-C executa a varredura em paralelo nas sessões, então o tempo que leva é determinado pela resposta mais lenta do UPF, não pela soma delas.

Um caso não pode ser reparado. Se o OCS enviou uma Ação de Unidade Final de TERMINATE na conexão, o PGW-C limpou os parâmetros de encaminhamento nessas regras. O PGW-C não pode reconstruí-los a partir do estado da sessão. O log nomeia cada regra que não pôde ser restaurada. O assinante deve se reconectar para se recuperar.

Sessões que já têm uma sessão Gy aberta

Enquanto a sobrescrição está ativa, o PGW-C não envia CCR-Update para essas sessões. O relatório de uso é descartado em vez de armazenado em buffer. Isso é deliberado. A desativação não deve cobrar retroativamente a janela em que a sobrescrição estava ativa.

Na desconexão, o PGW-C não envia CCR-Terminate por padrão. Um CCR-Terminate libera a reserva no OCS de forma limpa, mas um por desconexão ataca um OCS que já está doente. O OCS colhe a reserva com seu próprio temporizador de sessão. Defina `send_ccr_t` como `true` se você quiser que o CCR-Terminate seja enviado de qualquer maneira.

Desativação

A desativação retorna o nó à cobrança normal imediatamente. Nenhuma reinicialização é necessária. O PGW-C não reporta a janela em que a

sobrescrição estava ativa. O uso nessa janela é medido offline e escrito em CDRs, mas nunca chega ao OCS.

Referência da API

A sobrescrição é um recurso singleton na coleção de cobrança. O identificador `override` é reservado. Nenhum IMSI pode ser essa string.

Todas as rotas são servidas sob o prefixo `/api` na API OAM. Veja Monitoramento de Operações em Tempo Real.

Ativar

```
PUT /api/charging/override
```

Corpo da solicitação:

```
{
  "ttl_seconds": 3600,
  "reason": "Incidente OCS INC-1234",
  "actor": "noc@example.com",
  "send_ccr_t": false
}
```

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
`ttl_seconds`	Inteiro	Sim	-	Segundos até que a sobrescrição expire automaticamente. Mínimo `1`, máximo `86400` (24 horas).
`reason`	String	Sim	-	Texto livre que diz por que a sobrescrição está ativa. Aparece no log, no arquivo CDR e no `GET`.
`actor`	String	Não	IP do chamador	Quem acionou a bandeira. O PGW-C registra o endereço IP do chamador quando você omite isso.
`send_ccr_t`	Booleano	Não	`false`	Defina como `true` para ainda enviar um Gy CCR-Terminate na desconexão. Deixe `false` para permanecer em silêncio.

Resposta:

```
{
  "online_charging": "off",
  "override": {
    "active": true,
    "reason": "Incidente OCS INC-1234",
    "actor": "noc@example.com",
    "ttl_seconds": 3600,
    "remaining_seconds": 3599,
    "activated_at": "2026-09-15T04:10:22.481Z",
    "expires_at": "2026-09-15T05:10:22.481Z",
    "send_ccr_t": false,
    "max_ttl_seconds": 86400
  },
  "relief": {
    "sessions_swept": 1842,
    "walled_gardens_released": 37,
    "bearers_ungated": 5,
    "failed": 0,
    "changed": [
      { "imsi": "313380000003530", "apn": "internet", "actions":
["walled_garden_released"] }
    ]
  }
}
```

Campo	Descrição
`online_charging`	`off` enquanto a sobrescrição está ativa, `on` caso contrário.
`override.active`	`true` enquanto a sobrescrição está armada e o TTL não expirou.
`override.remaining_seconds`	Segundos até a expiração automática. `null` quando nenhuma sobrescrição está ativa.
`relief.sessions_swept`	Quantas sessões ativas o PGW-C pediu para derrubar seus bloqueios de cobrança.
`relief.walled_gardens_released`	Quantas sessões deixaram o jardim murado.
`relief.bearers_ungated`	Quantas sessões tiveram portadoras bloqueadas restauradas para encaminhamento.
`relief.failed`	Quantas sessões não responderam. Seu estado permanece inalterado.
`relief.changed`	As sessões que realmente mudaram, com a ação tomada em cada uma.

O PGW-C rejeita a solicitação com `400 Bad Request` quando `ttl_seconds` está ausente, fora do intervalo ou não é um inteiro, e quando `reason` está ausente ou em branco.

Ler status

```
GET /api/charging/override
```

A resposta contém os mesmos campos `online_charging` e `override` que a resposta de ativação, com `remaining_seconds` contado a partir do momento da solicitação. Não há campo `relief`.

O endpoint de lista `GET /api/charging` também contém o status da sobrescrição em um campo `charging_override`, ao lado do estado de cobrança por sessão.

Desativar

```
DELETE /api/charging/override
```

Corpo da solicitação (opcional):

```
{ "actor": "oncall@example.com" }
```

A resposta contém os mesmos campos que `GET`, com `active` definido como `false`.

Procedimento Operacional

Siga estas etapas durante um incidente de OCS.

1. Confirme que o OCS é a causa. Verifique `GET /api/diameter?status=disconnected` para o estado do peer Gy, e `GET /api/charging` para sessões no jardim murado.
2. Arme a sobrescrição no primeiro PGW-C. Dê um `reason` que nomeie o ticket do incidente. Defina `ttl_seconds` para um valor que cubra seu tempo de reparo esperado.
3. Leia o bloco `relief` na resposta. Confirme que `failed` é `0`. Se não for, veja Solução de Problemas.
4. Repita as etapas 2 e 3 em cada nó PGW-C restante.
5. Confirme que a métrica `charging_override_active` lê `1` em cada nó.
6. Repare o OCS.

7. Desative a sobrescrição em cada nó com `DELETE /api/charging/override`.
8. Confirme que a métrica `charging_override_active` lê `0` em cada nó, e que novas conexões enviam um CCR-Initial novamente.

Se você não desativar a sobrescrição, ela expira automaticamente quando o TTL expira. O PGW-C escreve um aviso no log e uma linha de comentário no arquivo CDR na expiração.

Registro de Auditoria

Uma sobrescrição que pode ser deixada ativa silenciosamente é um vazamento de receita. O PGW-C registra cada mudança de estado em quatro lugares.

Log

O PGW-C escreve um `warning` na ativação, na desativação, na expiração do TTL e na re-armação após uma reinicialização. Cada linha é prefixada com `[CHARGING OVERRIDE]` e contém o motivo, o ator e o prazo. A varredura escreve um segundo `warning` com a contagem de sessões alteradas e falhadas.

Arquivo CDR

O PGW-C escreve uma linha de comentário no arquivo CDR que está aberto no momento. A escrita é assíncrona, então um diretório CDR em um mount lento não pode travar ou falhar a sobrescrição em si. As linhas de comentário começam com `#`, o mesmo marcador que o cabeçalho do arquivo CDR usa, então a mediação as ignora. A linha permanece no registro do arquivo, que é o que você precisa quando uma janela de tráfego foi deliberadamente não tarifada.

```
# SOBRESCRIÇÃO DE COBRANÇA ATIVADA em 2026-09-15T04:10:22.481Z:
cobrança online (Gy/OCS) desativada em todo o nó; uso nesta janela
é medido offline mas NÃO reportado ao OCS. motivo=Incidente OCS
INC-1234 ator=noc@example.com ttl_seconds=3600 expires_at=2026-09-
15T05:10:22.481Z send_ccr_t=false
# SOBRESCRIÇÃO DE COBRANÇA DESATIVADA em 2026-09-15T04:52:09.113Z:
cobrança online (Gy/OCS) reabilitada em todo o nó. uso entre 2026-
09-15T04:10:22.481Z e 2026-09-15T04:52:09.113Z (2507s) não foi
reportado ao OCS. motivo_original=Incidente OCS INC-1234
ativado_por=noc@example.com desativado_por=oncall@example.com
```

Quebras de linha incorporadas no `reason` são achatadas para espaços. Um comentário nunca pode dividir um registro CDR. Veja Formato de CDR.

Métricas

Veja Métricas. Alerta sobre `charging_override_active` para que um operador seja informado enquanto a sobrescrição estiver ativa.

Painel Web

A página de Sessões PGW mostra um banner enquanto a sobrescrição está ativa. O banner contém o motivo, o ator e o tempo restante.

Configuração

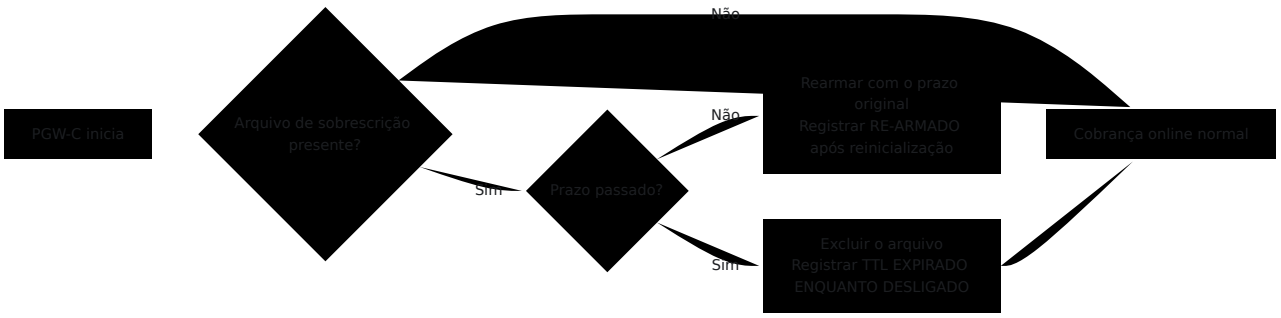
A sobrescrição não precisa de configuração para funcionar. Uma chave opcional controla onde o estado armado é escrito para que sobreviva a uma reinicialização.

```
config :pgw_c,
  # Onde a sobrescrição armada é persistida através de uma
reinicialização
  charging_override_file: "/var/lib/pgw_c/charging_override.json"
```

Parâmetro	Tipo	Obrigatório	Padrã
`charging_override_file`	String	Não	`/var/lib/pgw_c/chargi`

Comportamento de Reinicialização

Uma falha durante um incidente não deve rearmar silenciosamente o corte para cada assinante. O PGW-C escreve a sobrescrição armada no `charging_override_file` e a lê na inicialização.



Uma reinicialização não estende o TTL. O PGW-C mantém o original `expires_at`.

Se o PGW-C não puder escrever o arquivo, ele registra um aviso e continua. A sobrescrição está em vigor no nó, mas não sobrevive a uma reinicialização. Verifique as permissões do sistema de arquivos no diretório.

Métricas

Métrica: `charging_override_active`
Tipo: Gauge
Descrição: `1` enquanto a sobrescrição a nível de nó desativa a cobrança online, `0` caso contrário.
Rótulos: nenhum

Métrica: `charging_override_remaining_seconds`
Tipo: Gauge
Descrição: Segundos restantes antes que a sobrescrição expire automaticamente. `0` quando nenhuma sobrescrição está ativa.
Rótulos: nenhum

Ambos os gauges são atualizados no intervalo de polling de métricas. Veja Monitoramento & Métricas para o intervalo de polling e a configuração do exportador.

Consultas de exemplo:

```
# Qualquer nó com cobrança online ignorada agora
max(charging_override_active)

# Nós onde a sobrescrição expira em menos de 10 minutos
charging_override_remaining_seconds > 0 and
charging_override_remaining_seconds < 600
```

Alerta de exemplo:

```
- alert: OnlineChargingOverrideActive
  expr: charging_override_active == 1
  for: 5m
  labels:
    severity: critical
  annotations:
    summary: "A cobrança online está sendo ignorada em {{
$labels.instance }}"
    description: "O uso do assinante está sendo medido para CDRs,
mas não está tarifado online."
```

Solução de Problemas

A solicitação de ativação retorna 400

Sintomas: O corpo da resposta nomeia `ttl_seconds` ou `reason`.

Possíveis causas:

- `ttl_seconds` está ausente, `0`, negativo ou maior que `86400`
- `reason` está ausente ou em branco

Resolução:

1. Defina `ttl_seconds` como um inteiro entre `1` e `86400`.
2. Defina `reason` como texto que nomeie o incidente.

A varredura de alívio relata falhas

Sintomas: `relief.failed` é maior que `0` na resposta de ativação.

Possíveis causas:

- Um processo de sessão não respondeu dentro do tempo limite da chamada
- O UPF não aceitou a Modificação de Sessão PFCP

Uma sessão que se desconectou enquanto a varredura estava em execução não é contada como uma falha. Ela não aplica mais nada, então não há nada a liberar.

Resolução:

1. Pesquise o log por `[CHARGING OVERRIDE] falhou ao liberar a aplicação de cobrança`.
2. Verifique a acessibilidade do UPF com `GET /api/upf`.
3. Execute `PUT /api/charging/override` novamente uma vez que o UPF esteja saudável. A varredura é segura para repetir. Uma sessão que não está bloqueada não é afetada.

Um assinante ainda está bloqueado após a varredura

Sintomas: `GET /api/charging/<imsi>` mostra que o jardim murado foi liberado, mas o tráfego não flui.

Possíveis causas:

- O OCS enviou uma Ação de Unidade Final de TERMINATE na conexão. Os parâmetros de encaminhamento nessa portadora foram limpos e não podem ser reconstruídos.

Resolução:

1. Pesquise o log por `Não é possível desbloquear FAR(s)` com esse IMSI.
2. Peça ao assinante para se reconectar. Uma nova conexão entra com encaminhamento de tráfego enquanto a sobrescrição está ativa.

A sobrescrição voltou após uma reinicialização

Sintomas: O log contém `[CHARGING OVERRIDE] RE-ARMADO após reinicialização`.

Este é o comportamento projetado. A sobrescrição ainda estava dentro de seu TTL quando o nó reiniciou.

Resolução:

1. Execute `DELETE /api/charging/override` se você não quiser mais.
2. Caso contrário, aguarde o TTL expirar.

A cobrança online está desligada e nenhuma sobrescrição está ativa

Sintomas: Nenhum CCR-Initial é enviado, e `GET /api/charging/override` reporta `active: false`.

Possíveis causas:

- `charging.online` está definido como `:off` na configuração em tempo de execução
- `gy.enabled` é `false`
- O PCRF definiu o AVP Gx CCA `Online` como desativado

Resolução:

1. Verifique os blocos `charging` e `gy`. Veja Referência de Configuração.
2. Verifique o Gx CCA com `GET /api/charging/<imsi>`. Veja Interface Diameter Gx.

Documentação Relacionada

- Interface Diameter Gy - cobrança online, cota e jardim murado
- Formato de CDR - cobrança offline e o arquivo CDR

- Monitoramento & Métricas - exportador de métricas e painéis
- Solução de Problemas - procedimentos gerais de solução de problemas
- Guia de Operações - ponto de entrada

Guia de Configuração do OmniPGW

Referência Completa para Configuração do runtime.exs

por Omnitouch Network Services

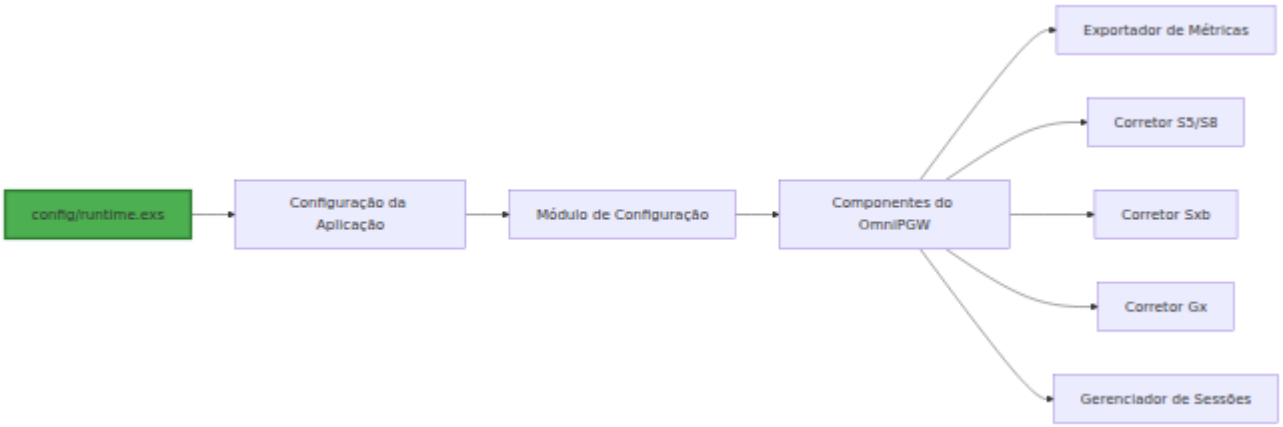
Índice

1. Visão Geral
2. Estrutura do Arquivo de Configuração
3. Configuração de Métricas
4. Configuração do Diameter/Gx
5. Configuração do S5/S8
6. Configuração do Gn/Gp (GGSN)
7. Configuração do Sxb/PFCP
    - Estratégias de Seleção de UPF
    - Balanceamento de Carga com Pools de UPF
    - Seleção Baseada em DNS
    - Modo de Execução Secundária
8. Configuração do Pool de IP do UE
9. Configuração do PCO
10. Configuração da API OAM
11. Exemplo Completo
12. Validação da Configuração

Visão Geral

OmniPGW utiliza **configuração em tempo de execução** definida em `config/runtime.exs`. Este arquivo é avaliado na **inicialização da aplicação** e permite configuração dinâmica com base em variáveis de ambiente ou fontes externas.

Filosofia de Configuração



Princípios Chave:

- **Fonte Única de Verdade** - Toda configuração em um único arquivo
- **Segurança de Tipo** - Configuração validada na inicialização
- **Flexibilidade de Ambiente** - Suporte para desenvolvimento, teste, produção
- **Defaults Claros** - Defaults sensatos com substituições explícitas

---

Estrutura do Arquivo de Configuração

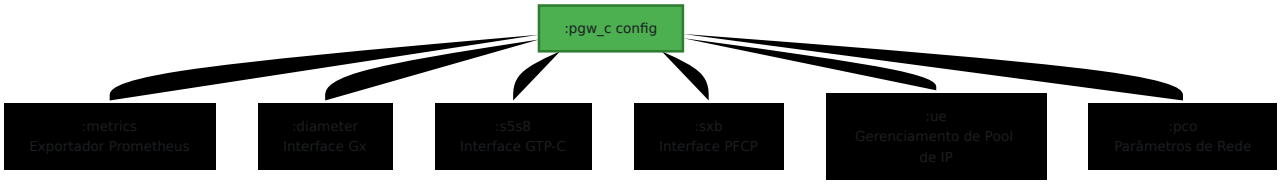
Localiza��ão do Arquivo

```
pgw_c/
├── config/
│   ├── config.exs         # Configuração base (importa
runtime.exs)
│   ├── dev.exs            # Configuração específica para
desenvolvimento
│   ├── prod.exs           # Configuração específica para produção
│   └── runtime.exs        # ← Arquivo de configuração principal
```

Estrutura de Nível Superior

```
# config/runtime.exs
import Config

config :logger, level: :info

config :pgw_c,
  metrics: %{...},
  diameter: %{...},
  s5s8: %{...},
  sxb: %{...},
  ue: %{...},
  pco: %{...}
```

Seções de Configuração



Configuração de Métricas

Propósito

Configurar o exportador de métricas Prometheus para monitorar o OmniPGW.

Bloco de Configuração

```
config :pgw_c,
  metrics: %{
    # Habilitar/desabilitar exportador de métricas
    enabled: true,

    # Endereço IP para vincular o servidor HTTP
    ip_address: "0.0.0.0",

    # Porta para o endpoint de métricas
    port: 9090,

    # Com que frequência consultar registros (milissegundos)
    registry_poll_period_ms: 10_000
  }
```

Parâmetros

Parâmetro	Tipo	Padrão	Descrição
`enabled`	Booleano	`true`	Habilitar exportador de métricas
`ip_address`	String (IP)	`"0.0.0.0"`	Endereço de vinculação (0.0.0.0 = todas as interfaces)
`port`	Inteiro	`9090`	Porta HTTP para o endpoint `/metrics`
`registry_poll_period_ms`	Inteiro	`10_000`	Intervalo de consulta para contagens de registro

Exemplos

Produção - Vincular a um IP específico:

```
metrics: %{
  enabled: true,
  ip_address: "10.0.0.20",  # Rede de gerenciamento
  port: 9090,
  registry_poll_period_ms: 5_000  # Consultar a cada 5 segundos
}
```

Desenvolvimento - Apenas localhost:

```
metrics: %{
  enabled: true,
  ip_address: "127.0.0.1",
  port: 42069,  # Porta não padrão
  registry_poll_period_ms: 10_000
}
```

Desabilitar métricas:

```
metrics: %{
  enabled: false
}
```

Acessando Métricas

```
# Endpoint padrão
curl http://<ip_address>:<port>/metrics

# Exemplo
curl http://10.0.0.20:9090/metrics
```

Veja: Guia de Monitoramento & Métricas para documentação detalhada sobre métricas.

Configuração do Diameter/Gx

Propósito

Configurar o protocolo Diameter para a interface Gx (comunicação PCRF).

Bloco de Configuração

```
config :pgw_c,
  diameter: %{
    # Endereço IP para escutar conexões Diameter
    listen_ip: "0.0.0.0",

    # Identidade Diameter do OmniPGW (Origin-Host)
    host: "omnipgw.epc.mnc001.mcc001.3gppnetwork.org",

    # Reino Diameter do OmniPGW (Origin-Realm)
    realm: "epc.mnc001.mcc001.3gppnetwork.org",

    # Lista de pares PCRF
    peer_list: [
      %{
        # Identidade Diameter do PCRF
        host: "pcrf.epc.mnc001.mcc001.3gppnetwork.org",

        # Reino do PCRF
        realm: "epc.mnc001.mcc001.3gppnetwork.org",

        # Endereço IP do PCRF
        ip: "10.0.0.30",

        # Iniciar conexão com o PCRF
        initiate_connection: true
      }
    ]
  }
```

Parâmetros

Parâmetro	Tipo	Obrigatório	Descrição
`listen_ip`	String (IP)	Sim	Endereço de escuta Diameter
`host`	String (FQDN)	Sim	Origin-Host do OmniPGW (deve ser FQDN)
`realm`	String (Domínio)	Sim	Origin-Realm do OmniPGW
`peer_list`	Lista	Sim	Configurações de pares PCRF

Configuração do Par:

Parâmetro	Tipo	Obrigatório	Descrição
`host`	String (FQDN)	Sim	Identidade Diameter do PCRF
`realm`	String (Domínio)	Sim	Reino do PCRF
`ip`	String (IP)	Sim	Endereço IP do PCRF
`initiate_connection`	Booleano	Sim	Se o OmniPGW se conecta ao PCRF

Formato FQDN

Identidades Diameter DEVEM ser FQDNs:

```
# CORRETO
host: "omnipgw.epc.mnc001.mcc001.3gppnetwork.org"

# INCORRETO
host: "omnipgw"              # Não é um FQDN
host: "10.0.0.20"            # IP não permitido
```

Formato 3GPP:

```
<hostname>.epc.mnc<MNC>.mcc<MCC>.3gppnetwork.org

Exemplos:
- omnipgw.epc.mnc001.mcc001.3gppnetwork.org  (MCC=001, MNC=001)
- pgw-c.epc.mnc260.mcc310.3gppnetwork.org   (MCC=310, MNC=260 - T-Mobile dos EUA)
```

Exemplos

Um único PCRF:

```
diameter: %{
  listen_ip: "0.0.0.0",
  host: "omnipgw.epc.mnc001.mcc001.3gppnetwork.org",
  realm: "epc.mnc001.mcc001.3gppnetwork.org",
  peer_list: [
    %{
      host: "pcrf.epc.mnc001.mcc001.3gppnetwork.org",
      realm: "epc.mnc001.mcc001.3gppnetwork.org",
      ip: "10.0.0.30",
      initiate_connection: true
    }
  ]
}
```

Múltiplos PCRFs (Redundância):

```
diameter: %{
  listen_ip: "0.0.0.0",
  host: "omnipgw.epc.mnc001.mcc001.3gppnetwork.org",
  realm: "epc.mnc001.mcc001.3gppnetwork.org",
  peer_list: [
    %{
      host: "pcrf-primary.epc.mnc001.mcc001.3gppnetwork.org",
      realm: "epc.mnc001.mcc001.3gppnetwork.org",
      ip: "10.0.1.30",
      initiate_connection: true
    },
    %{
      host: "pcrf-backup.epc.mnc001.mcc001.3gppnetwork.org",
      realm: "epc.mnc001.mcc001.3gppnetwork.org",
      ip: "10.0.2.30",
      initiate_connection: true
    }
  ]
}
```

Conexão Iniciada pelo PCRF:

```
diameter: %{
  listen_ip: "0.0.0.0",
  host: "omnipgw.epc.mnc001.mcc001.3gppnetwork.org",
  realm: "epc.mnc001.mcc001.3gppnetwork.org",
  peer_list: [
    %{
      host: "pcrf.epc.mnc001.mcc001.3gppnetwork.org",
      realm: "epc.mnc001.mcc001.3gppnetwork.org",
      ip: "10.0.0.30",
      initiate_connection: false  # Aguardar o PCRF se conectar
    }
  ]
}
```

Veja: Documentação da Interface Diameter Gx

Configuração do S5/S8

Propósito

Configurar a interface GTP-C para comunicação com o SGW-C.

Bloco de Configuração

```
config :pgw_c,
  s5s8: %{
    # Endereço IPv4 local para a interface S5/S8
    local_ipv4_address: "10.0.0.20",

    # Opcional: Endereço IPv6 local
    local_ipv6_address: nil,

    # Opcional: Substituir a porta GTP-C padrão (2123)
    local_port: 2123,

    # Timeout de requisição GTP-C em milissegundos (padrão: 500ms)
    # Timeout por tentativa ao aguardar respostas GTP-C
    request_timeout_ms: 500,

    # Número de tentativas de reenvio para requisições GTP-C
(padrão: 3)
    # Tempo máximo total de espera = request_timeout_ms *
request_attempts
    request_attempts: 3
  }
```

Parâmetros

Parâmetro	Tipo	Padrão	Descrição
`local_ipv4_address`	String (IPv4)	Obrigatório	Endereço IPv4 da interface S5/S8
`local_ipv6_address`	String (IPv6)	`nil`	Endereço IPv6 da interface S5/S8 (opcional)
`local_port`	Inteiro	`2123`	Porta UDP para GTP-C (porta padrão 2123)
`request_timeout_ms`	Inteiro	`500`	Timeout por tentativa de reenvio em milissegundos
`request_attempts`	Inteiro	`3`	Número de tentativas de reenvio antes de desistir

Detalhes do Protocolo

- **Protocolo:** GTP-C Versão 2
- **Transporte:** UDP
- **Porta Padrão:** 2123
- **Direção:** Recebe do SGW-C

Exemplos

Apenas IPv4 (Comum):

```
s5s8: %{
  local_ipv4_address: "10.0.0.20"
}
```

IPv4 + IPv6 Dual-Stack:

```
s5s8: %{
  local_ipv4_address: "10.0.0.20",
  local_ipv6_address: "2001:db8::20"
}
```

Porta Personalizada (Não Padrão):

```
s5s8: %{
  local_ipv4_address: "10.0.0.20",
  local_port: 2124  # Porta personalizada
}
```

Rede de Alta Latência:

```
s5s8: %{
  local_ipv4_address: "10.0.0.20",
  request_timeout_ms: 1500,  # 1.5 segundos por tentativa
  request_attempts: 3         # Total: 4.5 segundos no máximo
}
```

Configuração de Timeout

A interface S5/S8 utiliza timeouts configuráveis para transações de requisição/resposta GTP-C (Create Bearer Request, Delete Bearer Request).

Cálculo do Tempo Total de Espera:

```
Tempo Máximo Total = request_timeout_ms × request_attempts
Padrão: 500ms × 3 = 1.5 segundos
```

Diretrizes de Ajuste:

Latência de Rede	Timeout Recomendado	Tempo Total de Espera
Baixa latência (<50ms)	200-300ms	600-900ms
Normal (50-150ms)	500ms (padrão)	1.5s
Alta latência (>150ms)	1000-2000ms	3-6s
Satélite/instável	2000-3000ms	6-9s

Quando Ajustar:

- **Aumentar o timeout** se houver frequentes erros de "Create Bearer Request timed out", mas o Wireshark mostra respostas chegando
- **Diminuir o timeout** para detecção de falhas mais rápida em ambientes de baixa latência
- **Aumentar tentativas de reenvio** para redes não confiáveis com perda de pacotes

Comportamento do Timeout:

- Em caso de timeout, o erro é registrado: `"Create Bearer Request timed out"`
- Erro Diameter retornado ao PCRF: Result-Code 5012 (UNABLE_TO_COMPLY)
- Bearer permanece em armazenamento inicial para limpeza quando Charging-Rule-Remove chega

Planejamento de Rede

Seleção de Endereço IP:

- Use rede de gerenciamento/sinalização dedicada
- Garantir acessibilidade de todos os nós SGW-C
- Considerar redundância (VRRP/HSRP) para HA

Regras de Firewall:

```
# Permitir GTP-C do SGW-C
iptables -A INPUT -p udp --dport 2123 -s <sgw_c_network> -j ACCEPT
```

Configuração do Gn/Gp (GGSN)

Propósito

Configurar a interface GTP-C v1 para comunicação com SGSNs, habilitando a funcionalidade GGSN para redes 2G/3G.

Bloco de Configuração

```
config :pgw_c,
  gn: %{
    # Endereço IPv4 local para a interface Gn
    local_ipv4_address: "10.0.0.20",

    # Opcional: Endereço IPv6 local (a maioria das redes 2G/3G é
apenas IPv4)
    local_ipv6_address: nil,

    # Porta GTP-C (compartilhada com S5/S8)
    local_port: 2123
  },

  # Servidores DNS retornados no PCO (compartilhados com S5/S8)
  dns: %{
    primary_ipv4: {8, 8, 8, 8},
    secondary_ipv4: {8, 8, 4, 4}
  }
```

Parâmetros

Parâmetro	Tipo	Padrão	Descrição
`local_ipv4_address`	String (IPv4)	Obrigatório	Endereço IPv4 da interface Gn (endereço GGSN)
`local_ipv6_address`	String (IPv6)	`nil`	Endereço IPv6 da interface Gn (opcional, raramente usado)
`local_port`	Inteiro	`2123`	Porta UDP para GTP-C v1 (compartilhada com S5/S8)

Parâmetros DNS

Parâmetro	Tipo	Padrão	Descrição
`primary_ipv4`	Tupla	`{8, 8, 8, 8}`	Servidor DNS primário para resposta PCO
`secondary_ipv4`	Tupla	`{8, 8, 4, 4}`	Servidor DNS secundário para resposta PCO

Detalhes do Protocolo

- **Protocolo:** GTP-C Versão 1 (3GPP TS 29.060)
- **Transporte:** UDP
- **Porta Padrão:** 2123 (compartilhada com GTP-C v2)
- **Direção:** Recebe de SGSNs

Coexistência com S5/S8

OmniPGW pode operar como PGW (4G) e GGSN (2G/3G) simultaneamente:

- Ambas as interfaces compartilham a porta UDP 2123
- A versão GTP é detectada automaticamente a partir dos cabeçalhos das mensagens
- O mesmo endereço IP pode atender tanto o tráfego 4G quanto 2G/3G
- Infraestrutura compartilhada: pools de IP, UPF/PFCP, cobrança online

Veja: Documentação da Interface Gn/Gp

Configuração do Sxb/PFCP

Propósito

Configurar a interface PFCP para comunicação com PGW-U (Plano do Usuário).

Bloco de Configuração

```
config :pgw_c,
  sxb: %{
    # Endereço IP local para comunicação PFCP
    local_ip_address: "10.0.0.20",

    # Opcional: Substituir a porta PFCP padrão (8805)
    local_port: 8805
  }
```

Parâmetros

Parâmetro	Tipo	Padrão	Descrição
`local_ip_address`	String (IP)	Obrigatório	Endereço de escuta PFCP
`local_port`	Inteiro	`8805`	Porta UDP PFCP

> **Importante:**

> - **Todos os pares UPF são registrados automaticamente** a partir da configuração `upf_selection` (regras + pool de fallback) na inicialização
> - UPFs registrados automaticamente usam defaults sensatos:
>   - Nome gerado automaticamente: `"UPF-<ip>:<port>"`
>   - Associação PFCP passiva (aguardar UPF iniciar)
>   - Intervalo de heartbeat de 5 segundos
> - Regras e pools de seleção de UPF são configurados na seção separada `upf_selection`. Veja Estratégias de Seleção de UPF abaixo.
> - Registro dinâmico de UPF é suportado para UPFs descobertos por DNS que não estão na configuração

Exemplos

Configuração Mínima:

```
sxb: %{
  local_ip_address: "10.0.0.20"
}

# Todos os UPFs em upf_selection serão registrados automaticamente
com:
# - Nome gerado automaticamente: "UPF-10.0.0.21:8805"
# - Associação PFCP passiva (aguardar UPF conectar)
# - Intervalo de heartbeat de 5 segundos
```

Porta PFCP Personalizada:

```
sxb: %{
  local_ip_address: "10.0.0.20",
  local_port: 8806  # Porta PFCP não padrão
}
```

Exemplo Completo com Seleção de UPF:

```
sxb: %{
  local_ip_address: "10.0.0.20"
},
upf_selection: %{
  rules: [
    %{
      name: "Pool IMS",
      priority: 10,
      match_field: :apn,
      match_regex: ~r/^ims$/,
      upf_pool: [
        %{remote_ip_address: "10.0.1.21", remote_port: 8805,
weight: 100},
        %{remote_ip_address: "10.0.1.22", remote_port: 8805,
weight: 100}
      ]
    }
  ],
  fallback_pool: [
    %{remote_ip_address: "10.0.2.21", remote_port: 8805, weight:
100}
  ]
}
# Todos os 3 UPFs (10.0.1.21, 10.0.1.22, 10.0.2.21) são
registrados automaticamente
```

Seleção Baseada em DNS (Registro Dinâmico):

```
sxb: %{
  local_ip_address: "10.0.0.20"
},
upf_selection: %{
  dns_enabled: true,
  dns_query_priority: [:ecgi, :tai],
  dns_suffix: "epc.3gppnetwork.org",
  fallback_pool: [
    %{remote_ip_address: "10.0.2.21", remote_port: 8805, weight:
100}
  ]
}
# UPFs descobertos por DNS serão registrados dinamicamente no
primeiro uso
```

Estratégias de Seleção de UPF

Importante: A configuração de seleção de UPF foi simplificada. Todos os pares UPF são registrados automaticamente a partir da configuração `upf_selection`.

Estrutura de Configuração

A seleção de UPF é configurada na seção **`upf_selection`** que define:

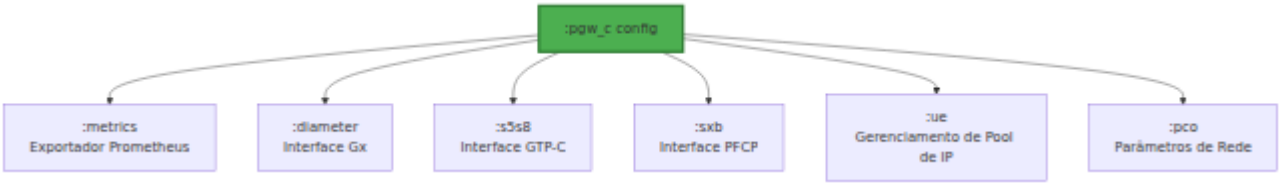
1. **Regras Estáticas** - Roteamento baseado em padrões com pools de balanceamento de carga
2. **Configurações de DNS** - Descoberta dinâmica de UPF baseada em localização
3. **Pool de Fallback** - Pool padrão quando nenhuma regra corresponde e o DNS falha

Ordem de Prioridade de Seleção

1. **Regras Estáticas** (Maior Prioridade) - Roteamento baseado em padrões com pools de balanceamento de carga

2. **Seleção Baseada em DNS** (Menor Prioridade) - Descoberta dinâmica de UPF baseada em localização
3. **Pool de Fallback** (Menor Prioridade) - Pool padrão quando nenhuma regra corresponde e o DNS falha

Fluxo de Decisão de Seleção de UPF



Campos de Correspondência Disponíveis

Regras estáticas podem corresponder a qualquer um desses atributos de sessão:

Campo de Correspondência	Descrição	Exemplo de Padrão
`:imsi`	Identidade Internacional do Assinante Móvel	`^001001.*` (rede de teste)
`:apn`	Nome do Ponto de Acesso / DNN	`^internet\.` ou `^ims\.`
`:serving_network_plmn_id`	Identificador da rede de serviço	`^001001$`
`:sgw_ip_address`	Endereço IP do SGW	`^10\.100\..*`
`:uli_tai_plmn_id`	ID PLMN da Área de Rastreamento	`^313.*`
`:uli_ecgi_plmn_id`	ID PLMN da Célula E-UTRAN	`^313.*`

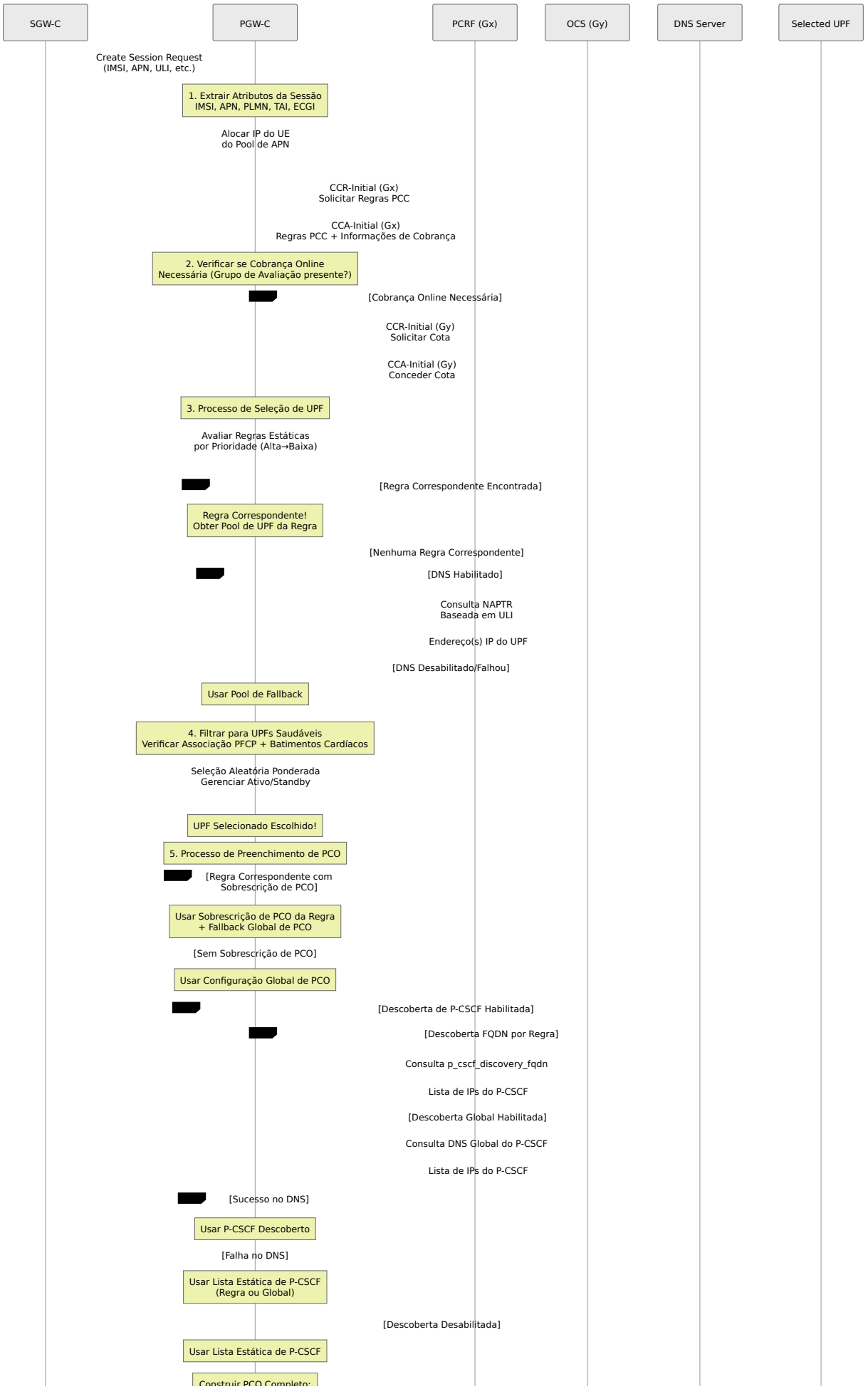
Comparação de Métodos de Seleção

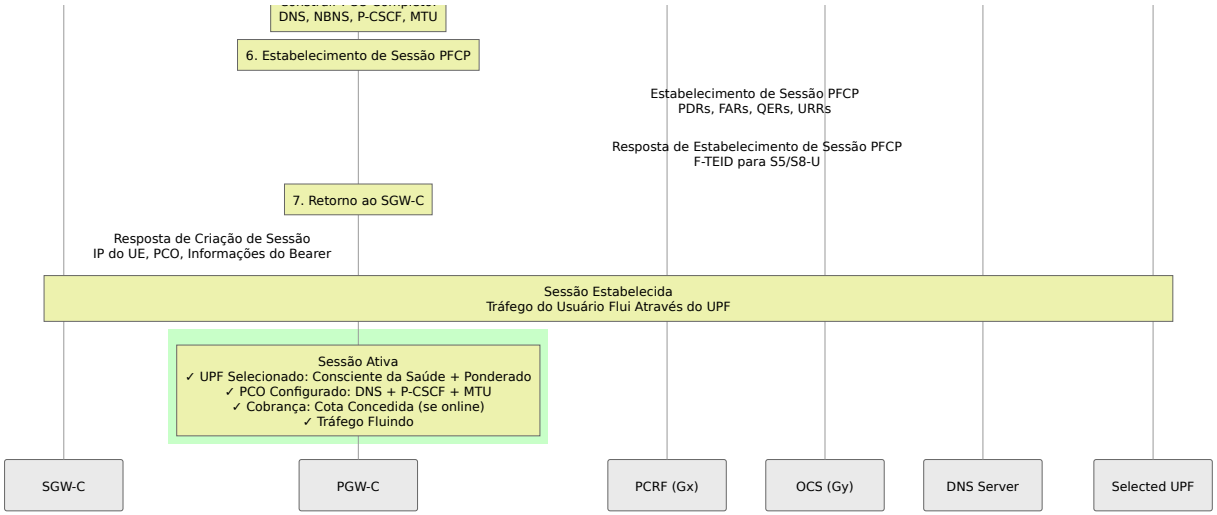
Método	Quando Usar	Prós	Contras
Pools de UPF	Implantações de produção	Balanceamento de carga, HA, pesos flexíveis	Requer múltiplos UPFs
Baseado em APN	Diferenciação de serviço	Roteamento separado IMS/Internet	Configuração estática
Baseado em IMSI	Cenários de roaming	Roteamento geográfico	Complexidade de regex
Baseado em DNS	MEC/Computação de borda	Dinâmico, ciente da localização	Requer infraestrutura DNS
Pool de Fallback	Rede de segurança	Sempre ter um UPF	Pode não ser ótimo
Modo de Execução Secundária	Testando configurações	Teste seguro	Sem tráfego real

---

Fluxo Completo de Estabelecimento de Sessão

Este diagrama mostra o fluxo completo de ponta a ponta do estabelecimento de sessão, incluindo seleção de UPF e preenchimento de PCO:


SGW-C
PGW-C
PCRF (Gx)
OCS (Gy)
DNS Server
Selected UPF
Create Session Request
(IMSI, APN, ULI, etc.)
1. Extrair Atributos da Sessão
IMSI, APN, PLMN, TAI, ECGI
Alocar IP do UE
do Pool de APN
CCR-Initial (Gx)
Solicitar Regras PCC
CCA-Initial (Gx)
Regras PCC + Informações de Cobrança
2. Verificar se Cobrança Online
Necessária (Grupo de Avaliação presente?)
[Cobrança Online Necessária]
CCR-Initial (Gy)
Solicitar Cota
CCA-Initial (Gy)
Conceder Cota
3. Processo de Seleção de UPF
Avaliar Regras Estáticas
por Prioridade (Alta→Baixa)
[Regra Correspondente Encontrada]
Regra Correspondente!
Obter Pool de UPF da Regra
[Nenhuma Regra Correspondente]
[DNS Habilitado]
Consulta NAPTR
Baseada em ULI
Endereço(s) IP do UPF
[DNS Desabilitado/Falhou]
Usar Pool de Fallback
4. Filtrar para UPFs Saudáveis
Verificar Associação PFCP + Batimentos Cardíacos
Seleção Aleatória Ponderada
Gerenciar Ativo/Standby
UPF Selecionado Escolhido!
5. Processo de Preenchimento de PCO
[Regra Correspondente com
Sobrescrição de PCO]
Usar Sobrescrição de PCO da Regra
+ Fallback Global de PCO
[Sem Sobrescrição de PCO]
Usar Configuração Global de PCO
[Descoberta de P-CSCF Habilitada]
[Descoberta FQDN por Regra]
Consulta p_cscf_discovery_fqdn
Lista de IPs do P-CSCF
[Descoberta Global Habilitada]
Consulta DNS Global do P-CSCF
Lista de IPs do P-CSCF
[Sucesso no DNS]
Usar P-CSCF Descoberto
[Falha no DNS]
Usar Lista Estática de P-CSCF
(Regra ou Global)
[Descoberta Desabilitada]
Usar Lista Estática de P-CSCF
Construir PCO Completo:



Pontos de Decisão Chave:

1. **Prioridade de Seleção de UPF:**
    - Regras Estáticas (Correspondência de Padrão) → Descoberta DNS → Pool de Fallback
    - Filtragem de saúde aplicada em todas as etapas
    - Lógica Ativo/Standby para alta disponibilidade
    - **Veja:** Interface PFCP para detalhes de comunicação do UPF
2. **Prioridade de Preenchimento de PCO:**
    - Sobrescrição de PCO da Regra → Descoberta DNS do P-CSCF → Configuração Global de PCO
    - Mesclagem por campo (a regra sobrescreve campos específicos, o global fornece padrões)
    - **Veja:** Configuração de PCO para parâmetros detalhados de PCO
3. **Prioridade de Descoberta do P-CSCF:**
    - FQDN por Regra → Descoberta DNS Global → PCO de Regra Estática → PCO Estático Global
    - **Veja:** Monitoramento do P-CSCF para métricas de descoberta e rastreamento de saúde
4. **Integração de Cobrança:**

- O PCRF determina se a cobrança online é necessária (Grupo de Avaliação + Online=1)
- OCS concede cota antes do estabelecimento da sessão
- PGW-C rastreia a cota e solicita mais via CCR-Update
- **Veja:** Interface Diameter Gx e Interface Diameter Gy para detalhes de cobrança

Exemplo Completo de Configuração

Aqui está um exemplo completo mostrando seleção de UPF de múltiplos pools com registro automático de pares:

```
config :pgw_c,
  # Interface PFCP - Todos os UPFs são registrados automaticamente
a partir da seleção de upf
  sxb: %{
    local_ip_address: "127.0.0.20"
  },

  # Lógica de Seleção de UPF - Todos os UPFs definidos aqui são
registrados automaticamente
  upf_selection: %{
    # Configurações de seleção baseadas em DNS
    dns_enabled: false,
    dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],
    dns_suffix: "epc.3gppnetwork.org",
    dns_timeout_ms: 5000,

    # Regras de seleção estática (avaliadas em ordem de
prioridade)
    rules: [
      # Regra 1: Tráfego IMS - Maior Prioridade
      %{
        name: "Tráfego IMS",
        priority: 20,
        match_field: :apn,
        match_regex: "^ims",
        upf_pool: [
          %{remote_ip_address: "10.100.2.21", remote_port: 8805,
weight: 80},
          %{remote_ip_address: "10.100.2.22", remote_port: 8805,
weight: 20}
        ]
      },

      # Regra 2: APN Empresarial
      %{
        name: "Tráfego Empresarial",
        priority: 15,
        match_field: :apn,
        match_regex: "^(enterprise|corporate)\.apn",
        upf_pool: [
          %{remote_ip_address: "10.100.3.21", remote_port: 8805,
weight: 100}
        ]
```

```
      },

      # Regra 3: Tráfego de Internet - Balanceado
      %{
        name: "Tráfego de Internet",
        priority: 5,
        match_field: :apn,
        match_regex: "^internet",
        upf_pool: [
          %{remote_ip_address: "10.100.1.21", remote_port: 8805,
weight: 33},
          %{remote_ip_address: "10.100.1.22", remote_port: 8805,
weight: 33},
          %{remote_ip_address: "10.100.1.23", remote_port: 8805,
weight: 34}
        ]
      }
    ],

    # Pool de fallback - Usado quando nenhuma regra corresponde e
o DNS falha
    fallback_pool: [
      %{remote_ip_address: "127.0.0.21", remote_port: 8805,
weight: 100}
    ]
  }
```

Principais Recursos

Formato Atual:

- **Registro Automático**: Todos os UPFs da `upf_selection` são registrados automaticamente na inicialização
- **Configuração Centralizada**: Toda a seleção de UPF e configuração de pares em uma seção
- **Pools Necessários**: Todas as regras usam o formato `upf_pool` (mesmo para um único UPF)
- **Fallback Estruturado**: `fallback_pool` dedicado com distribuição ponderada
- **Integração DNS**: Configurações DNS ao lado das regras de seleção

- 🔄 **Registro Dinâmico**: UPFs descobertos por DNS são registrados automaticamente no primeiro uso
- 🏥 **Monitoramento de Saúde**: Todos os UPFs configurados são monitorados com batimentos cardíacos de 5 segundos

Migração do Formato Anterior:

- Removido: campo `sxb.peer_list` (não é mais necessário)
- Removido: `selection_list` incorporado nas configurações de pares
- Todas as definições de UPF agora vão nas regras de `upf_selection` e no pool de fallback

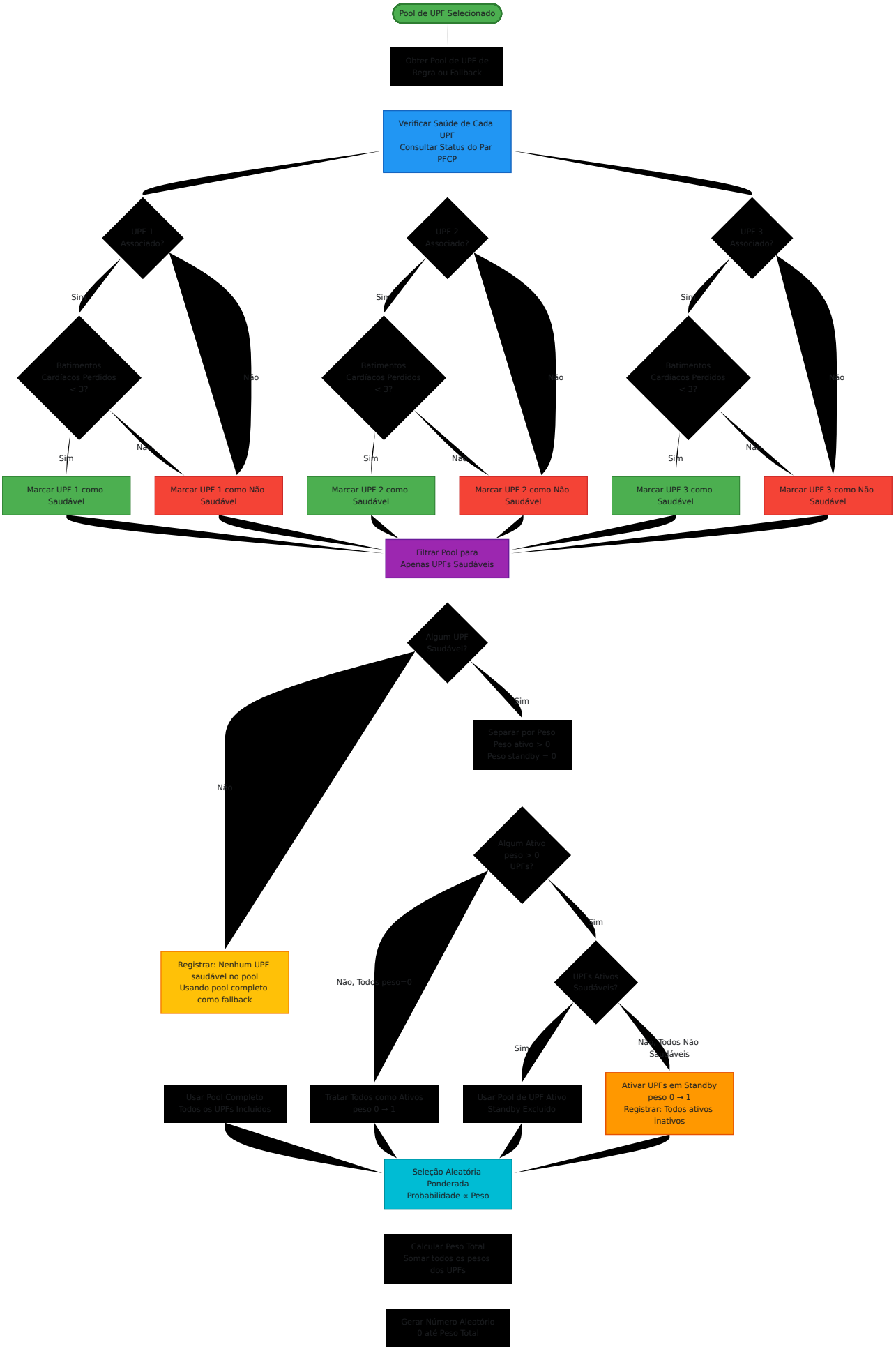
Como Funcionam os Pools de UPF:

1. **Seleção Consciente da Saúde**: Apenas UPFs saudáveis recebem tráfego
    - Saudável = associação PFCP ativa + menos de 3 batimentos cardíacos consecutivos perdidos
    - UPFs não saudáveis são automaticamente filtrados
    - Retorna a todos os UPFs se nenhum estiver saudável (falha rápida)
2. **Suporte Ativo/Standby**: Use `weight: 0` para UPFs em standby
    - **UPFs Ativos** (peso > 0): Recebem tráfego quando saudáveis
    - **UPFs em Standby** (peso == 0): Recebem tráfego apenas quando todos os UPFs ativos estão inativos
    - UPFs em standby são tratados como `weight: 1` quando ativados
3. **Seleção Aleatória Ponderada**: Cada sessão é atribuída aleatoriamente a um UPF saudável com base nos pesos
    - No exemplo acima: 70% vão para .21, 20% para .22, 10% para .23
    - Peso maior = mais sessões atribuídas a esse UPF
    - Pesos iguais = distribuição igual
4. **Registro Automático**: Todos os UPFs nos pools são registrados automaticamente na inicialização
    - Nomes gerados automaticamente: `"UPF-<ip>:<port>"`

- Configurações padrão: associação PFCP passiva, batimentos cardíacos de 5 segundos
- Rastreamento imediato de saúde para todos os UPFs configurados

Seleção Consciente da Saúde com

Ativo/Standby


Pool de UPF Selecionado
Obter Pool de UPF de Regra ou Fallback
Verificar Saúde de Cada UPF
Consultar Status do Par PFCP
UPF 1 Associado?
UPF 2 Associado?
UPF 3 Associado?
Sim
Não
Batimentos Cardíacos Perdidos < 3?
Marcar UPF 1 como Saudável
Marcar UPF 1 como Não Saudável
Marcar UPF 2 como Saudável
Marcar UPF 2 como Não Saudável
Marcar UPF 3 como Saudável
Marcar UPF 3 como Não Saudável
Filtrar Pool para Apenas UPFs Saudáveis
Algum UPF Saudável?
Separar por Peso
Peso ativo > 0
Peso standby = 0
Registrar: Nenhum UPF saudável no pool
Usando pool completo como fallback
Algum Ativo peso > 0 UPFs?
UPFs Ativos Saudáveis?
Usar Pool Completo
Todos os UPFs Incluídos
Tratar Todos como Ativos
peso 0 → 1
Usar Pool de UPF Ativo
Standby Excluído
Ativar UPFs em Standby
peso 0 → 1
Registrar: Todos ativos inativos
Seleção Aleatória Ponderada
Probabilidade ∝ Peso
Calcular Peso Total
Somar todos os pesos dos UPFs
Gerar Número Aleatório
0 até Peso Total



Exemplo de Seleção Aleatória Ponderada:

```
Pool: [
  UPF-A: peso 50, saudável ✓
  UPF-B: peso 30, saudável ✓
  UPF-C: peso 20, saudável ✓
]

Peso Total: 50 + 30 + 20 = 100

Faixas de Peso:
  UPF-A: 0-49   (50%)
  UPF-B: 50-79  (30%)
  UPF-C: 80-99  (20%)

Número aleatório: 63 → Seleciona UPF-B
Número aleatório: 15 → Seleciona UPF-A
Número aleatório: 91 → Seleciona UPF-C
```

Exemplo de Failover Ativo/Standby:

```
Pool Inicial: [
  UPF-A: peso 100, saudável ✓   (Ativo)
  UPF-B: peso 0, saudável ✓     (Standby)
]

Cenário 1: UPF-A Saudável
→ Usar Pool Ativo: [UPF-A: 100]
→ Todo tráfego para UPF-A

Cenário 2: UPF-A Falha
→ Nenhum UPF ativo saudável
→ Ativar Standby: [UPF-B: 1]
→ Todo tráfego falha para UPF-B
→ Registrar: "Todos os UPFs ativos inativos, ativando UPFs em
standby"

Cenário 3: Ambos Não Saudáveis
→ Nenhum UPF saudável
→ Usar pool completo: [UPF-A: 100, UPF-B: 0]
→ Selecionar com pesos (tentar conexão, pode falhar)
→ Registrar: "Nenhum UPF saudável no pool, usando pool completo
como fallback"
```

Padrões Comuns de Peso:

```
# Distribuição igual (25% cada)
upf_pool: [
  %{remote_ip_address: "10.0.1.1", remote_port: 8805, weight: 1},
  %{remote_ip_address: "10.0.1.2", remote_port: 8805, weight: 1},
  %{remote_ip_address: "10.0.1.3", remote_port: 8805, weight: 1},
  %{remote_ip_address: "10.0.1.4", remote_port: 8805, weight: 1}
]

# Balanceamento de carga primário/backup (90% / 10%)
upf_pool: [
  %{remote_ip_address: "10.0.1.21", remote_port: 8805, weight:
90},
  %{remote_ip_address: "10.0.1.22", remote_port: 8805, weight: 10}
]

# Ativo/Standby (100% primário, 0% standby até que o primário
falhe)
upf_pool: [
  %{remote_ip_address: "10.0.1.21", remote_port: 8805, weight:
100},  # Ativo
  %{remote_ip_address: "10.0.1.22", remote_port: 8805, weight: 0}
# Standby (apenas quando ativo falhar)
]

# Ativo com múltiplos standbys (balanceado quando ativado)
upf_pool: [
  %{remote_ip_address: "10.0.1.1", remote_port: 8805, weight:
100},   # Ativo
  %{remote_ip_address: "10.0.1.2", remote_port: 8805, weight: 0},
# Standby 1
  %{remote_ip_address: "10.0.1.3", remote_port: 8805, weight: 0}
# Standby 2
]
# Resultado: Ativo recebe 100%. Se ativo falhar, standbys recebem
50/50%.

# Teste A/B (50% / 50%)
upf_pool: [
  %{remote_ip_address: "10.0.1.100", remote_port: 8805, weight:
50},  # Versão antiga
  %{remote_ip_address: "10.0.1.200", remote_port: 8805, weight:
```

```
50}   # Nova versão
]
```

Casos de Uso:

- **Failover Ativo/Standby**: Use `weight: 0` para UPFs em standby quentes que só ativam quando os primários falham
- **HA Consciente da Saúde**: Failover automático quando UPFs perdem associação PFCP ou perdem batimentos cardíacos
- **Escalonamento Horizontal**: Distribuir carga entre múltiplos UPFs para aumentar a capacidade
- **Alta Disponibilidade**: Distribuição automática evita sobrecarga de um único UPF
- **Implantações Gradativas**: Use pesos para implantações canário (ex: 95% antigo, 5% novo)
- **Otimização de Custos**: Roteie mais tráfego para UPFs de maior capacidade
- **Distribuição Geográfica**: Balancear sessões entre UPFs de borda

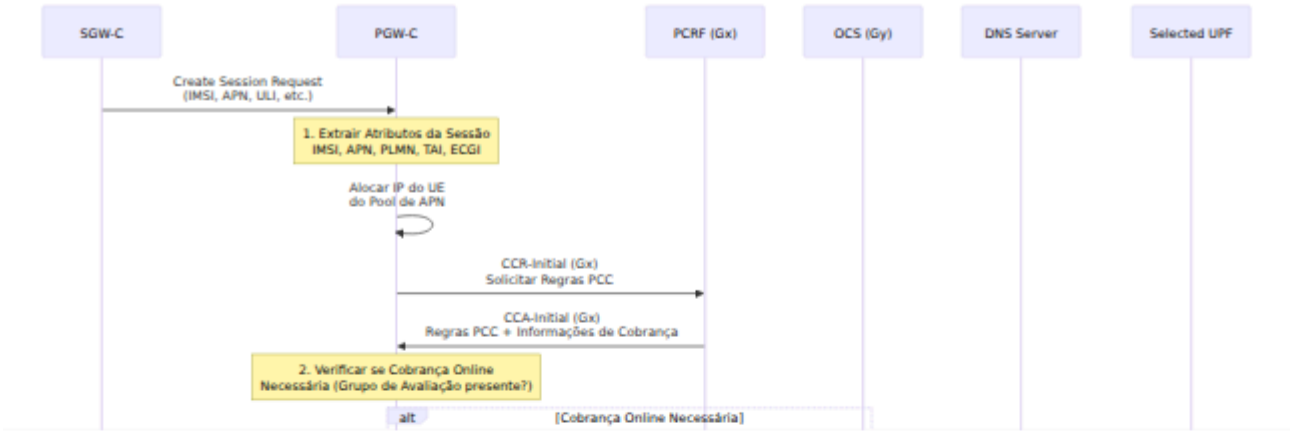
Sobrescrições de PCO (Opções de Configuração de Protocolo):

Cada regra de seleção de UPF pode opcionalmente especificar valores de PCO personalizados que sobrescrevem a configuração padrão de PCO para sessões correspondentes. Isso permite que diferentes APNs ou tipos de tráfego recebam diferentes parâmetros de rede.

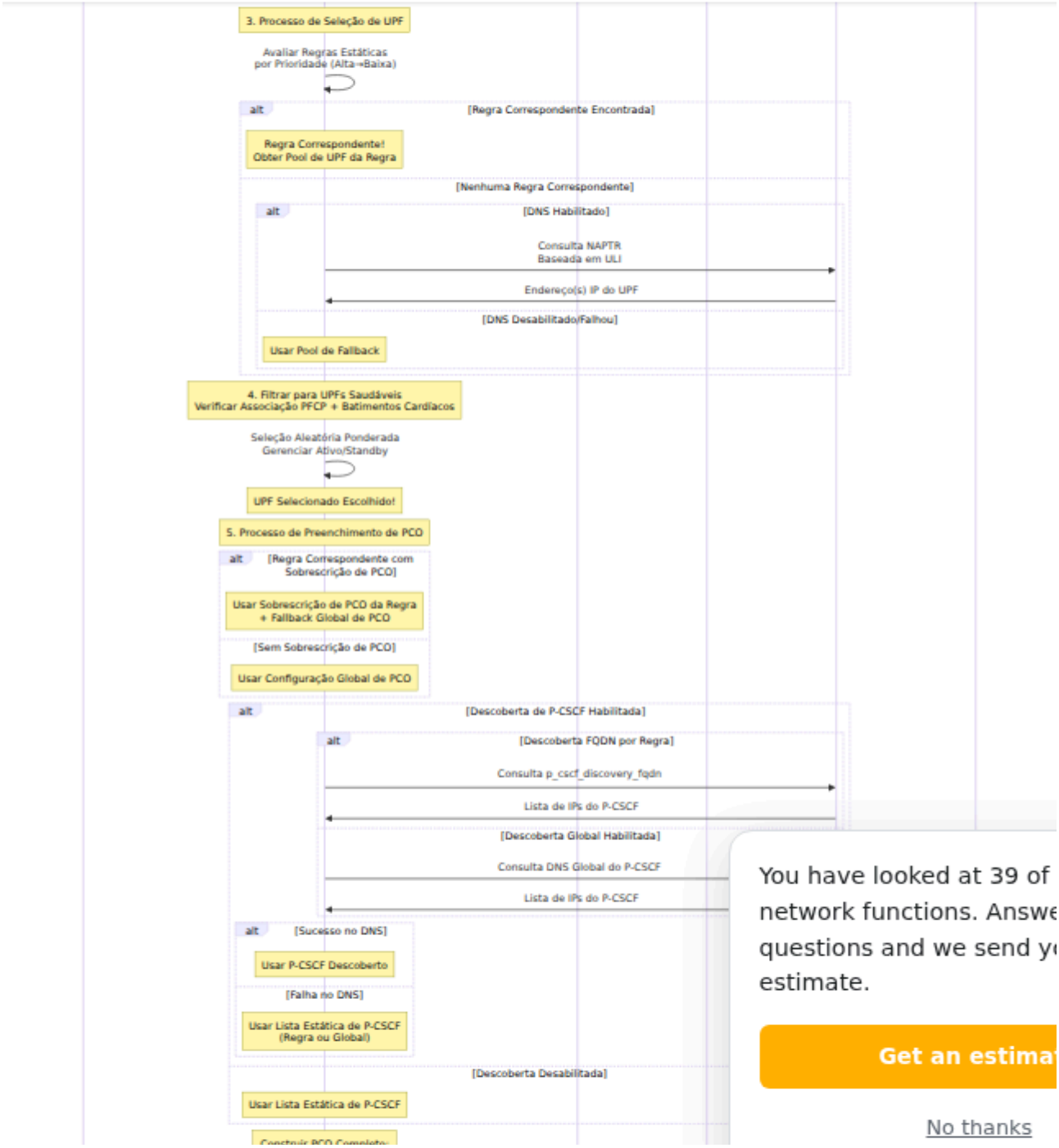
Como Funcionam as Sobrescrições de PCO:

1. **Sobrescrições Parciais**: Especifique apenas os campos de PCO que você deseja sobrescrever
2. **Fallback Padrão**: Campos não especificados usam valores da configuração principal de `pco`
3. **Específico da Regra**: Cada regra pode ter diferentes sobrescrições de PCO
4. **Mesclagem de Prioridade**: PCO da regra tem prioridade sobre o PCO padrão

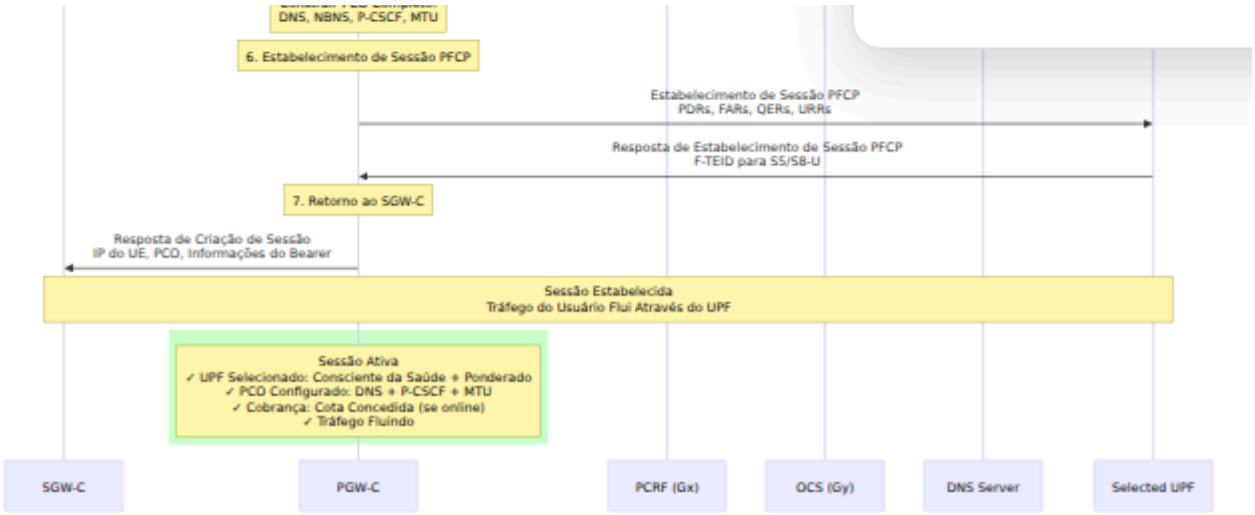
Hierarquia de Preenchimento de PCO



OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
Português



You have looked at 39 of
network functions. Answe
questions and we send y
estimate.
Get an estima
No thanks



Ordem de Prioridade para Cada Campo de PCO:

1. **Sobrescrição de PCO da Regra** (Maior Prioridade)
2. **Descoberta DNS do P-CSCF** (apenas para endereços do P-CSCF)
3. **Configuração Global de PCO** (Menor Prioridade / Fallback)

Exemplo: Regra IMS Sobrescreve DNS, Regra Empresarial Sobrescreve Tudo

```
Sessão IMS (correspondente à regra "Tráfego IMS"):
├─ Servidores DNS: DA GLOBAL (não sobrescrito na regra)
├─ P-CSCF: DA DESCOBERTA DNS (p_cscf_discovery_fqdn definido na
regra)
│  └─ Fallback: DA REGRA se DNS falhar
└─ MTU: DA GLOBAL (não sobrescrito na regra)

Sessão Empresarial (correspondente à regra "Tráfego Empresarial"):
├─ Servidores DNS: DA REGRA (192.168.1.10, 192.168.1.11)
├─ P-CSCF: DA GLOBAL (não sobrescrito na regra)
└─ MTU: DA REGRA (1500)

Sessão Padrão (nenhuma regra correspondente):
├─ Servidores DNS: DA GLOBAL
├─ P-CSCF: DA GLOBAL ou DNS se descoberta global habilitada
└─ MTU: DA GLOBAL
```

Campos de Sobrescrição de PCO Disponíveis:

- `primary_dns_server_address` - Endereço IP do servidor DNS principal (IPv4)
- `secondary_dns_server_address` - Endereço IP do servidor DNS secundário (IPv4)
- `primary_dns_server_ipv6_address` - Endereço IP do servidor DNS principal (IPv6, container 0x0003)
- `secondary_dns_server_ipv6_address` - Endereço IP do servidor DNS secundário (IPv6, container 0x0003)
- `primary_nbns_server_address` - Endereço IP do servidor WINS principal
- `secondary_nbns_server_address` - Endereço IP do servidor WINS secundário
- `p_cscf_ipv4_address_list` - Lista de IPs de servidores P-CSCF IPv4 (para IMS, container 0x000C) - Veja Configuração de PCO e Monitoramento do P-CSCF para descoberta dinâmica do P-CSCF
- `p_cscf_ipv6_address_list` - Lista de IPs de servidores P-CSCF IPv6 (para IMS, container 0x0001)
- `ipv4_link_mtu_size` - Tamanho do MTU em bytes

Descoberta do P-CSCF por Regra:

Além das sobrescrições de PCO, as regras de seleção de UPF podem especificar descoberta dinâmica do P-CSCF:

- `p_cscf_discovery_fqdn` - (String) FQDN para descoberta do P-CSCF baseada em DNS (ex: `"pcscf.mnc001.mcc001.3gppnetwork.org"`)

Quando este parâmetro está definido:

1. O PGW-C realiza a consulta DNS para o FQDN especificado durante o estabelecimento da sessão
2. O servidor DNS retorna uma lista de endereços IP do P-CSCF
3. Os endereços P-CSCF descobertos são enviados para o UE via PCO
4. Se a consulta DNS falhar, volta para `p_cscf_ipv4_address_list` da substituição do PCO (se especificado) ou configuração global do PCO
5. Veja Monitoramento do P-CSCF para monitorar as taxas de sucesso/falha da descoberta

Isso é particularmente útil para:

- **APNs IMS** - Diferentes redes IMS com diferentes servidores P-CSCF
- **Implantações multi-inquilino** - Diferentes empresas com infraestrutura P-CSCF dedicada
- **Roteamento geográfico** - O DNS retorna o P-CSCF mais próximo com base na localização do UE
- **Alta disponibilidade** - O DNS retorna automaticamente apenas servidores P-CSCF saudáveis

Exemplo: Tráfego IMS com P-CSCF Personalizado:

```
rules: [
  %{
    name: "Tráfego IMS",
    priority: 20,
    match_field: :apn,
    match_regex: "^ims",
    upf_pool: [
      %{remote_ip_address: "10.100.2.21", remote_port: 8805,
weight: 80},
      %{remote_ip_address: "10.100.2.22", remote_port: 8805,
weight: 20}
    ],
    # Descoberta do P-CSCF: Consulta dinamicamente o DNS para
endereços P-CSCF
    # A consulta DNS retorna os IPs P-CSCF atuais com base neste
FQDN
    p_cscf_discovery_fqdn: "pcscf.mnc001.mcc001.3gppnetwork.org",
    # As sessões IMS recebem servidores P-CSCF personalizados
(usados como fallback se o DNS falhar)
    pco: %{
      p_cscf_ipv4_address_list: ["203.0.113.100", "203.0.113.101"]
      # DNS, NBNS, MTU usarão padrões da configuração principal do
PCO
    }
  }
]
```

Alocação de IP do UE por Regra:

Cada regra de seleção UPF escolhe **como o IP do UE é alocado**. Existem três cenários mutuamente consistentes:

Cenário	Configuração da regra	Quem aloca	Pool usado
1. Padrão	*(nenhum campo)*	PGW-C (endereço enviado para o UPF)	`ue.subnet_map` baseado em APN, voltando para `:default`
2. Pool nomeado	`ue_ip_pool: "name"`	PGW-C (endereço enviado para o UPF)	o pool nomeado `ue.subnet_map` `"name"`
3. Delegado ao UPF	`upf_assigns_ip: true`	UPF (PFCP CHV4)	um pool no UPF

- `upf_assigns_ip` - (Booleano, padrão `false`) Quando `true`, o PGW-C envia o IE UE-IP-Address com a flag **CHV4** ("UP aloca") em vez de um endereço explícito, e lê o endereço alocado de volta do PDR Criado do UPF na Resposta de Estabelecimento de Sessão. **Qual** pool o UPF utiliza é uma preocupação totalmente do lado do UPF (o UPF pode baseá-lo na APN / Instância de Rede PFCP se ele tiver vários pools) — nada sobre o pool é sinalizado do PGW-C.
- `ue_ip_pool` - (String, opcional) Nome de um pool `ue.subnet_map` do **PGW-C** alocado quando esta regra é selecionada. Isso vincula um pool do PGW-C a um UPF via a regra de seleção (por exemplo, para que o UPF que possui/anuncia `100.64.20.0/24` só receba endereços desse pool). Um pool pode ser referenciado por uma regra ou muitas. `nil` → o comportamento padrão do `ue.subnet_map` baseado em APN (Cenário 1).

`upf_assigns_ip` e `ue_ip_pool` são **mutuamente exclusivos** — um delega a alocação para o UPF, o outro a mantém no PGW-C. Se uma regra define ambos,

a configuração registra um aviso e `upf_assigns_ip` vence (`ue_ip_pool` é ignorado).

Ou a alocação do PGW-C de um pool vinculado ao UPF (Cenário 2) *ou* a delegação do UPF (Cenário 3) permite que cada UPF possua uma sub-rede e anuncie um único **agregado** OSPF (por exemplo, `100.64.20.0/24`) em vez de um `/32` por UE por anexo — o agregado está presente desde a inicialização do UPF, então o downlink de um UE recém-conectado é roteável imediatamente, removendo a janela de buraco negro de convergência de rota por UE.

> **Importante — alinhamento de pool/UPF:** o pool do qual um UE é alocado deve estar dentro do agregado anunciado do UPF que o transporta. Para o Cenário 2, vincule cada pool por UPF a uma regra cujo `upf_pool` seja aquele único UPF. Para o Cenário 3, cada UPF `upf_pool` deve ser configurado com seu próprio `ueip_pool` não sobreposto (omniupf `feature_ueip: true`) anunciando aquele agregado (`route_advertise_pools`). Se um UE aterrissar em um UPF, mas seu endereço cair dentro do agregado de outro UPF, ocorre buracos negros no downlink.

Exemplo — Cenário 2 (PGW-C aloca de pools por UPF vinculados pela regra):

```
# ue.subnet_map define os pools nomeados:
ue: %{
  subnet_map: %{
    "ims_upf03" => ["100.64.20.0/24"],
    "ims_upf04" => ["100.64.21.0/24"],
    default:     ["100.64.16.0/24"]
  }
},
# cada regra fixa um UPF e seu pool:
upf_selection: %{
  rules: [
    %{name: "IMS→upf03", priority: 21, match_field: :apn,
match_regex: "^ims",
      upf_pool: [%{remote_ip_address: "10.179.1.42", remote_port:
8805, weight: 1}],
      ue_ip_pool: "ims_upf03"},
    %{name: "IMS→upf04", priority: 20, match_field: :apn,
match_regex: "^ims",
      upf_pool: [%{remote_ip_address: "10.179.1.45", remote_port:
8805, weight: 1}],
      ue_ip_pool: "ims_upf04"}
  ]
}
```

Exemplo — Cenário 3 (UPF aloca o IP do UE via CHV4):

```
rules: [
  %{
    name: "IMS",
    priority: 20,
    match_field: :apn,
    match_regex: "^ims",
    upf_pool: [
      %{remote_ip_address: "10.179.1.42", remote_port: 8805,
weight: 50},
      %{remote_ip_address: "10.179.1.45", remote_port: 8805,
weight: 50}
    ],
    upf_assigns_ip: true   # UPF aloca o IP do UE (CHV4); pool é
do lado do UPF
  }
]
```

Exemplo: Tráfego Empresarial com DNS Personalizado:

```
rules: [
  %{
    name: "Tráfego Empresarial",
    priority: 15,
    match_field: :apn,
    match_regex: "^(enterprise|corporate)\.apn",
    upf_pool: [
      %{remote_ip_address: "10.100.3.21", remote_port: 8805,
weight: 100}
    ],
    # Sessões empresariais recebem DNS corporativo e MTU
personalizado
    pco: %{
      primary_dns_server_address: "192.168.1.10",
      secondary_dns_server_address: "192.168.1.11",
      ipv4_link_mtu_size: 1500
      # P-CSCF, NBNS usarão padrões da configuração principal do
PCO
    }
  }
]
```

Exemplo: Substituição Completa (Todos os Campos do PCO):

```
rules: [
  %{
    name: "IoT APN - Totalmente Personalizado",
    priority: 10,
    match_field: :apn,
    match_regex: "^iot\.m2m",
    upf_pool: [
      %{remote_ip_address: "10.100.5.21", remote_port: 8805,
weight: 100}
    ],
    # Sessões IoT recebem PCO totalmente personalizado
    pco: %{
      primary_dns_server_address: "8.8.8.8",
      secondary_dns_server_address: "8.8.4.4",
      primary_nbns_server_address: "10.0.0.100",
      secondary_nbns_server_address: "10.0.0.101",
      p_cscf_ipv4_address_list: [],  # Sem P-CSCF para IoT
      ipv4_link_mtu_size: 1280       # MTU menor para dispositivos
restritos
    }
  }
]
```

Casos de Uso:

- **IMS/VoLTE**: Fornecer servidores P-CSCF específicos do operador para serviços de voz
- **APNs Empresariais**: Roteamento de tráfego corporativo através de servidores DNS da empresa
- **IoT/M2M**: Usar DNS público e MTU otimizado para dispositivos de baixa largura de banda
- **Roaming**: Fornecer servidores DNS locais para assinantes visitantes
- **Diferenciação de Serviços**: Diferentes parâmetros de rede por tipo de serviço

Seleção de UPF Baseada em DNS:

Ative a seleção dinâmica de UPF com base nas Informações de Localização do Usuário (ULI) usando consultas NAPTR DNS. As configurações de DNS agora são configuradas na seção `upf_selection`.

Nota: Isso fornece seleção de UPF baseada em geografia ou topologia. Veja Interface PFCP para configuração de associação PFCP com UPFs descobertos dinamicamente e Gerenciamento de Sessão para fluxos de estabelecimento de sessão.

```
upf_selection: %{
  # Habilitar seleção baseada em DNS
  dns_enabled: true,

  # Tipos de localização a serem consultados em ordem de
prioridade
  dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],

  # Sufixo DNS para consultas NAPTR 3GPP
  dns_suffix: "epc.3gppnetwork.org",

  # Tempo limite da consulta DNS em milissegundos
  dns_timeout_ms: 5000,

  # ... regras e fallback_pool ...
}
```

A seleção baseada em DNS funciona da seguinte forma:

1. **Prioridade**: A seleção DNS é usada apenas quando **NENHUMA regra estática corresponde** (prioridade mais baixa)
2. **Geração de Consulta**: Constrói consultas NAPTR DNS com base na localização do UE:
   - Consulta ECGI: `eci-<hex>.ecgi.epc.mnc<MNC>.mcc<MCC>.epc.3gppnetwork.org`
   - Consulta TAI: `tac-lb<hex>.tac-hb<hex>.tac.epc.mnc<MNC>.mcc<MCC>.epc.3gppnetwork.org`
   - Consultas RAI, SAI, CGI seguem um formato semelhante ao 3GPP TS 23.003

3. **Hierarquia de Fallback**: Tenta cada tipo de localização em ordem de prioridade até que uma correspondência seja encontrada
4. **Correspondência de Pares**: Os resultados DNS são filtrados contra a lista de pares configurada
5. **Seleção**: Escolhe o par correspondente (atualmente a primeira correspondência, seleção baseada em carga em breve)

Exemplo de Registros DNS (configure em seu servidor DNS):

```
; Registro NAPTR para TAC 100 no PLMN 001-001
tac-lb64.tac-hb00.tac.epc.mnc001.mcc001.epc.3gppnetwork.org IN
NAPTR 10 50 "a" "x-3gpp-upf:x-sxb" "" upf-edge-1.example.com.

; Registro A para o UPF
upf-edge-1.example.com IN A 10.100.1.21
```

Casos de Uso:

- **Computação de Borda Multi-Acesso (MEC)**: Roteie sessões para os UPFs de borda mais próximos geograficamente
- **Descoberta Dinâmica de UPF**: Adicione/remova UPFs sem reconfigurar o PGW-C
- **Balanceamento de Carga**: Distribua a carga entre UPFs com base na localização
- **Fatiamento de Rede**: Roteie diferentes fatias para diferentes UPFs por localização

Monitoramento de Saúde do UPF

Seleção Automática Consciente da Saúde: O PGW-C monitora continuamente a saúde de todos os UPFs e exclui automaticamente UPFs não saudáveis da seleção.

Critérios de Verificação de Saúde

Um UPF é considerado **saudável** quando TODAS as seguintes condições são atendidas:

1. **Associação PFCP Ativa**: O UPF tem uma associação PFCP estabelecida
2. **Responsividade do Heartbeat**: Menos de 3 batimentos cardíacos consecutivos perdidos
3. **Processo Vivo**: O processo GenServer do par UPF está em execução

Um UPF é considerado **não saudável** se QUALQUER um dos seguintes for verdadeiro:

- A associação PFCP não está estabelecida (`associated: false`)
- 3 ou mais timeouts consecutivos de heartbeat
- O processo do par UPF falhou ou não está respondendo

Mecanismo de Monitoramento

Para UPFs Configurados (em `upf_selection`):

- O rastreamento de saúde começa imediatamente na inicialização
- A associação PFCP é monitorada continuamente
- Batimentos cardíacos são enviados a cada 5 segundos
- O contador `missed_heartbeats_consecutive` rastreia falhas consecutivas
- Todos os UPFs das regras e do pool de fallback são registrados automaticamente

Para UPFs Descobertos por DNS (registro dinâmico):

- Presumido saudável até a primeira tentativa de sessão
- Registrado automaticamente no primeiro uso
- O rastreamento de saúde começa após o registro

Comportamento de Seleção

Modo Ativo/Standby (ao usar `weight: 0`):

1. Filtra apenas UPFs saudáveis
2. Separa em **ativo** (peso > 0) e **standby** (peso == 0)
3. Usa UPFs ativos se algum estiver saudável
4. Ativa UPFs standby (trata como peso 1) se todos os ativos estiverem não saudáveis

5. Volta para o pool completo se não houver UPFs saudáveis

Modo Balanceado por Carga (todos os pesos > 0):

1. Filtra apenas UPFs saudáveis
2. Realiza seleção aleatória ponderada entre UPFs saudáveis
3. Volta para o pool completo se não houver UPFs saudáveis

Registro:

```
[debug] Usando pool de UPF ativo (2/3 UPFs saudáveis, 1 standby)
[info] Todos os UPFs ativos estão inativos, ativando UPFs standby
(1 UPFs standby, tratando peso 0 como 1)
[warning] Nenhum UPF saudável no pool (3 no total), usando pool
completo como fallback
```

Verificando a Saúde do UPF

Programaticamente:

```
# Verifica se um UPF específico está saudável
iex> PGW_C.PFCP_Node.is_peer_healthy?({10, 100, 1, 21})
true

# Obtém informações detalhadas de saúde
iex> PGW_C.PFCP_Node.get_peer_health({10, 100, 1, 21})
%{
  associated: true,
  missed_heartbeats: 0,
  healthy: true,
  registered: true
}
```

Via API OAM:

- Consulte o endpoint `/api/upf_selection`: `curl -k https://localhost:8443/api/upf_selection`
- Retorna o status de saúde em tempo real para todos os UPFs em cada pool
- Valores de status: Active-UP, Standby-Ready, Active-DOWN, Not Associated

- Valores de função: ACTIVE (peso > 0), STANDBY (peso == 0), DYNAMIC (descoberto por DNS, não na configuração)
- Contador de misses de heartbeat relatado para UPFs associados

Melhores Práticas de Monitoramento de Saúde

1. **Configure UPFs em upf_selection**: Todos os UPFs nas regras e pools de fallback são monitorados automaticamente

```
upf_selection: %{
  rules: [
    %{
      name: "Tráfego de Internet",
      priority: 10,
      match_field: :apn,
      match_regex: "^internet",
      upf_pool: [
        %{remote_ip_address: "10.100.1.21", remote_port: 8805,
weight: 100}
      ]
    }
  ],
  fallback_pool: [
    %{remote_ip_address: "10.100.2.21", remote_port: 8805,
weight: 100}
  ]
}
# Todos os UPFs automaticamente recebem:
# - Batimentos cardíacos a cada 5 segundos
# - Monitoramento de saúde desde a inicialização
# - Nomes gerados automaticamente
```

2. **Use UPFs standby**: Configure standbys quentes com `weight: 0` para failover automático

```
upf_pool: [
  %{remote_ip_address: "10.1.1.1", remote_port: 8805, weight:
100},  # Ativo
  %{remote_ip_address: "10.1.1.2", remote_port: 8805, weight:
0}     # Standby
]
```

3. **Monitore via API OAM**: Consulte regularmente o endpoint `/api/upf_selection` para verificar o status de saúde do UPF

4. **Monitoramento de heartbeat**: O sistema usa um limite fixo de 3 batimentos cardíacos consecutivos perdidos para determinar a saúde do par.

Registro Dinâmico de UPF

Recurso: O PGW-C registra e monitora automaticamente UPFs descobertos através do DNS, mesmo que não estejam na configuração de `upf_selection`.

Como Funciona

Quando qualquer método de seleção (regras estáticas, pools ou DNS) retorna um UPF que ainda não está registrado, o sistema automaticamente:

1. **Cria um Par PFCP**: Gera uma configuração de par padrão para o UPF desconhecido
2. **Inicia a Associação PFCP**: Tenta estabelecer uma associação PFCP com o UPF
3. **Registra no Registro de Pares**: Adiciona o UPF ao sistema interno de rastreamento de pares
4. **Inicia o Monitoramento de Batimentos**: Começa trocas periódicas de batimentos (intervalos de 10 segundos)
5. **Rastreia a Vida**: Monitora o UPF para falhas e recuperação

Configuração Padrão para UPFs Dinamicamente Registrados

Quando um UPF é registrado dinamicamente, ele recebe a seguinte configuração padrão:

```
%{
  name: "Dynamic-UPF-<IP>",              # por exemplo, "Dynamic-
UPF-10-100-1-21"
  remote_ip_address: <discovered_ip>,     # IP do DNS ou seleção
  remote_port: 8805,                      # Porta PFCP padrão
(substituível)
  initiate_pfcp_association_setup: true,  # PGW-C inicia a
associação
  heartbeat_period_ms: 10_000             # Intervalo de batimento
de 10 segundos
}
```

> **Nota**: UPFs dinâmicos são registrados apenas para gerenciamento de associação. Eles são usados como alvos nas regras de `upf_selection`, não como fontes de lógica de seleção.

Exemplo: DNS Retorna UPF Desconhecido

```
# A consulta DNS retorna: upf-edge-2.example.com -> 10.200.5.99
# Este UPF NÃO está na sua configuração de upf_selection

# Fluxo de registro dinâmico:
# 1. O sistema detecta o UPF desconhecido 10.200.5.99
# 2. Registra: "UPF {10, 200, 5, 99} não pré-configurado, tentando
registro dinâmico..."
# 3. Envia Solicitação de Configuração de Associação PFCP para
10.200.5.99:8805
# 4. Se o UPF responder: Associação estabelecida, sessão continua
normalmente
# 5. Se o UPF não responder: A sessão falha graciosamente com uma
mensagem de erro clara
```

Benefícios

 Descoberta Dinâmica Verdadeira: A seleção de UPF baseada em DNS agora funciona sem pré-configuração
 Escalonamento Automático: Adicione UPFs à sua rede sem reiniciar o PGW-C
 Degradação Graciosa: Se a associação falhar, as sessões falham de forma limpa (sem falhas)

 Compatível com Versões Anteriores: UPFs pré-configurados continuam a funcionar exatamente como antes
 Monitoramento Completo: UPFs dinâmicos recebem o mesmo monitoramento de batimentos que pares estáticos

Tratamento de Falhas

Se um UPF descoberto dinamicamente não responder à Solicitação de Associação PFCP:

```
[error] A configuração da associação PFCP falhou para o UPF
dinâmico {10, 200, 5, 99}: :timeout
[error] Falha ao registrar dinamicamente o UPF {10, 200, 5, 99}:
:timeout.
        A criação da sessão falhará. Considere adicionar este UPF
à configuração de upf_selection.
```

A criação da sessão falhará, mas o PGW-C permanece estável e continua processando outras sessões.

Quando Pré-Configurar vs. Registro Dinâmico

Cenário	Recomendação
UPFs de Núcleo de Produção	Pré-configurar em `upf_selection` (configuração explícita, monitorada desde a inicialização)
UPFs de Borda Descobertos por DNS	Usar registro dinâmico (escalona automaticamente com a infraestrutura)
UPFs de Teste/Desenvolvimento	Qualquer abordagem funciona (dinâmica é mais conveniente)
UPFs Críticos para a Missão	Pré-configurar em `upf_selection` (garante monitoramento desde a inicialização)
UPFs Efêmeros/Auto-Escalados	Usar registro dinâmico (UPFs vêm e vão dinamicamente)

Monitoramento de UPFs Dinâmicos

UPFs dinâmicos aparecem nos logs com seus nomes gerados automaticamente:

```
[info] Criando configuração de par PFCP dinâmica para Dynamic-UPF-10-200-5-99 ({10, 200, 5, 99}:8805)
[info] Par UPF dinâmico Dynamic-UPF-10-200-5-99 registrado com sucesso com PID #PID<0.1234.0>
```

Você pode consultar o registro de pares para ver todos os pares registrados (tanto estáticos quanto dinâmicos):

```
# Obter todos os pares registrados
PGW_C.PFCP_Node.registered_peer_count()

# Verificar se um UPF específico está registrado
PGW_C.PFCP_Node.get_peer({10, 200, 5, 99})
# Retorna: {:ok, #PID<0.1234.0>} se registrado, :error caso
contrário
```

Porta Personalizada para UPFs Dinâmicos

Se seus UPFs usam uma porta PFCP não padrão, você pode acionar manualmente o registro:

```
# Registrar UPF na porta personalizada
PGW_C.PFCP_Node.register_dynamic_peer({10, 200, 5, 99}, 9999)
```

No entanto, a seleção baseada em DNS e o registro automático sempre usam a porta 8805 (porta PFCP padrão).

Modo de Execução a Seco da Seleção de UPF:

Teste e valide sua configuração de seleção de UPF sem afetar sessões reais:

```
config :pgw_c,
  # Habilitar modo a seco para testes (desativado por padrão)
  upf_selection_dry_run: true
```

Quando o modo a seco está habilitado:

1. **Nenhuma Atribuição Real**: Sessões não são realmente atribuídas a UPFs
2. **Registro Detalhado**: Decisões de seleção são registradas com detalhes completos
3. **Retorno de Erro**: `assign_sxb_peer/1` retorna `{:error, :dry_run_mode}`
4. **Prevenção de Sessão**: Retornar um erro impede a criação de sessão
5. **Ambos os Métodos**: Funciona com regras estáticas e seleção baseada em DNS

Exemplo de Saída de Log:

```
[warning] ⚠  MODO DE EXECUÇÃO A SECO HABILITADO - Nenhuma
atribuição real ocorrerá

[info] □ EXECUÇÃO A SECO: Regra estática correspondente
  Método: Regra Estática
  Campo de Correspondência: :apn
  Expressão Regular de Correspondência: ~r/^internet\./
  Prioridade: 10
  UPF Selecionado: 10.0.1.21:8805

[warning] □ EXECUÇÃO A SECO: Atribuição de UPF seria feita, mas
pulando a atribuição real
  IMSI da Sessão: 001001000000670
  APN da Sessão: internet.apn
  Método de Seleção: estático
  Ligaria a: 10.0.1.21:8805
  ⚠  Retornando erro para impedir a criação da sessão
```

Testando via Interface LiveView:

A página LiveView de Seleção de UPF (`/api/upf_selection`) inclui uma interface de teste interativa:

1. Navegue até a página de Seleção de UPF no painel da web
2. Role até a seção "Testar Seleção de UPF"
3. Insira atributos de sessão de teste:
   - IMSI (por exemplo, `001001000000670`)
   - APN (por exemplo, `internet.apn`)
   - ID PLMN da Rede Servidora (por exemplo, `001001`)
4. Clique em "Testar Seleção" para simular a seleção
5. Veja os resultados detalhados mostrando:
   - Qual regra correspondeu (ou se o DNS seria usado)
   - UPF selecionado e nome do par
   - Detalhes do campo de correspondência e padrão
   - Prioridade da regra

A interface de teste LiveView simula a lógica de seleção **sem exigir que o modo a seco** seja habilitado globalmente, tornando seguro testar em ambientes de produção sem afetar o tráfego real.

Casos de Uso:

- **Teste de Configuração**: Valide regras de roteamento antes de implantar em produção
- **Solução de Problemas**: Entenda por que sessões específicas são roteadas para UPFs específicos
- **Treinamento**: Demonstre a lógica de seleção de UPF para equipes de operações
- **Desenvolvimento**: Teste novas regras de seleção durante o desenvolvimento

Campos de Correspondência:

- `:imsi` - Identidade de Assinante Móvel Internacional
- `:apn` - Nome do Ponto de Acesso (APN/DNN)
- `:serving_network_plmn_id` - ID PLMN da rede servidora
- `:sgw_ip_address` - Endereço IP do SGW
- `:uli_tai_plmn_id` - ID PLMN da Área de Rastreamento
- `:uli_ecgi_plmn_id` - ID PLMN da Célula E-UTRAN

Ajuste de Heartbeat:

```
# Agressivo (detectar falhas rapidamente)
heartbeat_period_ms: 2_000  # 2 segundos

# Padrão (recomendado)
heartbeat_period_ms: 5_000  # 5 segundos

# Relaxado (redes de alta latência)
heartbeat_period_ms: 10_000  # 10 segundos
```

Veja: Documentação da Interface PFCP

Configuração do Pool de IP do UE

Propósito

Configurar pools de endereços IP para alocação de UE, organizados por APN.

Bloco de Configuração

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      # Pools APN "internet"
      "internet" => [
        "100.64.0.0/20"  # 4094 IPs utilizáveis
      ],

      # Pools APN "ims"
      "ims" => [
        "100.64.16.0/22"  # 1022 IPs utilizáveis
      ],

      # Pool padrão para APNs desconhecidos
      default: [
        "42.42.42.0/24"  # 254 IPs utilizáveis
      ]
    }
  }
```

Parâmetros

Parâmetro	Tipo	Necessário	Descrição
`subnet_map`	Mapa	Sim	Mapeia nomes de APN para listas de sub-rede
`default`	Lista (Sub-redes)	Sim	Pool de fallback para APNs desconhecidos

Formato da Sub-rede

Notação CIDR: `<endereço_rede>/<comprimento_prefixo>`

Cálculo de IP Utilizáveis:

CIDR	Total de IPs	IPs Utilizáveis	Faixa de Exemplo
/24	256	254	100.64.1.1 - 100.64.1.254
/23	512	510	100.64.0.1 - 100.64.1.254
/22	1024	1022	100.64.0.1 - 100.64.3.254
/21	2048	2046	100.64.0.1 - 100.64.7.254
/20	4096	4094	100.64.0.1 - 100.64.15.254
/16	65536	65534	100.64.0.1 - 100.64.255.254

Exemplos

Configuração Simples:

```
ue: %{
  subnet_map: %{
    "internet" => ["100.64.1.0/24"],
    default: ["42.42.42.0/24"]
  }
}
```

Configuração de Produção:

```
ue: %{
  subnet_map: %{
    # Internet geral - grande pool
    "internet" => [
      "100.64.0.0/18"  # 16.382 IPs
    ],

    # IMS (VoLTE)
    "ims" => [
      "100.64.64.0/22"  # 1.022 IPs
    ],

    # APN Empresarial
    "enterprise.corp" => [
      "10.100.0.0/16"  # 65.534 IPs
    ],

    # Dispositivos IoT
    "iot.m2m" => [
      "100.64.72.0/20"  # 4.094 IPs
    ],

    # Fallback padrão
    default: [
      "42.42.42.0/24"  # 254 IPs
    ]
  }
}
```

Balanceamento de Carga (Múltiplas Sub-redes por APN):

```
ue: %{
  subnet_map: %{
    "internet" => [
      "100.64.0.0/22",   # 1022 IPs
      "100.64.4.0/22",   # 1022 IPs
      "100.64.8.0/22",   # 1022 IPs
      "100.64.12.0/22"   # 1022 IPs
    ],
    # Total: 4088 IPs, distribuídos aleatoriamente
    default: ["42.42.42.0/24"]
  }
}
```

Correspondência de Padrão Regex:

Para correspondência de APN com curinga, use chaves que começam com `^` para indicar padrões regex:

```
ue: %{
  subnet_map: %{
    # Regex: APNs começando com "ims" (corresponde a "ims",
"ims.apn", "ims.something")
    "^ims" => [
      "100.64.10.0/24"
    ],

    # Regex: APNs começando com "m2m." (corresponde a "m2m.test",
"m2m.prod")
    "^m2m\." => [
      "100.64.20.0/24"
    ],

    # Correspondência exata apenas
    "enterprise.corp" => [
      "10.100.0.0/16"
    ],

    default: ["42.42.42.0/24"]
  }
}
```

Regras de Padrão:

- Chaves que começam com `^` são padrões regex
- Chaves sem `^` são correspondências exatas (compatibilidade retroativa)
- Padrões regex são compilados no momento da análise da configuração
- O primeiro padrão correspondente vence
- As barras invertidas devem ser escapadas (`\` para `\`)

Padrões Comuns:

- **Começa com:** `"^ims"` corresponde a `ims`, `ims.apn`, `ims.foo.bar`
- **Termina com:** `"^.*\.corp$"` corresponde a `foo.corp`, `bar.corp`
- **Contém:** `"^.*test.*"` corresponde a `test`, `foo.test.bar`

Suporte a IPv6:

```
ue: %{
  subnet_map: %{
    # Pools IPv4
    "internet" => [
      "100.64.0.0/20"
    ],

    # Pools IPv6 (delegação de prefixo)
    "internet.ipv6" => [
      "2001:db8:1::/48"
    ],

    default: [
      "42.42.42.0/24"
    ]
  }
}
```

Faixas de IP Recomendadas

RFC 6598 (NAT de Grau de Operadora):

- **Faixa:** `100.64.0.0/10`

- **Tamanho:** ~4 milhões de IPs
- **Propósito:** Projetado para NAT de provedores de serviço

Faixas de IP Privadas (RFC 1918):

- `10.0.0.0/8` - 16 milhões de IPs
- `172.16.0.0/12` - 1 milhão de IPs
- `192.168.0.0/16` - 65.534 IPs

Veja: Documentação de Alocação de IP do UE

Configuração do PCO

Propósito

Configurar Opções de Configuração de Protocolo (PCO) enviadas para o UE.

Bloco de Configuração

```
config :pgw_c,
  pco: %{
    # Servidores DNS
    primary_dns_server_address: "8.8.8.8",
    secondary_dns_server_address: "8.8.4.4",

    # Servidores DNS IPv6 (opcional, contêiner 0x0003)
    primary_dns_server_ipv6_address: nil,
    secondary_dns_server_ipv6_address: nil,

    # Servidores NBNS (opcional, para dispositivos Windows)
    primary_nbns_server_address: nil,
    secondary_nbns_server_address: nil,

    # Endereços P-CSCF para IMS
    p_cscf_ipv4_address_list: ["10.0.0.50", "10.0.0.51"],   #
contêiner 0x000C
    p_cscf_ipv6_address_list: [],                           #
contêiner 0x0001

    # Tamanho MTU IPv4
    ipv4_link_mtu_size: 1400
  }
```

Parâmetros

Parâmetro	Tipo	Padrão	Descrição
`primary_dns_server_address`	String (IPv4)	Necessário	Servidor DNS primário
`secondary_dns_server_address`	String (IPv4)	Opcional	Servidor DNS secundário
`primary_dns_server_ipv6_address`	String (IPv6)	`nil`	Servidor DNS IPv6 primário (contêiner 0x0003)
`secondary_dns_server_ipv6_address`	String (IPv6)	`nil`	Servidor DNS IPv6 secundário (contêiner 0x0003)
`primary_nbns_server_address`	String (IPv4)	`nil`	Servidor NBNS (NetBIOS) primário
`secondary_nbns_server_address`	String (IPv4)	`nil`	Servidor NBNS secundário
`p_cscf_ipv4_address_list`	Lista (IPv4)	`[]`	Endereços IPv4 P-CSCF para IMS

Parâmetro	Tipo	Padrão	Descrição
			(contêiner 0x000C)
`p_cscf_ipv6_address_list`	Lista (IPv6)	`[]`	Endereços IPv6 P-CSCF para IMS (contêiner 0x0001)
`ipv4_link_mtu_size`	Inteiro	`1400`	Tamanho da Unidade Máxima de Transmissão

Exemplos

DNS Público (Google):

```
pco: %{
  primary_dns_server_address: "8.8.8.8",
  secondary_dns_server_address: "8.8.4.4",
  ipv4_link_mtu_size: 1400
}
```

DNS Privado:

```
pco: %{
  primary_dns_server_address: "10.0.0.10",
  secondary_dns_server_address: "10.0.0.11",
  ipv4_link_mtu_size: 1400
}
```

Configuração IMS:

```
pco: %{
  primary_dns_server_address: "10.0.0.10",
  secondary_dns_server_address: "10.0.0.11",

  # P-CSCF para IMS/VoLTE
  p_cscf_ipv4_address_list: [
    "10.0.0.50",  # P-CSCF Primário
    "10.0.0.51"   # P-CSCF Secundário
  ],

  ipv4_link_mtu_size: 1400
}
```

NBNS (Compatibilidade com Windows):

```
pco: %{
  primary_dns_server_address: "10.0.0.10",
  secondary_dns_server_address: "10.0.0.11",
  primary_nbns_server_address: "10.0.0.20",
  secondary_nbns_server_address: "10.0.0.21",
  ipv4_link_mtu_size: 1400
}
```

Ajuste de MTU:

```
# Ethernet padrão
ipv4_link_mtu_size: 1500

# Reduzido para sobrecarga de tunelamento
ipv4_link_mtu_size: 1400

# Jumbo frames (se suportado)
ipv4_link_mtu_size: 9000
```

Veja: Documentação de Configuração do PCO

Configuração da API OAM

Propósito

Configurar os endpoints da API REST OAM para gerenciar e monitorar o OmniPGW. Todo o monitoramento e gerenciamento é realizado através desta API.

> **Nota:** A configuração da API está ativa apenas em ambientes não de teste. A configuração é automaticamente ignorada quando `config_env()` é `:test`, `:test_mock`, ou `:test_impl`.

Configuração da API REST

```
config :api_ex,
  api: %{
    # Configurações de rede
    port: 8443,
    listen_ip: "0.0.0.0",

    # Metadados da API
    product_name: "PGW-C",
    title: "API - PGW-C",
    hostname: "localhost",

    # Configurações TLS
    enable_tls: true,
    tls_cert_path: "priv/cert/omnitouch.crt",
    tls_key_path: "priv/cert/omnitouch.pem",

    # Definições de rotas
    routes: [
      %{
        path: "/status",
        module: ApiEx.Api.StatusController,
        actions: [:index]
      }
    ]
  }
```

Parâmetros da API

Parâmetro	Tipo	Padrão	Descrição
`port`	Inteiro	`8443`	Porta HTTPS para a API REST
`listen_ip`	String (IP)	`"0.0.0.0"`	Endereço de escuta da API (0.0.0.0 = todas as interfaces)
`product_name`	String	`"PGW-C"`	Nome do produto (para metadados da API)
`title`	String	`"API - PGW-C"`	Título da documentação da API
`hostname`	String	`"localhost"`	Nome do host do servidor da API
`enable_tls`	Booleano	`true`	Habilitar HTTPS para a API
`tls_cert_path`	String (Caminho)	Necessário	Caminho para o arquivo de certificado SSL/TLS
`tls_key_path`	String (Caminho)	Necessário	Caminho para o arquivo de chave privada SSL/TLS
`routes`	Lista (Mapas)	`[]`	Definições de rotas da API

Formato de Definição de Rota

Cada rota na lista `routes` é um mapa com:

Campo	Tipo	Descrição	Exemplo
`path`	String	Caminho da URL para a rota	`"/status"`
`module`	Módulo	Módulo do controlador que manipula as solicitações	`ApiEx.Api.StatusController`
`actions`	Lista (Átomos)	Ações permitidas do controlador	`[:index, :show]`

Exemplos

Configuração Padrão (Desenvolvimento):

```
# API em localhost:8443
config :api_ex,
  api: %{
    port: 8443,
    listen_ip: "127.0.0.1",  # apenas localhost
    hostname: "localhost",
    enable_tls: true,
    tls_cert_path: "priv/cert/dev-cert.crt",
    tls_key_path: "priv/cert/dev-key.pem",
    routes: [
      %{path: "/status", module: ApiEx.Api.StatusController,
actions: [:index]}
    ]
  }
```

Configuração de Produção:

```
# API na interface de gerenciamento
config :api_ex,
  api: %{
    port: 8443,
    listen_ip: "10.0.0.20",  # Rede de gerenciamento
    product_name: "OmniPGW-C",
    hostname: "pgw-c-api.example.com",
    enable_tls: true,
    tls_cert_path: "/etc/ssl/certs/pgw-c-api.crt",
    tls_key_path: "/etc/ssl/private/pgw-c-api.key",
    routes: [
      %{path: "/status", module: ApiEx.Api.StatusController,
actions: [:index]},
      %{path: "/sessions", module: ApiEx.Api.SessionController,
actions: [:index, :show]}
    ]
  }
```

Acessando a API

Consultar um endpoint:

```
# Endpoint de status (certificado autoassinado, desenvolvimento)
curl -k https://localhost:8443/api/status

# Com um certificado confiável (produção)
curl https://pgw-c-api.example.com:8443/api/status
```

Todas as rotas são servidas sob o prefixo `/api`. Os endpoints disponíveis incluem `/api/status`, `/api/sessions`, `/api/upf`, `/api/diameter`, `/api/charging`, `/api/ip_pools`, `/api/session_history`, `/api/topology`, `/api/ue_search` (POST), `/api/pcscf_monitor`, e `/api/upf_selection`.

Documentação interativa da API (Swagger UI):

```
https://localhost:8443/api/docs
```

Configuração do Certificado TLS

Gerar Certificado Autoassinado (Desenvolvimento):

```
# Gerar chave privada e certificado
openssl req -x509 -newkey rsa:4096 -keyout priv/cert/omnitouch.pem
\
  -out priv/cert/omnitouch.crt -days 365 -nodes \
  -subj "/CN=localhost"
```

Certificado de Produção:

Para produção, use certificados de uma Autoridade Certificadora (CA) confiável:

- Let's Encrypt (gratuito, automatizado)
- CA Comercial (DigiCert, GlobalSign, etc.)
- CA Interna para implantações empresariais

Considerações de Segurança

1. **Sempre use TLS em produção** - Defina `enable_tls: true`
2. **Restringir listen_ip** - Use endereços IP específicos em produção (não 0.0.0.0)
3. **Use certificados válidos** - Evite certificados autoassinados em produção
4. **Proteção de firewall** - Restringir acesso à porta de gerenciamento (8443)
5. **Arquivos de chave fortes** - Use RSA de 4096 bits ou equivalente
6. **Atualizações regulares** - Rotacione certificados antes da expiração

Solução de Problemas

Problema: Não é possível acessar a API

```
# Verifique se a porta está escutando
netstat -tulpn | grep 8443

# Verifique se os arquivos de certificado existem
ls -la priv/cert/

# Verifique os logs em busca de erros de inicialização
tail -f /var/log/pgw_c/application.log
```

Problema: Erros SSL/TLS

- Verifique se os caminhos do certificado e da chave estão corretos
- Certifique-se de que o certificado corresponda ao nome do host
- Verifique a expiração do certificado: `openssl x509 -in cert.crt -text -noout`
- Verifique as permissões do arquivo da chave (deve ser legível pelo processo PGW-C)

Veja: Guia de Monitoramento para detalhes de uso da API OAM

Exemplo Completo

Configuração Pronta para Produção

```
# config/runtime.exs
import Config

# Configuração do Logger
config :logger, level: :info

config :pgw_c,
  # Métricas (Prometheus)
  metrics: %{
    enabled: true,
    ip_address: "10.0.0.20",  # Rede de gerenciamento
    port: 9090,
    registry_poll_period_ms: 5_000
  },

  # Diameter/Gx (interface PCRF)
  diameter: %{
    listen_ip: "0.0.0.0",
    host: "omnipgw.epc.mnc001.mcc001.3gppnetwork.org",
    realm: "epc.mnc001.mcc001.3gppnetwork.org",
    peer_list: [
      %{
        host: "pcrf-primary.epc.mnc001.mcc001.3gppnetwork.org",
        realm: "epc.mnc001.mcc001.3gppnetwork.org",
        ip: "10.0.1.30",
        initiate_connection: true
      },
      %{
        host: "pcrf-backup.epc.mnc001.mcc001.3gppnetwork.org",
        realm: "epc.mnc001.mcc001.3gppnetwork.org",
        ip: "10.0.2.30",
        initiate_connection: true
      }
    ]
  },

  # S5/S8 (interface SGW-C)
  s5s8: %{
```

```
    local_ipv4_address: "10.0.0.20"
  },

  # Sxb/PFCP (interface PGW-U)
  sxb: %{
    local_ip_address: "10.0.0.20"
  },

  # Seleção de UPF
  upf_selection: %{
    rules: [
      %{
        name: "Internet Padrão",
        priority: 1,
        match_field: :apn,
        match_regex: ~r/^internet/,
        upf_pool: [
          %{remote_ip_address: "10.0.0.21", remote_port: 8805,
weight: 100},
          %{remote_ip_address: "10.0.0.22", remote_port: 8805,
weight: 0}  # Standby
        ]
      }
    ],
    fallback_pool: [
      %{remote_ip_address: "10.0.0.21", remote_port: 8805, weight:
100}
    ]
  },

  # Pools de IP do UE
  ue: %{
    subnet_map: %{
      "internet" => [
        "100.64.0.0/18"  # 16.382 IPs
      ],
      "ims" => [
        "100.64.64.0/22"  # 1.022 IPs
      ],
      "enterprise.corp" => [
        "10.100.0.0/16"  # 65.534 IPs
      ],
      default: [
        "100.64.127.0/24"  # 254 IPs
```

```
      ]
    }
  },

  # Opções de Configuração de Protocolo
  pco: %{
    primary_dns_server_address: "8.8.8.8",
    secondary_dns_server_address: "8.8.4.4",
    p_cscf_ipv4_address_list: ["10.0.0.50", "10.0.0.51"],
    ipv4_link_mtu_size: 1400
  }
```

Validação da Configuração

Validação na Inicialização

OmniPGW valida a configuração na inicialização. Verifique os logs:

```
[info] Carregando configuração de runtime.exs
[info] Validando configuração...
[info] Configuração válida
[info] Iniciando OmniPGW...
```

Erros Comuns de Validação

Endereço IP Inválido:

```
[error] Endereço IP inválido em s5s8.local_ipv4_address: "10.0.0"
```

Campo Necessário Ausente:

```
[error] Configuração necessária ausente: sxb.local_ip_address
```

CIDR Inválido:

```
[error] Sub-rede inválida em ue.subnet_map: "100.64.1.0/33"
```

Identidade Diameter Inválida:

```
[error] O host Diameter deve ser FQDN, não IP: "10.0.0.20"
```

Testando a Configuração

Testar configuração sem iniciar:

```
# Validar sintaxe
mix compile

# Verificar carregamento da configuração
iex -S mix
iex> Application.get_env(:pgw_c, :metrics)
```

Configuração Específica do

Ambiente

Desenvolvimento

```
# config/dev.exs
import Config

config :logger, level: :debug

config :pgw_c,
  metrics: %{
    enabled: true,
    ip_address: "127.0.0.1",
    port: 42069,
    registry_poll_period_ms: 10_000
  },
  diameter: %{
    listen_ip: "0.0.0.0",
    host: "omnipgw-dev.local",
    realm: "local",
    peer_list: [
      %{
        host: "pcrf-dev.local",
        realm: "local",
        ip: "127.0.0.1",
        initiate_connection: true
      }
    ]
  },
  s5s8: %{
    local_ipv4_address: "127.0.0.10"
  },
  sxb: %{
    local_ip_address: "127.0.0.20"
  },
  upf_selection: %{
    fallback_pool: [
      %{remote_ip_address: "127.0.0.21", remote_port: 8805,
weight: 100}
    ]
  },
  ue: %{
```

```
    subnet_map: %{
      "internet" => ["100.64.1.0/24"],
      default: ["42.42.42.0/24"]
    }
  },
  pco: %{
    primary_dns_server_address: "8.8.8.8",
    secondary_dns_server_address: "8.8.4.4",
    ipv4_link_mtu_size: 1400
  }
```

Usando Variáveis de Ambiente

```
# config/runtime.exs
import Config

config :pgw_c,
  metrics: %{
    enabled: System.get_env("METRICS_ENABLED", "true") == "true",
    ip_address: System.get_env("METRICS_IP", "0.0.0.0"),
    port: String.to_integer(System.get_env("METRICS_PORT",
"9090")),
    registry_poll_period_ms: 10_000
  },
  diameter: %{
    listen_ip: System.get_env("DIAMETER_LISTEN_IP", "0.0.0.0"),
    host: System.get_env("DIAMETER_HOST") || raise("DIAMETER_HOST
required"),
    realm: System.get_env("DIAMETER_REALM") ||
raise("DIAMETER_REALM required"),
    peer_list: [
      %{
        host: System.get_env("PCRF_HOST") || raise("PCRF_HOST
required"),
        realm: System.get_env("PCRF_REALM") ||
System.get_env("DIAMETER_REALM"),
        ip: System.get_env("PCRF_IP") || raise("PCRF_IP
required"),
        initiate_connection: true
      }
    ]
  }
  # ... resto da configuração
```

Uso:

```
export DIAMETER_HOST="omnipgw.epc.mnc001.mcc001.3gppnetwork.org"
export DIAMETER_REALM="epc.mnc001.mcc001.3gppnetwork.org"
export PCRF_HOST="pcrf.epc.mnc001.mcc001.3gppnetwork.org"
export PCRF_IP="10.0.0.30"

mix run --no-halt
```

Documentação Relacionada

Configuração da Interface

- **Interface PFCP** - Configuração Sxb/PFCP, comunicação UPF, estabelecimento de sessão
- **Interface Diameter Gx** - Controle de política PCRF, regras PCC, gerenciamento de QoS
- **Interface Diameter Gy** - Cobrança online OCS, gerenciamento de cotas, controle de crédito
- **Interface S5/S8** - Configuração GTP-C v2, comunicação SGW-C (4G/LTE)
- **Interface Gn/Gp** - Configuração GTP-C v1, comunicação SGSN (2G/3G GGSN)

Configuração de Rede

- **Alocação de IP do UE** - Gerenciamento de pools de IP, alocação baseada em APN, DHCP
- **Configuração do PCO** - Opções de Configuração de Protocolo, DNS, P-CSCF, MTU
- **Monitoramento do P-CSCF** - Monitoramento de descoberta do P-CSCF, rastreamento de saúde do IMS

Guias Operacionais

- **Gerenciamento de Sessão** - Ciclo de vida da sessão PDN, gerenciamento de bearer
- **Guia de Monitoramento** - Métricas Prometheus, alertas, painéis
- **Formato de CDR de Dados** - Registros de cobrança offline, integração de faturamento

---

Voltar ao Guia de Operações

Guia de Configuração do OmniPGW - *por Omnitouch Network Services*

Formato do Registro de Dados de Cobrança (CDR)

Cobrança Offline para PGW-C

OmniPGW da Omnitouch Network Services

Índice

1. Visão Geral
2. Formato do Arquivo CDR
3. Campos do CDR
4. Eventos do CDR
5. Estrutura do Arquivo
6. Configuração
7. Fluxo de Geração do CDR
8. Detalhes dos Campos
9. Exemplos
10. Integração

Visão Geral

O formato **Data CDR (Registro de Dados de Cobrança)** fornece capacidades de cobrança offline para o Plano de Controle do Gateway de Pacotes (PGW-C). Os CDRs são gerados para registrar eventos de sessão de portadora, uso de dados e informações do assinante para fins de faturamento e análise.

Este formato comum é compatível com os CDRs do SGW-C, garantindo consistência nos registros de cobrança em toda a infraestrutura EPC.

Principais Recursos

- **Formato baseado em CSV** - Valores separados por vírgula simples e legíveis por humanos
- **Registro baseado em eventos** - Captura eventos de início, atualização e término da portadora
- **Medição de volume** - Registra o uso de dados de uplink e downlink
- **Rotação automática** - Rotação de arquivos configurável com base em intervalos de tempo
- **Conformidade com 3GPP** - Segue 3GPP TS 32.251 (cobrança de domínio PS) e TS 32.298 (codificação de CDR)

Casos de Uso

Caso de Uso	Descrição
Cobrança Offline	Gerar CDRs para faturamento pós-pago
Análise	Analisar padrões de uso dos assinantes
Rastro de Auditoria	Rastrear todos os eventos de sessão de portadora
Planejamento de Capacidade	Monitorar a utilização de recursos da rede
Resolução de Problemas	Depurar problemas de sessão e portadora

---

Formato do Arquivo CDR

Convenção de Nomenclatura de Arquivo

```
<epoch_timestamp>
```

Exemplo:

```
1726598022
```

O nome do arquivo é o timestamp da época Unix (em segundos) de quando o arquivo foi criado.

Localização do Arquivo

Diretório padrão:

- PGW-C: `/var/log/pgw_c/cdrs/`

Configurável via parâmetro `cdr_directory` no config/runtime.exs.

Cabeçalho do Arquivo

Cada arquivo CDR começa com um cabeçalho de múltiplas linhas contendo metadados:

```
# Arquivo Data CDR:
# Hora de Início do Arquivo: HH:MM:SS (unix_timestamp)
# Hora de Fim do Arquivo: HH:MM:SS (unix_timestamp)
# Nome do Gateway: <gateway_name>
#
epoch,imsi,event,charging_id,msisdn,ue_imei,timezone_raw,plmn,tac,eci
```

Campos do Cabeçalho:

- **Hora de Início do Arquivo** - Quando o arquivo CDR foi criado (legível por humanos e timestamp Unix)
- **Hora de Fim do Arquivo** - Quando a rotação do arquivo ocorrerá (legível por humanos e timestamp Unix)
- **Nome do Gateway** - Identificador para a instância PGW-C (configurado via parâmetro `pgw_name`)
- **Cabeçalhos de Coluna** - Nomes dos campos CSV para os registros de dados

---

Campos do CDR

Resumo dos Campos

Posição	Nome do Campo	Tipo	Descrição
0	epoch	inteiro	Timestamp do evento (segundos da época Unix)
1	imsi	string	Identidade Internacional do Assinante Móvel
2	event	string	Tipo de evento CDR (ex: "default_bearer_start")
3	charging_id	inteiro	Identificador de cobrança único para a portadora
4	msisdn	string	Número ISDN da Estação Móvel (número de telefone)
5	ue_imei	string	Identidade Internacional de Equipamento Móvel
6	timezone_raw	string	Fuso horário do UE (reservado, atualmente vazio)
7	plmn	inteiro	Identificador da Rede Móvel Pública
8	tac	inteiro	Código da Área de Rastreamento
9	eci	inteiro	Identificador da Célula E-UTRAN
10	sgw_ip	string	Endereço IP do plano de controle S5/S8 do SGW-C

Posição	Nome do Campo	Tipo	Descrição
11	ue_ip	string	Endereço IP do UE (formato IPv4|IPv6)
12	pgw_ip	string	Endereço IP do plano de controle S5/S8 do PGW-C
13	apn	string	Nome do Ponto de Acesso
14	qci	inteiro	Identificador da Classe de QoS
15	octets_in	inteiro	Volume de dados de downlink (bytes)
16	octets_out	inteiro	Volume de dados de uplink (bytes)

Eventos do CDR

Tipos de Eventos

CDRs são gerados para três tipos de eventos:

Tipo de Evento	Formato	Descrição	Quando Gera
Início da Portadora	`<type>_bearer_start`	Estabelecimento da portadora	Resposta de Cria de Sessão envia
Atualização da Portadora	`<type>_bearer_update`	Relatório de uso durante a sessão	Relatórios de us periódicos do pl de usuário
Término da Portadora	`<type>_bearer_end`	Término da portadora	Solicitação/Resp de Exclusão de Sessão

Tipos de Portadora:

- `default` - Portadora padrão (uma por conexão PDN)
- `dedicated` - Portadora dedicada (zero ou mais por conexão PDN)

Exemplos de Eventos

```
default_bearer_start      - Portadora padrão estabelecida
default_bearer_update     - Atualização de uso da portadora padrão
default_bearer_end        - Portadora padrão terminada
dedicated_bearer_start    - Portadora dedicada estabelecida
dedicated_bearer_update   - Atualização de uso da portadora
dedicada
dedicated_bearer_end      - Portadora dedicada terminada
```

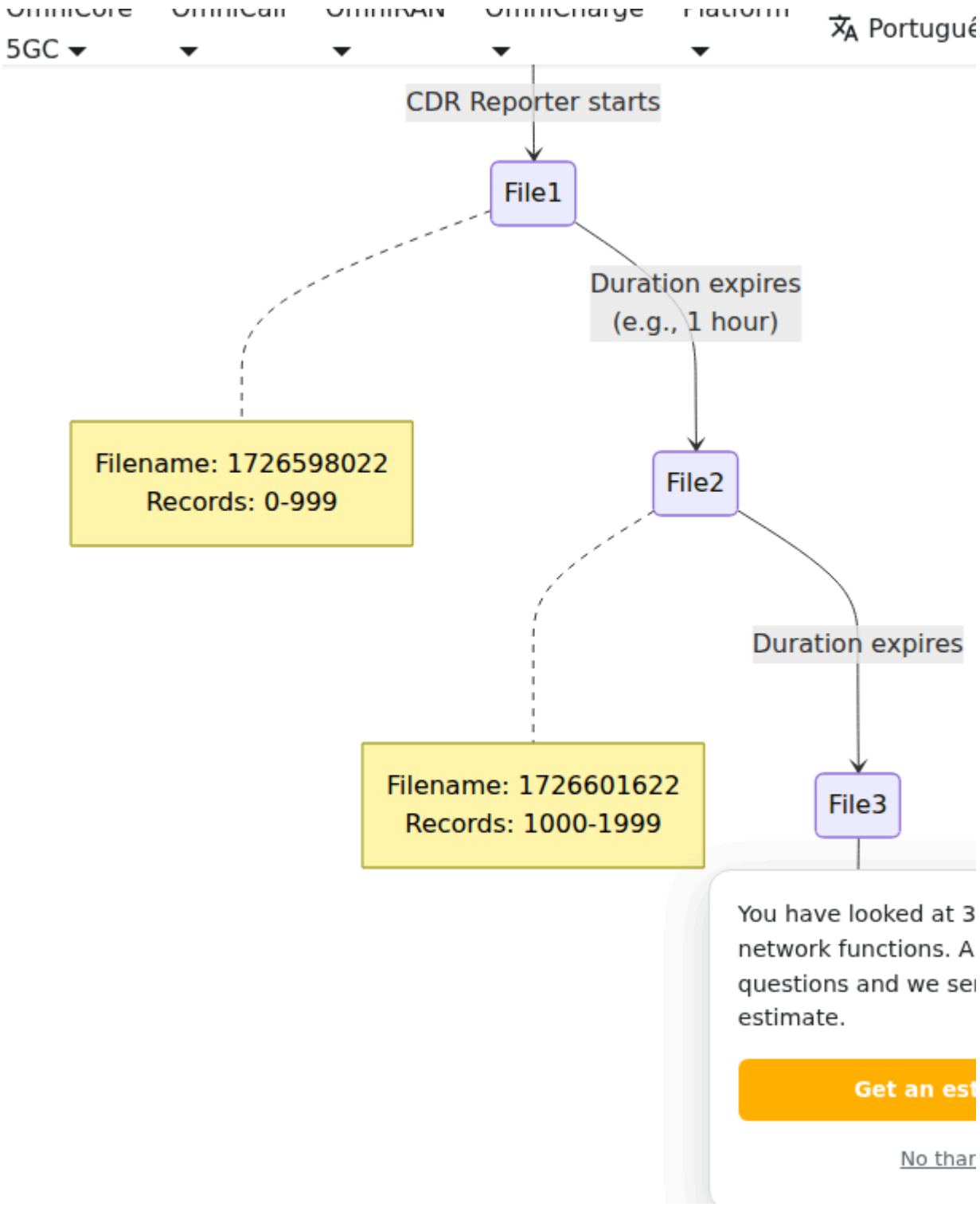
Estrutura do Arquivo

Exemplo de Arquivo CDR

```
# Arquivo Data CDR:
# Hora de Início do Arquivo: 18:53:42 (1726598022)
# Hora de Fim do Arquivo: 19:53:42 (1726601622)
# Nome do Gateway: sgw-c-prod-01
# epoch,imsi,event,charging_id,msisdn,ue_imei,timezone_raw,plmn,tac,e
1726598022,310260123456789,default_bearer_start,12345,15551234567,123
1726598322,310260123456789,default_bearer_update,12345,15551234567,12
1726598622,310260123456789,default_bearer_update,12345,15551234567,12
1726598922,310260123456789,default_bearer_end,12345,15551234567,12345
```

Rotação de Arquivo

Os arquivos CDR são automaticamente rotacionados com base na duração configurada:



Processo de Rotação:

1. Fechar o arquivo CDR atual
2. Criar um novo arquivo com o timestamp atual
3. Escrever o cabeçalho no novo arquivo
4. Continuar gravando CDRs no novo arquivo

Configuração

Parâmetros de Configuração

A geração de CDR do PGW-C é configurada em `config/runtime.exs`:

Parâmetro	Tipo	Descrição	Padrão	Reco
`pgw_name`	string	Identificador da instância PGW (aparece nos cabeçalhos do CDR)	"omni-pgw01"	Usar nom ID da inst
`cdr_file_duration`	inteiro	Intervalo de rotação do arquivo (ms)	3600000	3600000
`cdr_directory`	string	Caminho do diretório de saída do CDR	"/tmp/pgw_c"	`/var/log`
`usage_report_interval`	inteiro	Intervalo de relatório URR (ms) - com que frequência o PGW-U envia relatórios de uso	60000	60000 (1

Exemplos de Configuração

Configuração Mínima (config/runtime.exs):

```
config :pgw_c,
  # Configuração do arquivo CDR
  pgw_name: "omni-pgw01",
  cdr_file_duration: 3_600_000,           # 1 hora
  cdr_directory: "/var/log/pgw_c/cdrs",

  # Configuração do URR (dispara relatórios de uso do PGW-U)
  usage_report_interval: 60_000           # 60 segundos
```

Produção:

```
config :pgw_c,
  pgw_name: "pgw-c-prod-01",
  cdr_file_duration: 3_600_000,           # rotação de 1 hora
  cdr_directory: "/var/log/pgw_c/cdrs",
  usage_report_interval: 60_000           # atualizações de 1
minuto
```

Desenvolvimento:

```
config :pgw_c,
  pgw_name: "pgw-c-dev",
  cdr_file_duration: 300_000,             # rotação de 5 minutos
para testes
  cdr_directory: "/tmp/pgw_c_cdrs",
  usage_report_interval: 30_000           # atualizações de 30
segundos para testes mais rápidos
```

Alto Volume:

```
config :pgw_c,
  pgw_name: "pgw-c-prod-heavy",
  cdr_file_duration: 1_800_000,           # rotação de 30 minutos
  cdr_directory: "/mnt/fast-storage/cdrs",
  usage_report_interval: 300_000          # atualizações de 5
minutos (reduzir sobrecarga)
```

URR (Regras de Relatório de Uso)

O PGW-C usa **PFCP URRs (Regras de Relatório de Uso)** para disparar relatórios de uso do PGW-U. Quando um limite de URR é atingido ou o tempo expira, o PGW-U envia uma Solicitação de Relatório de Sessão contendo dados de uso, o que dispara a geração de CDR.

Como Funciona a Configuração do URR:

1. `usage_report_interval` (em ms) é convertido em segundos para o limite de tempo PFCP
2. O PGW-C cria URR com limite de tempo durante o estabelecimento da sessão
3. O PGW-U envia relatórios de uso periódicos no intervalo configurado
4. Cada relatório de uso dispara um evento CDR `bearer_update`
5. O relatório de uso final (na exclusão da sessão) dispara um evento CDR `bearer_end`

Exemplo: `usage_report_interval: 60_000` significa:

- O PGW-U relata uso a cada 60 segundos
- Eventos de atualização do CDR gerados a cada 60 segundos
- Rastreamento granular de uso para faturamento

Definição do Tipo de URR:

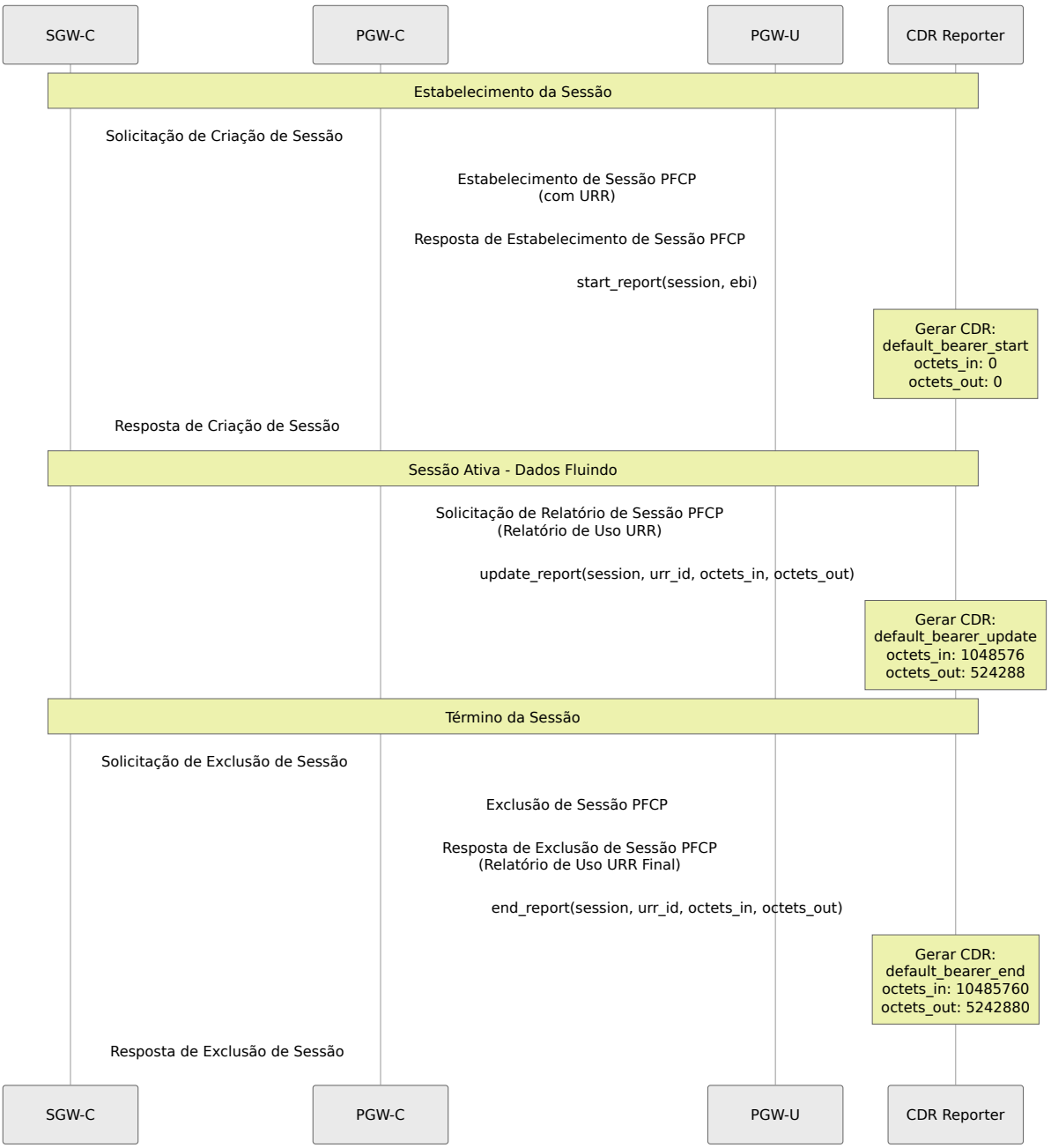
```
defmodule PGW_C.Session.Types.URR do
  typedstruct do
    field :urr_id, non_neg_integer()
    field :measurement_method, :duration | nil
    field :reporting_triggers, :time_threshold | nil
    field :time_threshold, non_neg_integer() | nil  # segundos
  end
end
```

Veja a Documentação da Interface PFCP para detalhes do PFCP URR, incluindo como os URRs são criados durante o estabelecimento da sessão.

Fluxo de Geração do CDR

Eventos do Ciclo de Vida da Portadora CDR

Geração de CDR do PGW-C:



Eventos de Geração do CDR

1. Início da Portadora:

- **Quando:** Resposta de Criação de Sessão é enviada
- **Propósito:** Registra o estabelecimento da portadora com uso zero
- **octets_in:** 0
- **octets_out:** 0

2. Atualização da Portadora:

- **Quando:** Solicitação de Relatório de Sessão PFCP recebida do PGW-U (relatório de uso URR)
- **Propósito:** Registra o uso incremental de dados
- **octets_in:** Bytes de downlink cumulativos desde o início da portadora
- **octets_out:** Bytes de uplink cumulativos desde o início da portadora
- **Disparador:** Limite de tempo URR expira (configurado via `usage_report_interval`)

3. Término da Portadora:

- **Quando:** Resposta de Exclusão de Sessão PFCP recebida do PGW-U (com relatório de uso final)
- **Propósito:** Registra o uso final de dados antes do término da sessão
- **octets_in:** Total final de bytes de downlink
- **octets_out:** Total final de bytes de uplink

Detalhes dos Campos

1. epoch (Timestamp)

Tipo: Timestamp da época Unix (segundos)

Descrição: O tempo em que o evento CDR ocorreu

Exemplo:

```
1726598022  → 2025-09-17 18:53:42 UTC
```

---

2. imsi (Identidade do Assinante)

Tipo: String (até 15 dígitos)

Formato: MCCMNC + MSIN

Descrição: Identidade Internacional do Assinante Móvel que identifica exclusivamente o assinante

Exemplo:

```
310260123456789
  └──┘ └──┘ └─────────┘
   MCC  MNC     MSIN
   (310)(260) (123456789)
```

Fonte: Contexto do UE, recebido na Solicitação de Criação de Sessão

---

3. event (Tipo de Evento CDR)

Tipo: String

Formato: `<bearer_type>_bearer_<event>`

Valores:

- `default_bearer_start`
- `default_bearer_update`
- `default_bearer_end`
- `dedicated_bearer_start`
- `dedicated_bearer_update`
- `dedicated_bearer_end`

Determinação:

- Se EBI (EPS Bearer ID) é igual a LBI (Linked Bearer ID): `default`
- Se EBI não é igual a LBI: `dedicated`

Fonte: Contexto da portadora (comparação EBI vs LBI)

---

4. charging_id (Identificador de Cobrança)

Tipo: Inteiro não assinado de 32 bits

Descrição: Identificador único para correlação de cobrança entre elementos de rede

Exemplo:

```
12345
```

Fonte: Atribuído pelo PGW-C, recebido na Resposta de Criação de Sessão

Uso:

- Correlaciona eventos de cobrança entre SGW e PGW
- Usado nas interfaces de cobrança Diameter Gy/Gz
- Único por portadora

---

5. msisdn (Número de Telefone)

Tipo: String (formato E.164)

Descrição: Número ISDN da Estação Móvel (número de telefone do assinante)

Formato: Código do país + número nacional

Exemplo:

```
15551234567
 └┘└─────┘
  CC  Nacional
  (1) (5551234567)
```

Fonte: Contexto do UE, tipicamente do HSS via MME

---

6. ue_imei (Identidade do Equipamento)

Tipo: String (15 dígitos)

Formato: TAC (8) + SNR (6) + Spare (1)

Descrição: Identidade Internacional de Equipamento Móvel (identificador do dispositivo)

Exemplo:

```
123456789012345
└──────┘└────┘└
   TAC   SNR S
```

Fonte: Contexto do UE, recebido do MME

---

7. timezone_raw (Fuso Horário do UE)

Tipo: String (atualmente reservado/vazio)

Descrição: Campo reservado para informações de fuso horário do UE

Status Atual: Não populado (campo vazio no CSV)

Uso Futuro: Pode incluir deslocamento de fuso horário e sinalizador de horário de verão

Exemplo:

```
,  (campo vazio)
```

---

8. plmn (Identificador da Rede)

Tipo: Inteiro (formato legado)

Descrição: Identificador da Rede Móvel Pública codificado como hex little-endian

Processo de Codificação:

```
MCC: 505, MNC: 57
  ↓
"50557"
  ↓
Trocar pares: "055570"
  ↓
Hex para decimal: 0x055570 = 349552
```

Exemplo:

```
349552  → MCC: 505, MNC: 57
```

Fonte: Informações de localização do UE do MME

Nota: Este é um formato de codificação legado para compatibilidade retroativa

---

9. tac (Código da Área de Rastreamento)

Tipo: Inteiro não assinado de 16 bits

Descrição: O Código da Área de Rastreamento identifica a área de rastreamento onde o UE está localizado

Intervalo: 0 - 65535

Exemplo:

```
1234
```

Fonte: Informações de localização do UE, recebidas do MME na Solicitação de Criação de Sessão

Uso:

- Identifica a área de gerenciamento de mobilidade
- Usado para paginação e atualizações de localização
- Parte do TAI (Identidade da Área de Rastreamento)

---

10. eci (Identificador da Célula E-UTRAN)

Tipo: Inteiro não assinado de 28 bits

Descrição: O Identificador da Célula E-UTRAN identifica exclusivamente a célula que atende o UE

Formato: ID do eNodeB (20 bits) + ID da Célula (8 bits)

Intervalo: 0 - 268.435.455

Exemplo:

```
5678
```

Fonte: Informações de localização do UE do MME

Uso:

- Identifica a torre de celular e setor específicos
- Usado para transferência e gerenciamento de mobilidade
- Informações de localização granular

---

11. sgw_ip (IP do Plano de Controle SGW)

Tipo: String (endereço IPv4 ou IPv6)

Descrição: Endereço IP do plano de controle S5/S8 do SGW-C (F-TEID)

Formato: Decimal pontuado (IPv4) ou hex com dois pontos (IPv6)

Exemplo:

```
10.0.0.15       (IPv4)
2001:db8::15    (IPv6)
```

Fonte: Configuração local, atribuída à interface S5/S8

---

12. ue_ip (Endereço IP do UE)

Tipo: String (formato IPv4|IPv6)

Descrição: Endereço IP atribuído ao UE para a conexão PDN

Formato: `<ipv4>|<ipv6>`

Exemplos:

```
172.16.1.100|                       (somente IPv4)
|2001:db8::1                        (somente IPv6)
172.16.1.100|2001:db8::1          (Dual-stack)
```

Fonte: Alocação de Endereço PDN (PAA) do PGW-C

Notas:

- IPv4 vazio: Nenhum endereço IPv4 alocado
- IPv6 vazio: Nenhum endereço IPv6 alocado
- Ambos presentes: Conexão PDN dual-stack

---

13. pgw_ip (IP do Plano de Controle PGW)

Tipo: String (endereço IPv4 ou IPv6)

Descrição: Endereço IP do plano de controle S5/S8 do PGW-C (F-TEID remoto)

Formato: Decimal pontuado (IPv4) ou hex com dois pontos (IPv6)

Exemplo:

```
10.0.0.20       (IPv4)
2001:db8::20    (IPv6)
```

Fonte: Recebido na Resposta de Criação de Sessão do PGW-C

---

14. apn (Nome do Ponto de Acesso)

Tipo: String (até 100 caracteres)

Descrição: Nome do Ponto de Acesso que identifica a rede externa (PDN)

Formato: Formato de rótulo semelhante ao DNS

Exemplos:

```
internet
ims
mms
enterprise.corporate
```

Fonte: Recebido na Solicitação de Criação de Sessão do MME

Uso:

- Determina a qual rede externa se conectar
- Direciona políticas e regras de cobrança
- Pode determinar o pool de endereços IP

---

15. qci (Identificador da Classe de QoS)

Tipo: Inteiro não assinado de 8 bits

Descrição: O Identificador da Classe de QoS define a qualidade de serviço da portadora

Intervalo: 1 - 9 (padronizado), 128-254 (específico do operador)

Valores de QCI Padronizados:

QCI	Tipo de Recurso	Prioridade	Atraso de Pacote	Perda de Pacote	Serviço Exemplo
1	GBR	2	100 ms	10^-2	Voz Conversacional
2	GBR	4	150 ms	10^-3	Vídeo Conversacional
3	GBR	3	50 ms	10^-3	Jogos em Tempo Real
4	GBR	5	300 ms	10^-6	Vídeo Não Conversacional
5	Non-GBR	1	100 ms	10^-6	Sinalização IMS
6	Non-GBR	6	300 ms	10^-6	Vídeo (buffered)
7	Non-GBR	7	100 ms	10^-3	Voz, Vídeo, Jogos
8	Non-GBR	8	300 ms	10^-6	Vídeo (buffered)
9	Non-GBR	9	300 ms	10^-6	Portadora Padrão

Exemplo:

```
9  → Portadora padrão (esforço máximo)
```

Fonte: Parâmetros de QoS da portadora do PGW-C

---

16. octets_in (Volume de Downlink)

Tipo: Inteiro não assinado de 64 bits

Descrição: Número de bytes transmitidos na direção de downlink (rede → UE)

Unidades: Bytes

Exemplo:

```
1048576  → 1 MB de downlink
```

Fonte: Medição de Volume PFCP do PGW-U (via relatórios de uso URR)

Notas:

- Cumulativo para eventos `update`
- Total final para eventos `end`
- Sempre 0 para eventos `start`
- Relatórios disparados pelo limite de tempo URR (configurado via `usage_report_interval`)

---

17. octets_out (Volume de Uplink)

Tipo: Inteiro não assinado de 64 bits

Descrição: Número de bytes transmitidos na direção de uplink (UE → rede)

Unidades: Bytes

Exemplo:

```
524288  → 512 KB de uplink
```

Fonte: Medição de Volume PFCP do PGW-U (via relatórios de uso URR)

Notas:

- Cumulativo para eventos `update`
- Total final para eventos `end`
- Sempre 0 para eventos `start`
- Relatórios disparados pelo limite de tempo URR (configurado via `usage_report_interval`)

Exemplos

Exemplo 1: Sessão Básica com Atualização Única

Linha do Tempo:

1. Portadora estabelecida
2. 5 minutos depois: Atualização de uso (10 MB down, 5 MB up)
3. Sessão terminada

Saída do CDR:

```
# Arquivo Data CDR:
# Hora de Início do Arquivo: 10:00:00 (1726570800)
# Hora de Fim do Arquivo: 11:00:00 (1726574400)
# Nome do Gateway: pgw-c-01
# epoch,imsi,event,charging_id,msisdn,ue_imei,timezone_raw,plmn,tac,e
1726570800,310260111111111,default_bearer_start,10001,15551111111,111
1726571100,310260111111111,default_bearer_update,10001,15551111111,11
1726571400,310260111111111,default_bearer_end,10001,15551111111,11111
```

Exemplo 2: Sessão Dual-Stack com Múltiplas Atualizações

Linha do Tempo:

1. Portadora dual-stack estabelecida (IPv4 + IPv6)
2. Múltiplas atualizações de uso
3. Sessão terminada

Saída do CDR:

```
1726570800,310260222222222,default_bearer_start,10002,15552222222,222
1726571100,310260222222222,default_bearer_update,10002,15552222222,22
1726571400,310260222222222,default_bearer_update,10002,15552222222,22
1726571700,310260222222222,default_bearer_update,10002,15552222222,22
1726572000,310260222222222,default_bearer_end,10002,15552222222,22222
```

Exemplo 3: Sessão com Portadora Dedicada

Linha do Tempo:

1. Portadora padrão estabelecida (QCI 9)
2. Portadora dedicada criada para vídeo (QCI 6)
3. Atualizações de uso para ambas as portadoras
4. Portadora dedicada excluída
5. Portadora padrão terminada

Saída do CDR:

```
1726570800,310260333333333,default_bearer_start,10003,15553333333,333
1726571100,310260333333333,dedicated_bearer_start,10004,15553333333,3
1726571400,310260333333333,default_bearer_update,10003,15553333333,33
1726571400,310260333333333,dedicated_bearer_update,10004,15553333333,
1726571700,310260333333333,dedicated_bearer_end,10004,15553333333,333
1726572000,310260333333333,default_bearer_end,10003,15553333333,33333
```

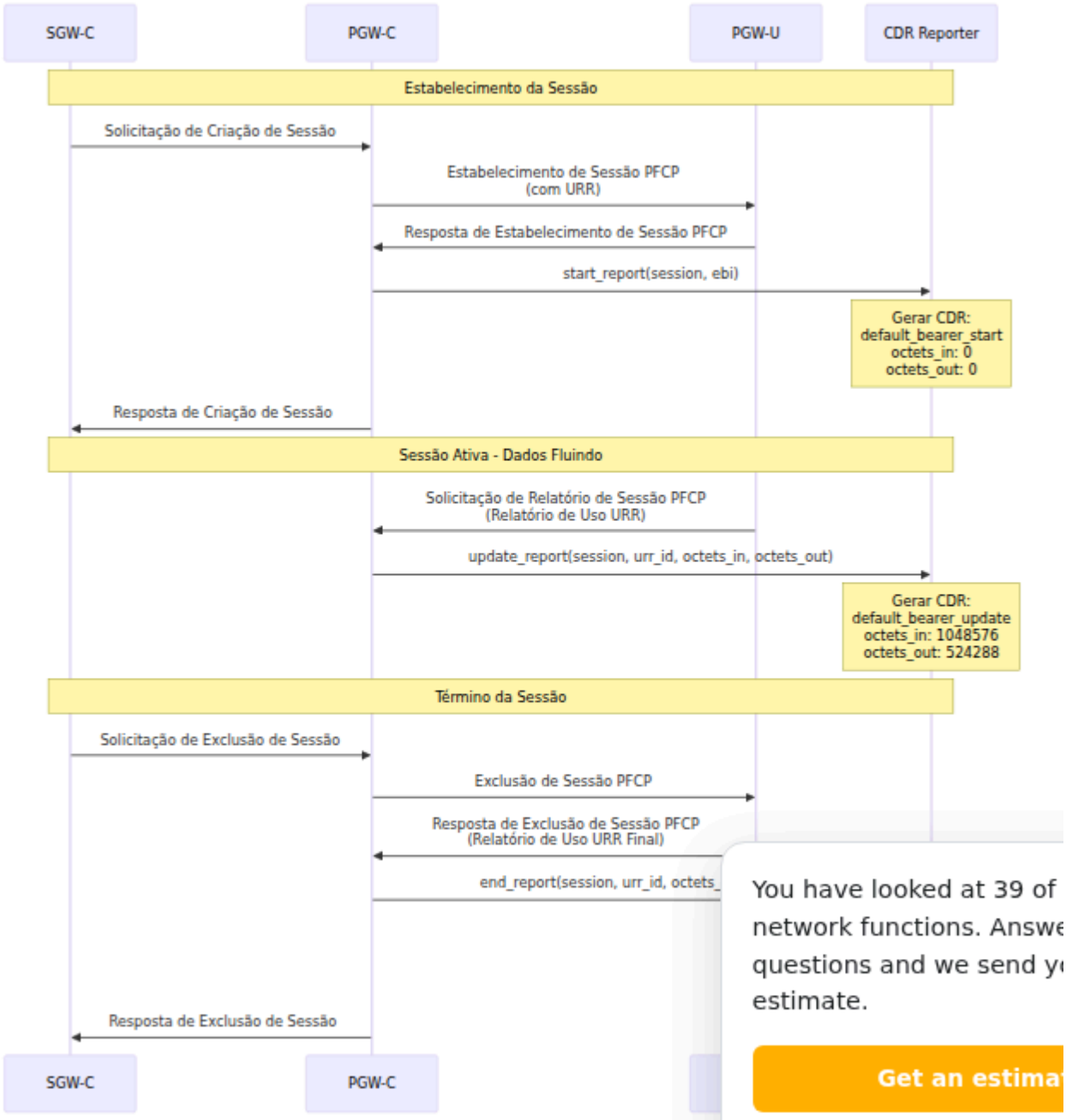
Análise:

- A portadora padrão (10003) transporta tráfego de fundo (10 MB down, 4 MB up)

- A portadora dedicada (10004) transporta tráfego de vídeo (200 MB down, 2 MB up)
- Diferentes valores de QCI (9 vs 6) refletem diferentes tratamentos de QoS

Integração

Pipeline de Processamento do CDR



You have looked at 39 of
network functions. Answe
questions and we send y
estimate.
Get an estima

Métodos de Coleta de CDR

1. Coleta Baseada em Arquivo:

```
# Monitorar diretório CDR (PGW-C)
inotifywait -m /var/log/pgw_c/cdrs/ -e close_write | while read
path action file; do
    # Rotação de arquivo concluída, processar CDR
    process_cdr "$path$file"
done
```

2. Streaming em Tempo Real:

```
# Tail e stream para pipeline de processamento
tail -F /var/log/pgw_c/cdrs/* | process_cdr_stream
```

Documentação Relacionada

- Gerenciamento de Sessão - Ciclo de vida da sessão e gatilhos do CDR
- Interface PFCP - Relatório de uso do PGW-U via URRs
- Guia de Monitoramento - Métricas de geração de CDR e alertas
- Guia de Configuração - Parâmetros de configuração do CDR e URR
- Interface Diameter Gx - Controle de políticas para valores de QCI nos CDRs
- Interface Diameter Gy - Integração de cobrança online

Referências 3GPP

- TS 32.251 - Cobrança de domínio Packet Switched (PS)
- TS 29.274 - Sistema de Pacotes Evoluído (EPS) da 3GPP; protocolo GTP-C
- TS 29.244 - Interface entre nós CP e UP (PFCP) - **suporte a URR**
- TS 32.298 - Codificação de CDR

Formato do CDR - *Registros de Cobrança Offline para PGW-C*

Desenvolvido pela Omnitouch Network Services

Versão da Documentação: 1.0 **Última Atualização:** 2025-12-28

Documentação da Interface Gx do Diameter

Função de Regras de Política e Cobrança (PCRF)

Índice

1. Visão Geral
2. Fundamentos da Interface Gx
3. Protocolo Diameter
4. Mensagens de Controle de Crédito
5. Regras de Política e Cobrança
6. Relatório de Mudança de Localização
7. Configuração
8. Fluxos de Mensagens
9. Tratamento de Erros
10. Solução de Problemas

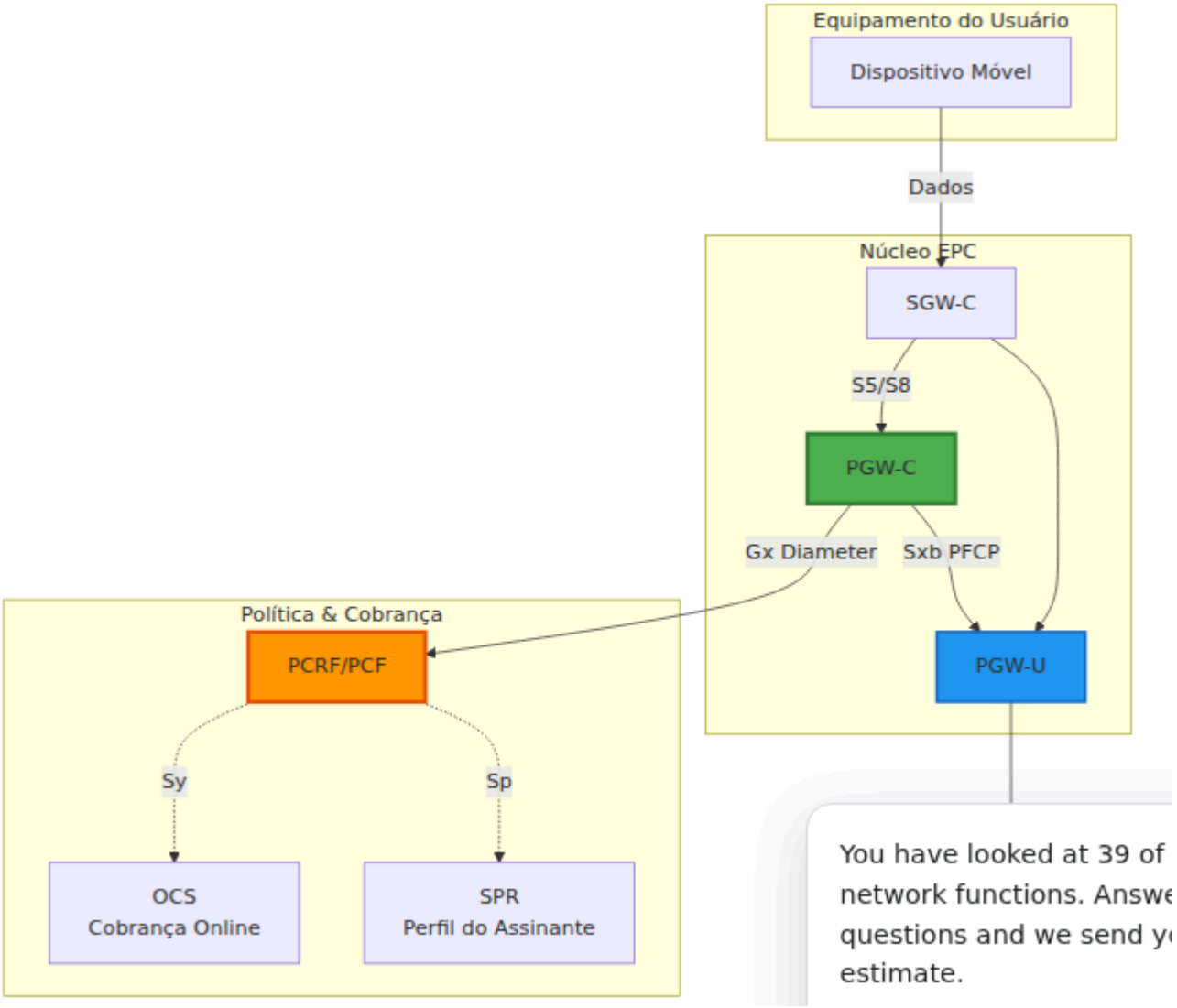
Visão Geral

A **interface Gx** conecta o PGW-C ao **PCRF (Função de Regras de Política e Cobrança)** ou **PCF (Função de Controle de Política)** em redes 5G. Esta interface permite:

- **Controle Dinâmico de Política** - Aplicação de QoS e políticas em tempo real
- **Controle de Cobrança** - Autorização de crédito e rastreamento de uso

- **Consciência de Serviço** - Diferenciação de tráfego em nível de aplicação
- **Gerenciamento de Perfil do Assinante** - Aplicação de políticas por usuário

Gx na Arquitetura da Rede



Funções Principais

Função	Descrição
Provisionamento de Política	O PCRF fornece regras de PCC definindo como lidar com o tráfego
Controle de QoS	Ajuste dinâmico de taxas de bits e parâmetros de QoS
Controle de Cobrança	Autorização de crédito para cenários pré-pagos/pós-pagos
Controle de Gating	Habilitar/desabilitar fluxos de tráfego com base na pol��tica
Monitoramento de Uso	Rastrear consumo de dados por serviço

Fundamentos da Interface Gx

Referência 3GPP

- **Especificação:** 3GPP TS 29.212
- **ID da Aplicação Diameter:** 16777238 (Gx)
- **Protocolo:** Protocolo Base Diameter (RFC 6733)

Conceito de Sessão

Cada conexão PDN do UE tem uma **sessão Gx** correspondente identificada por um **Session-ID**. Esta sessão:

- Criada quando o UE se conecta (CCR-Initial)
- Atualizada durante a vida útil da conexão (CCR-Update) - opcional, por exemplo, mudanças de política ou relatório de mudança de localização

- Terminada quando o UE se desconecta (CCR-Termination)

Formato do ID da Sessão

```
Session-ID: <Origin-Host>;<high32>;<low32>[;<optional>]
Exemplo: omni-
pgw_c.epc.mnc999.mcc999.3gppnetwork.org;1234567890;98765
```

Componentes:

- **Origin-Host:** Identidade Diameter do PGW-C
- **high32:** 32 bits altos do identificador único
- **low32:** 32 bits baixos do identificador único

Protocolo Diameter

Estrutura da Mensagem

As mensagens Diameter são codificadas em binário com a seguinte estrutura:

```
Cabeçalho Diameter (20 bytes)
├── Versão (1 byte) = 1
├── Comprimento da Mensagem (3 bytes)
├── Flags (1 byte)
│   ├── R: Requisição (1) / Resposta (0)
│   ├── P: Proxiável
│   ├── E: Erro
│   └── T: Potencialmente retransmitido
├── Código do Comando (3 bytes)
├── ID da Aplicação (4 bytes) = 16777238 (Gx)
├── ID Hop-by-Hop (4 bytes)
└── ID End-to-End (4 bytes)

AVPs (Atributo-Valor Pares)
├── Cabeçalho AVP
│   ├── Código AVP
│   ├── Flags (V, M, P)
│   ├── Comprimento AVP
│   └── ID do Fornecedor (opcional)
└── Dados AVP
```

Conceitos Chave do Diameter

AVP (Atributo-Valor Par):

- Unidade de dados básica no Diameter
- Contém um código, flags e valor
- Pode ser aninhado (AVP Agrupado)

Comando:

- Par Requisição/Resposta
- CCR (Credit-Control-Request) / CCA (Credit-Control-Answer)

Códigos de Resultado:

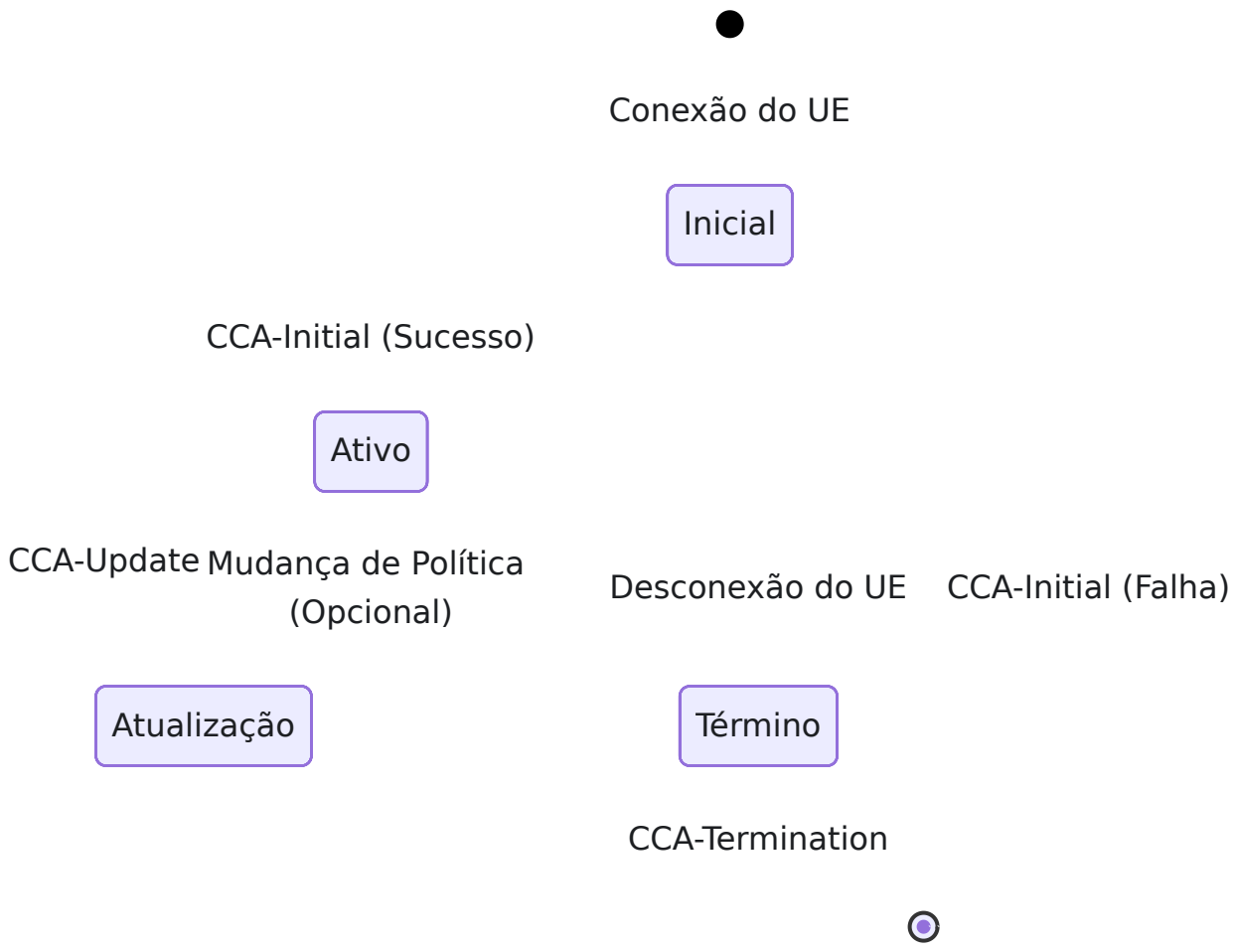
- `2001` - DIAMETER_SUCCESS
- `3xxx` - Erros de protocolo
- `4xxx` - Falhas transitórias

- `5xxx` - Falhas permanentes

Mensagens de Controle de Crédito

O PGW-C usa a **Aplicação de Controle de Crédito Diameter** (RFC 4006) para Gx.

Tipos de Mensagens



CCR-Initial (Requisição de Controle de Crédito - Inicial)

Quando: O UE cria uma nova conexão PDN

Objetivo:

- Solicitar regras de política e cobrança iniciais
- Fornecer contexto do UE e da rede ao PCRF
- Obter parâmetros de QoS e autorização de cobrança

Principais AVPs Enviados pelo PGW-C:

Nome do AVP	Código do AVP	Tipo	Descrição
Session-Id	263	UTF8String	Identificador único da sessão Gx
Auth-Application-Id	258	Unsigned32	16777238 (Gx)
Origin-Host	264	DiamIdent	Identidade Diameter do PGW-C
Origin-Realm	296	DiamIdent	Reino Diameter do PGW-C
Destination-Realm	283	DiamIdent	Reino do PCRF
CC-Request-Type	416	Enumerated	1 = INITIAL_REQUEST
CC-Request-Number	415	Unsigned32	Número da sequência (começa em 0)
Subscription-Id	443	Grouped	Identificador do UE (IMSI/MSISDN)
Called-Station-Id	30	UTF8String	Nome da APN
Framed-IP-Address	8	OctetString	Endereço IPv4 alocado ao UE (PDNs IPv4 e IPv4v6)
Framed-IPv6-Prefix	97	OctetString	Prefixo IPv6 alocado ao UE (PDNs somente IPv6)
3GPP-User-Location-Info	22	OctetString	ULI inicial do assinante (TS 29.061)

Nome do AVP	Código do AVP	Tipo	Descrição
User-Equipment-Info	458	Grouped	Identidade do UE (por exemplo, IMEISV), quando disponível
IP-CAN-Type	1027	Enumerated	5 = 3GPP-EPS
RAT-Type	1032	Enumerated	Mapeado do tipo RAT GTP (por exemplo, 1004 = EUTRAN)
QoS-Information	1016	Grouped	QoS atual (APN-AMBR UL/DL)
Default-EPS-Bearer-QoS	1049	Grouped	QoS do portador padrão QCI/ARP (QCI 5)
Network-Request-Support	1024	Enumerated	Procedimentos iniciados pela rede
Supported-Features	628	Grouped	Lista de recursos Gx (Rel-8/9/10 Gx)

> O AVP framed-address é selecionado pela família de endereços do UE: `Framed-IP-Address` para IPv4 (e o IPv4 de um PDN IPv4v6), `Framed-IPv6-Prefix` para um IPv6-only PDN. Para IPv4v6, o prefixo IPv6 não é transportado no Gx. Veja IPv6 / Dual-Stack.

Exemplo de Estrutura CCR-I:

```
CCR (Código do Comando: 272, Requisição)
├── Session-Id: "pgw_c.example.com;123;456"
├── Auth-Application-Id: 16777238
├── Origin-Host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org"
├── Origin-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── Destination-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── CC-Request-Type: INITIAL_REQUEST (1)
├── CC-Request-Number: 0
├── Subscription-Id (Agrupado)
│   ├── Subscription-Id-Type: END_USER_IMSI (1)
│   └── Subscription-Id-Data: "310260123456789"
├── Called-Station-Id: "internet"
├── Framed-IP-Address: 100.64.1.42
├── IP-CAN-Type: 3GPP-EPS (5)
├── RAT-Type: EUTRAN (1004)
├── QoS-Information (Agrupado)
│   ├── APN-Aggregate-Max-Bitrate-UL: 100000000 (100 Mbps)
│   └── APN-Aggregate-Max-Bitrate-DL: 50000000 (50 Mbps)
├── Network-Request-Support: 1
└── Supported-Features: [...]
```

CCA-Initial (Resposta de Controle de Crédito - Inicial)

Enviado por: PCRF em resposta ao CCR-I

Objetivo:

- Autorizar ou rejeitar a sessão
- Fornecer regras de PCC para tratamento de tráfego
- Especificar parâmetros de QoS

Principais AVPs Recebidos pelo PGW-C:

Nome do AVP	Código do AVP	Descrição
Result-Code	268	Sucesso (2001) ou código de erro
Experimental-Result	297	Códigos de resultado específicos do fornecedor
QoS-Information	1016	QoS autorizada (pode diferir da solicitação)
Charging-Rule-Install	1001	Regras de PCC a serem ativadas
Charging-Rule-Definition	1003	Definições de regras inline
Default-EPS-Bearer-QoS	1049	QoS para portador padrão

Exemplo de Resposta de Sucesso:

```
CCA (Código do Comando: 272, Resposta)
├── Session-Id: "pgw_c.example.com;123;456"
├── Result-Code: DIAMETER_SUCCESS (2001)
├── Origin-Host: "pcrf.example.com"
├── Origin-Realm: "example.com"
├── Auth-Application-Id: 16777238
├── CC-Request-Type: INITIAL_REQUEST (1)
├── CC-Request-Number: 0
├── QoS-Information (Agrupado)
│   ├── APN-Aggregate-Max-Bitrate-UL: 50000000 (50 Mbps -
reduzido)
│   └── APN-Aggregate-Max-Bitrate-DL: 100000000 (100 Mbps -
aumentado)
├── Charging-Rule-Install (Agrupado)
│   ├── Charging-Rule-Name: "default_internet_rule"
│   └── Charging-Rule-Name: "video_streaming_rule"
└── Charging-Rule-Definition (Agrupado)
    ├── Charging-Rule-Name: "default_internet_rule"
    ├── QoS-Information: {...}
    └── Precedence: 1000
```

CCR-Termination (Requisição de Controle de Crédito - Término)

Quando: O UE se desconecta ou a conexão PDN é excluída

Objetivo:

- Notificar o PCRF sobre a terminação da sessão
- Registro final de contabilidade/cobrança

Principais Diferenças em Relação ao CCR-I:

- `CC-Request-Type: TERMINATION_REQUEST (3)`
- Pode incluir estatísticas de uso
- Conjunto de AVPs simplificado

Exemplo CCR-T:

```
CCR (Código do Comando: 272, Requisição)
├── Session-Id: "pgw_c.example.com;123;456"
├── Auth-Application-Id: 16777238
├── Origin-Host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org"
├── Origin-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── Destination-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── CC-Request-Type: TERMINATION_REQUEST (3)
├── CC-Request-Number: 1
└── Termination-Cause: DIAMETER_LOGOUT (1)
```

CCA-Termination

Enviado por: PCRF em resposta ao CCR-T

Objetivo:

- Reconhecer a terminação da sessão
- Nenhuma regra de política retornada

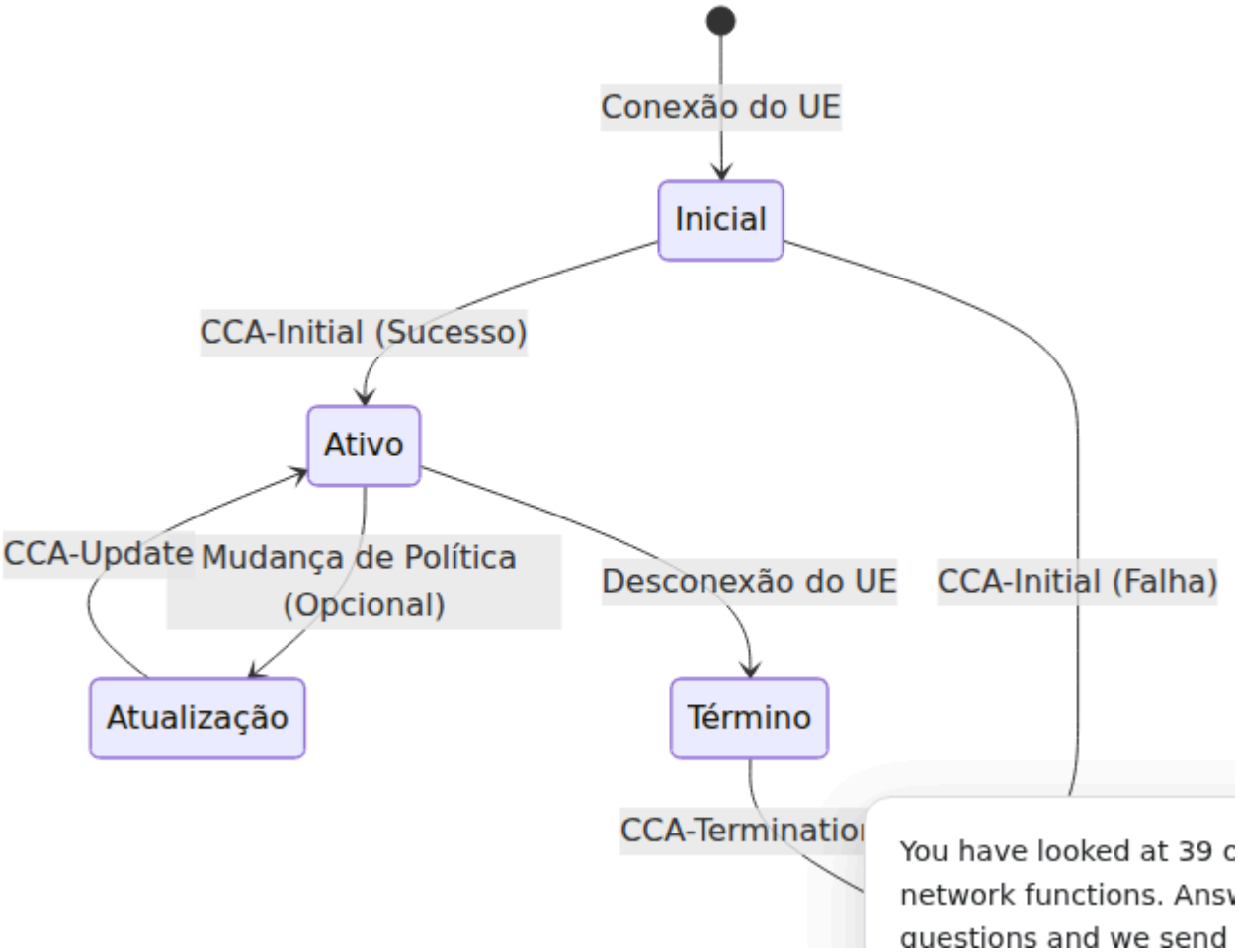
Exemplo CCA-T:

```
CCA (Código do Comando: 272, Resposta)
├── Session-Id: "pgw_c.example.com;123;456"
├── Result-Code: DIAMETER_SUCCESS (2001)
├── Origin-Host: "pcrf.example.com"
├── Origin-Realm: "example.com"
├── Auth-Application-Id: 16777238
├── CC-Request-Type: TERMINATION_REQUEST (3)
└── CC-Request-Number: 1
```

Regras de Política e Cobrança

Estrutura da Regra PCC

Uma **Regra PCC (Controle de Política e Cobrança)** define como lidar com fluxos de tráfego específicos:



Componentes da Regra

1. Nome da Regra:

- Identificador único para a regra
- Exemplo: `"video_streaming_rule"`

2. Precedência:

- Número menor = maior prioridade
- Intervalo: 0-65535
- Usado quando várias regras correspondem

3. Filtros de Fluxo (TFT - Modelo de Fluxo de Tráfego):

- Define quais pacotes correspondem a esta regra
- Exemplos:
  - Tupla IP 5: Protocolo, IP Src/Dst, Porta Src/Dst

- `"permit out ip from any to 8.8.8.8 80"`

4. Informações de QoS:

- **QCI (Identificador de Classe de QoS):** 1-9 (padronizado), 128-254 (específico do operador)
  - QCI 1: Voz Conversacional
  - QCI 5: Sinalização IMS
  - QCI 9: Internet Padrão
- **ARP (Prioridade de Alocação e Retenção):** Capacidade de preempção
- **MBR/GBR:** Taxas de Bits Máximas/Garantidas

5. Informações de Cobrança:

- **Grupo de Avaliação:** Identifica a categoria de cobrança (usado pelo OCS - veja Interface Diameter Gy)
- **Método de Medição:** Baseado em volume, tempo ou evento
- **Cobrança Online/Offline:** OCS (pré-pago via Diameter Gy) vs. CDRs offline (pós-pago - veja Formato de CDR de Dados)

6. Status de Gating:

- **ABERTO:** Permitir tráfego
- **FECHADO:** Bloquear tráfego

Provisionamento Dinâmico de Regras

O PCRF pode fornecer regras de duas maneiras:

1. Regras Predefinidas (por nome):

```
Charging-Rule-Install (Agrupado)
├── Charging-Rule-Name: "gold_subscriber_internet"
└── Charging-Rule-Name: "video_qos_boost"
```

2. Regras Dinâmicas (definição inline):

```
Charging-Rule-Definition (Agrupado)
├── Charging-Rule-Name: "dynamic_rule_123"
├── Precedence: 100
├── Flow-Information (Agrupado)
│   ├── Flow-Description: "permit out ip from any to 192.0.2.0/24"
│   └── Flow-Direction: DOWNLINK
├── QoS-Information (Agrupado)
│   ├── QoS-Class-Identifier: 5
│   ├── Max-Requested-Bandwidth-UL: 10000000
│   └── Max-Requested-Bandwidth-DL: 50000000
└── Rating-Group: 1000
```

AVP de Informações de QoS

APN-AMBR (Taxa de Bits Máxima Agregada):

Aplica-se a todos os portadores não-GBR para esta APN:

```
QoS-Information (Agrupado)
├── APN-Aggregate-Max-Bitrate-UL: 100000000  # 100 Mbps
└── APN-Aggregate-Max-Bitrate-DL: 200000000  # 200 Mbps
```

Resposta do PGW-C:

- Atualiza o estado interno do AMBR
- Envia uma Solicitação de Modificação de Sessão ao PGW-U com o QER atualizado

---

Relatório de Mudança de Localização

O PGW-C pode notificar o PCRF quando a **localização** do assinante, **tecnologia de acesso rádio (RAT)**, **PLMN de serviço** ou **fuso horário do UE** mudam durante a vida de uma sessão. A mudança é transportada em um **CCR-Update** como o **3GPP-User-Location-Info** e AVPs relacionados.

O relatório é opt-in (controlado pelo PCRF)

Esse relatório é **condicional à solicitação do PCRF** — exatamente como exigido pelo 3GPP TS 29.212 §4.5.7 / §5.3.7. O PCRF *arma* os eventos que deseja ouvir incluindo um ou mais AVPs **Event-Trigger** em um **CCA** (ou em um **RAR**). O PGW-C armazena o conjunto armado por sessão Gx e só envia um CCR-Update quando:

1. a condição de um Event-Trigger armado realmente ocorre, **e**
2. uma sessão Gx existe para o assinante.

Se o PCRF não armar **nenhum** Event-Trigger relacionado à localização, o PGW-C permanece silencioso sobre mudanças de localização — ele simplesmente atualiza seu estado interno e responde à Solicitação de Modificação de Portador GTP-C. Isso mantém a interface silenciosa para PCRFs que não se importam com mobilidade e em conformidade para aqueles que se importam.

O conjunto armado pode mudar a qualquer momento durante a sessão. O PGW-C o atualiza a partir de cada conjunto de AVPs Event-Trigger que o PCRF envia — no CCA-Initial, em um CCA-Update ou em um **RAR** iniciado pelo PCRF. Um conjunto de Event-Trigger presente substitui o conjunto armado atual; uma mensagem que não transporta AVP Event-Trigger deixa o conjunto armado inalterado. Assim, um PCRF pode iniciar uma sessão sem relatório de localização e depois ativá-lo (ou desativá-lo) com um RAR, e vice-versa.

Event-Triggers Suportados

Event-Trigger	Valor	Dispara quando…	AVP Reportado
`RAT_CHANGE`	2	Mudanças no tipo RAT (por exemplo, EUTRAN → UTRAN)	`RAT-Type`
`PLMN_CHANGE`	4	Mudanças na rede de serviço (MCC/MNC)	`3GPP-SGSN-MCC-MNC`
`USER_LOCATION_CHANGE`	13	Qualquer mudança nas Informações de Localização do Usuário	`3GPP-User-Location-Info`
`UE_TIME_ZONE_CHANGE`	25	Mudanças no fuso horário do UE	`3GPP-MS-TimeZone`
`TAI_CHANGE`	26	Mudanças na Identidade da Área de Rastreamento	`3GPP-User-Location-Info`
`ECGI_CHANGE`	27	Mudanças na Identidade Global da Célula E-UTRAN	`3GPP-User-Location-Info`

Múltiplos triggers podem disparar ao mesmo tempo (por exemplo, mover para uma nova célula em uma nova área de rastreamento dispara `USER_LOCATION_CHANGE`, `TAI_CHANGE` **e** `ECGI_CHANGE`); todos os triggers disparados e armados são listados nos AVPs `Event-Trigger` do CCR-Update.

De onde vem a localização

- **CCR-Initial** agora inclui o `3GPP-User-Location-Info` inicial do assinante, dando ao PCRF uma linha de base.

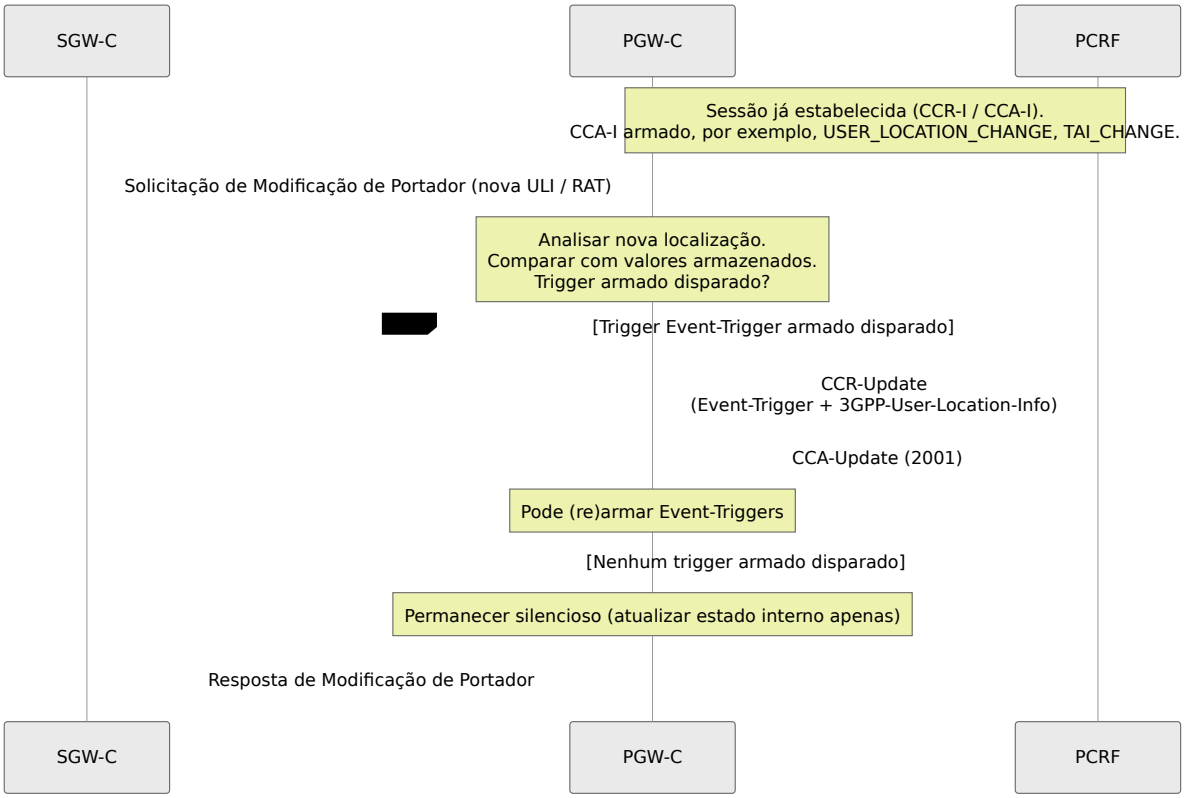
- Durante a sessão, uma **Solicitação de Modificação de Portador GTP-C** (S5/S8) transporta as Informações de Localização do Usuário atualizadas, RAT-Type, Rede de Serviço e Fuso Horário do UE. O PGW-C analisa esses dados, compara-os com os valores armazenados e relata qualquer mudança armada.
- IEs **ausentes** de uma Solicitação de Modificação de Portador mantêm seu valor anterior, de modo que uma Solicitação de Modificação de Portador parcial nunca parece uma mudança de localização espúria.

Codificação do 3GPP-User-Location-Info

O AVP é codificado de acordo com **3GPP TS 29.061 §16.4.7.2**: um tipo de *Localização Geográfica* de um octeto seguido pelos campos de localização, com identidades PLMN codificadas de acordo com 3GPP TS 24.008 §10.5.1.3. Tipos suportados:

Tipo	Valor	Conteúdo
CGI	0	PLMN + LAC + CI (2G)
SAI	1	PLMN + LAC + SAC (3G)
RAI	2	PLMN + LAC + RAC
TAI	128	PLMN + TAC (LTE)
ECGI	129	PLMN + ECI (LTE)
TAI e ECGI	130	PLMN + TAC + PLMN + ECI (LTE, preferido)

Fluxo de Mensagens



> **Robustez:** relatar a mudança ao PCRF nunca bloqueia a Resposta de Modificação de Portador GTP-C. Se o CCR-Update expirar ou o PCRF retornar um erro, o PGW-C registra e ainda responde ao SGW-C — o caminho de dados do assinante não é interrompido por uma falha de relatório de política.

Inspecionando os triggers armados via API

A API de Sessões (`GET /api/sessions` e `GET /api/sessions/:imsi`) retorna um despejo completo e seguro em JSON da sessão sob a chave `state`, além dos campos de nível superior curados. Os Event-Triggers armados pelo PCRF são visíveis em `state.gx.requested_event_triggers`, juntamente com o restante do estado da sessão Gx (`session_id`, `cc_request_number`, host/reino de destino) e o atual `state.uli`. Isso torna possível confirmar a partir de ferramentas de operações se um assinante está realmente inscrito para o relatório de localização.

Substituindo os triggers armados via API

O conjunto armado também pode ser alterado diretamente em uma sessão ativa com **`PATCH /api/sessions/:imsi`** (útil para testar o recurso sem um PCRF, ou para forçar o relatório ligado/desligado). O corpo é **mesclado profundamente no estado da sessão**, então o caminho reflete a estrutura retornada por `GET` — os triggers armados vivem sob `gx`:

```
PATCH /api/sessions/310260000000001
Content-Type: application/json

{ "gx": { "requested_event_triggers": [13, 26, 27] } }
```

- O valor é uma lista de códigos de Event-Trigger Gx; ele substitui o conjunto armado (outros campos `gx` são deixados inalterados pela mesclagem).
- Uma lista vazia (`[]`) desarma todo o relatório de localização para a sessão.
- A resposta ecoa o estado da sessão atualizado e sanitizado.

Este endpoint não possui **proteções**: o corpo é mesclado profundamente na sessão ativa, então *qualquer* campo pode ser alterado — incluindo para um valor que quebra a sessão. Alterações que têm efeitos a jusante são propagadas para o UPF. Atualizar o APN-AMBR ou a QoS do portador padrão recalcula o QER do portador padrão (QCI / MBR / GBR) e o empurra como um QER de Atualização em uma Modificação de Sessão PFCP:

```
PATCH /api/sessions/310260000000001
Content-Type: application/json

{ "ambr": { "uplink": 200000, "downlink": 200000 } }
```

```
PATCH /api/sessions/310260000000001
Content-Type: application/json

{ "default_bearer_qos": { "qci": 5, "mbr_ul": 300000, "mbr_dl":
250000 } }
```

Quando `default_bearer_qos` fornece MBR/GBR explícitos (portadores GBR), esses dirigem os limites de taxa do QER; caso contrário, o APN-AMBR é usado como o MBR do portador padrão.

> Nota: substituir os triggers armados altera a visão local do PGW-C sobre o que o PCRF armou. O PCRF ainda pode rearmar o conjunto mais tarde via um CCA ou RAR, o que substituirá uma substituição manual.

Configuração

Configuração Básica do Gx

Edite `config/runtime.exs`:

```
config :pgw_c,
  diameter: %{
    # Endereço IP para escutar conexões Diameter
    listen_ip: "0.0.0.0",

    # Identidade Diameter do PGW-C (Origin-Host)
    host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org",

    # Reino Diameter do PGW-C (Origin-Realm)
    realm: "epc.mnc999.mcc999.3gppnetwork.org",

    # Lista de pares PCRF
    peer_list: [
      %{
        # Identidade Diameter do PCRF
        host: "pcrf.epc.mnc999.mcc999.3gppnetwork.org",

        # Reino do PCRF (geralmente o mesmo que o reino do PGW-C)
        realm: "epc.mnc999.mcc999.3gppnetwork.org",

        # Endereço IP do PCRF
        ip: "10.0.0.30",

        # Se o PGW-C inicia a conexão com o PCRF
        # true = PGW-C conecta ao PCRF
        # false = Espera o PCRF se conectar
        initiate_connection: true
      }
    ]
  }
```

Múltiplos Pares PCRF

Para redundância ou distribuição geográfica:

```
config :pgw_c,
  diameter: %{
    listen_ip: "0.0.0.0",
    host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org",
    realm: "epc.mnc999.mcc999.3gppnetwork.org",
    peer_list: [
      %{
        host: "pcrf-primary.example.com",
        realm: "epc.mnc999.mcc999.3gppnetwork.org",
        ip: "10.0.1.30",
        initiate_connection: true
      },
      %{
        host: "pcrf-backup.example.com",
        realm: "epc.mnc999.mcc999.3gppnetwork.org",
        ip: "10.0.2.30",
        initiate_connection: true
      }
    ]
  }
```

Balanceamento de Carga:

- O protocolo Diameter lida com a seleção de pares
- Solicitações distribuídas com base na disponibilidade
- Failover automático em caso de falha do par

Resolução de Nome de Host

Identidades Diameter devem ser FQDNs (Nomes de Domínio Totalmente Qualificados):

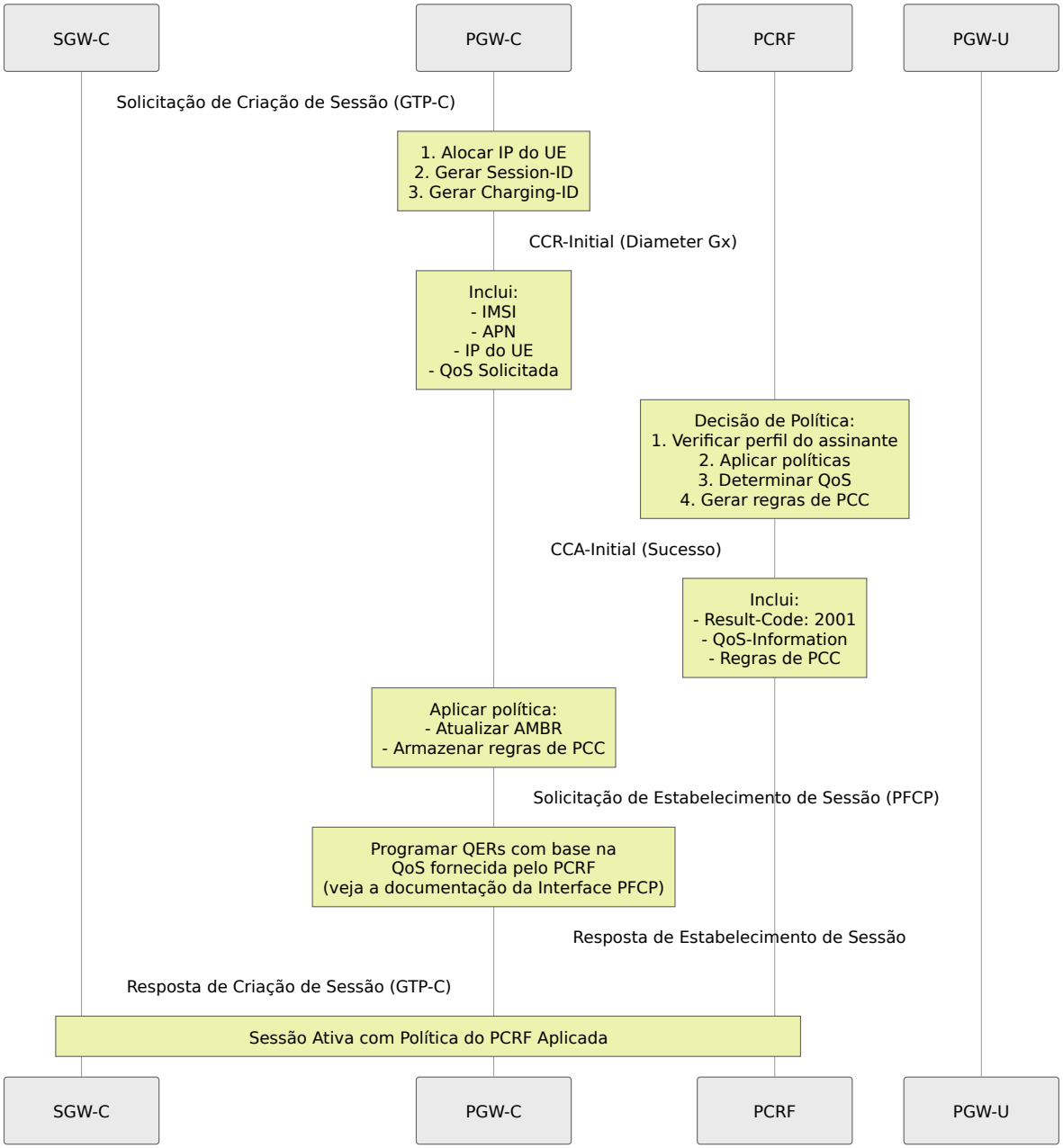
```
# CORRETO - formato FQDN
host: "pgw_c.epc.mnc999.mcc999.3gppnetwork.org"

# INCORRETO - Não é uma Identidade Diameter válida
host: "pgw_c"
host: "10.0.0.20"  # Endereços IP não são permitidos
```

Formato do Reino:

- Deve ser um nome de domínio válido
- Normalmente corresponde ao formato PLMN 3GPP: `epc.mncXXX.mccYYY.3gppnetwork.org`

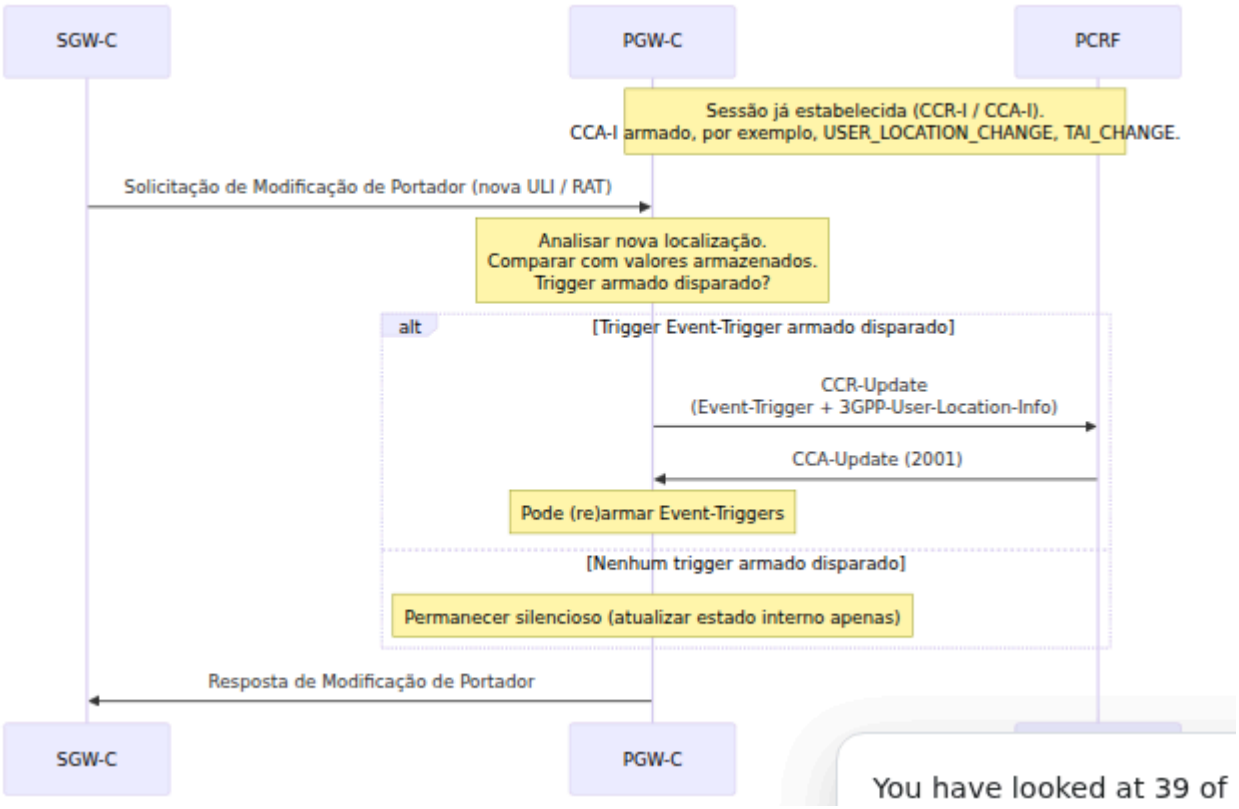
Fluxos de Mensagens

Estabelecimento de Sessão Bem-Sucedido



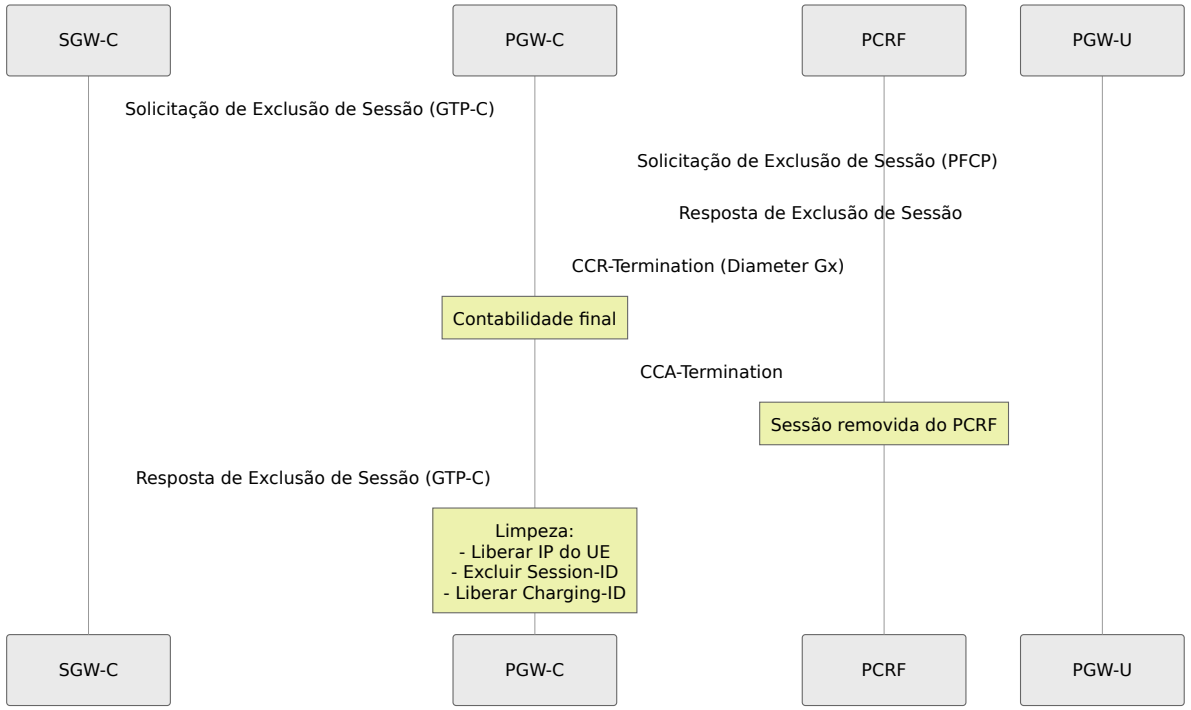
Nota: Os parâmetros de QoS do PCRF são traduzidos em QERs (Regras de Aplicação de QoS) e programados no PGW-U via PFCP. Veja Interface PFCP para detalhes do QER.

Atualização de Política (Iniciada pela Rede)



You have looked at 39 of

Término da Sessão



Tratamento de Erros

Códigos de Resultado

O PGW-C lida com vários códigos de resultado Diameter nas mensagens CCA:

Códigos de Sucesso:

Código	Nome	Ação
2001	DIAMETER_SUCCESS	Continuar o estabelecimento da sessão

Falhas Permanentes (5xxx):

Código	Nome	Ação do PGW-C
5002	DIAMETER_UNKNOWN_SESSION_ID	Registrar erro, falhar sessão
5030	DIAMETER_USER_UNKNOWN	Rejeitar sessão (Usuário Desconhecido)
5140	DIAMETER_ERROR_INITIAL_PARAMETERS	Registrar erro, tentar novamente ou falhar
5003	DIAMETER_AUTHORIZATION_REJECTED	Rejeitar sessão (Não Autorizado)

Falhas Transitórias (4xxx):

Código	Nome	Ação do PGW-C
4001	DIAMETER_AUTHENTICATION_REJECTED	Tentar novamente ou falhar sessão
4010	DIAMETER_TOO_BUSY	Tentar novamente com espera
4012	DIAMETER_UNABLE_TO_COMPLY	Registrar erro, pode tentar novamente

Códigos de Resultado Experimentais

Códigos de erro específicos do fornecedor:

```
Experimental-Result (Agrupado)
├── Vendor-Id: 10415 (3GPP)
└── Experimental-Result-Code: <código específico do fornecedor>
```

Códigos Experimentais Comuns do 3GPP:

Código	Nome	Significado
5065	IP_CAN_SESSION_NOT_AVAILABLE	PCRF não pode estabelecer sessão
5143	INVALID_SERVICE_INFORMATION	Dados de serviço inválidos

Tratamento de Timeouts

Timeout do CCR-I:

Se o PCRF não responder ao CCR-Initial dentro do timeout:

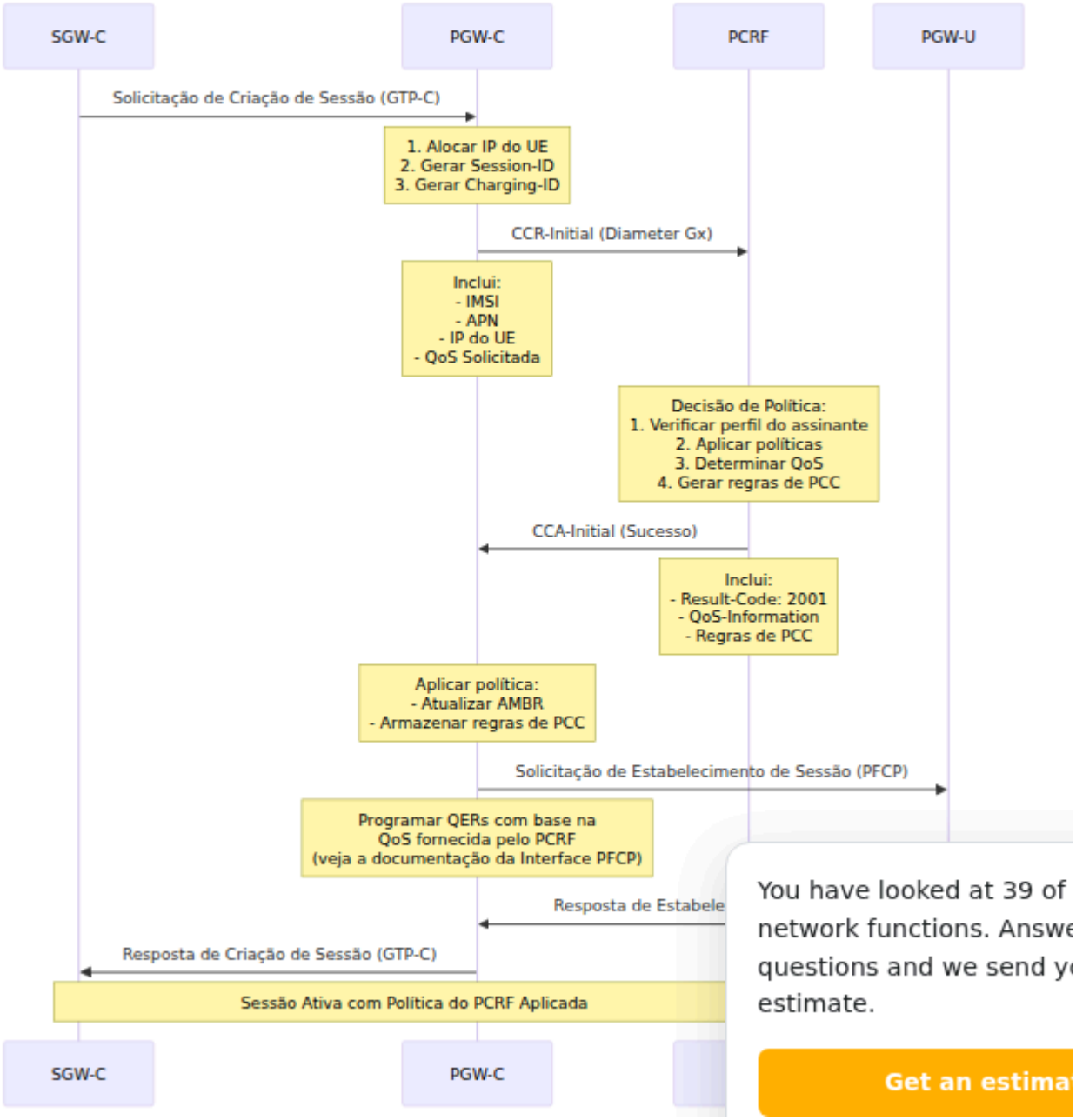
```
1. PGW-C aguarda o timeout configurado (por exemplo, 5 segundos)
2. Se nenhum CCA recebido:
   - Registrar: "timeout CCR-Initial para Session-ID: ..."
   - Responder ao SGW-C com causa de erro
   - Limpar recursos alocados
3. SGW-C recebe: Resposta de Criação de Sessão (Causa: Par Remoto
Não Respondendo)
```

Resposta de Erro ao SGW-C:

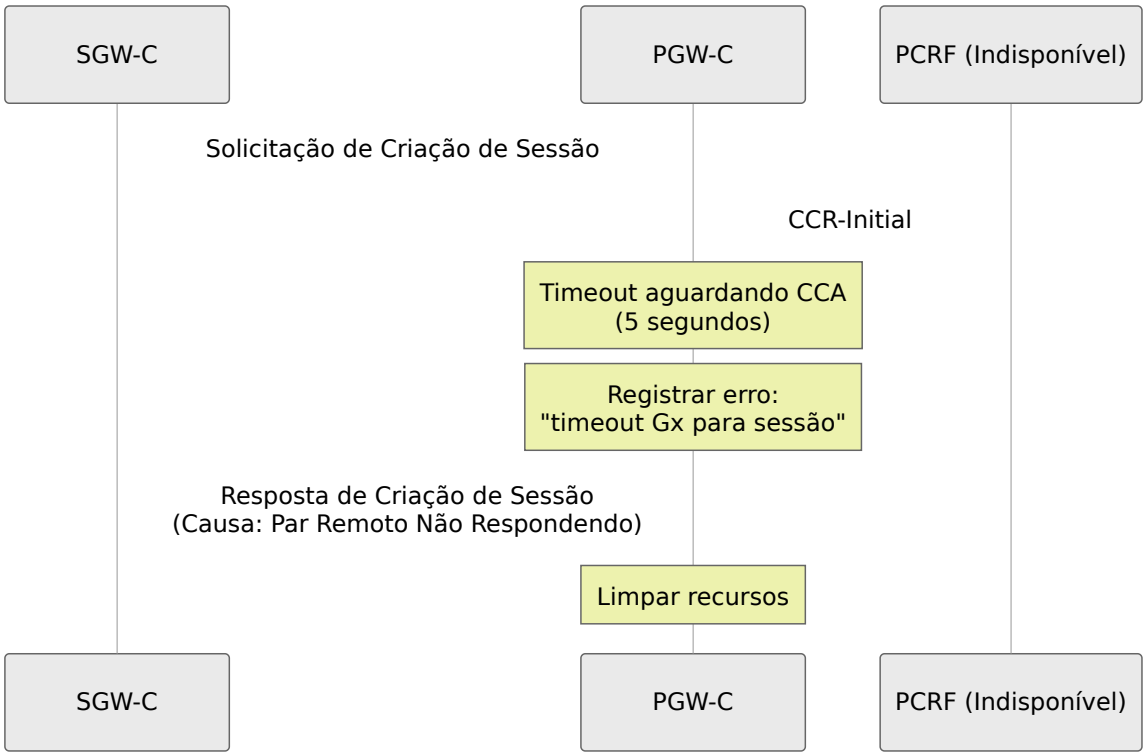
Quando o CCR-Initial expira, o PGW-C envia uma Resposta de Criação de Sessão com o código de causa `:remote_peer_not_responding` para o SGW-C.

Cenários de Falha

Cenário 1: PCRF Rejeita a Sessão (Usuário Desconhecido)



Cenário 2: PCRF Temporariamente Indisponível



Solução de Problemas

Problemas Comuns

1. Falha na Conexão do Par Diameter

Sintomas:

- Log: "Par Diameter não conectado"
- Nenhum CCR-Initial enviado

Possíveis Causas:

- PCRF não acessível
- IP do PCRF incorreto na configuração
- Firewall bloqueando a porta Diameter (3868)
- Identidades Diameter incorretas (host/reino)

Resolução:

```
# Testar conectividade de rede
ping <pcrf_ip>

# Testar porta Diameter (TCP 3868)
telnet <pcrf_ip> 3868

# Verificar configuração da identidade Diameter
# Garantir que host e reino sejam FQDNs, não IPs
```

Verificar Configuração:

```
config :pgw_c,
  diameter: %{
    # Deve ser FQDN, não IP
    host: "pgw_c.epc.mnc999.mcc999.3gppnetwork.org",
    realm: "epc.mnc999.mcc999.3gppnetwork.org",
    peer_list: [
      %{
        host: "pcrf.epc.mnc999.mcc999.3gppnetwork.org",
        ip: "10.0.0.30"
      }
    ]
  }
```

2. Timeouts do CCR-Initial

Sintomas:

- Falha na Solicitação de Criação de Sessão
- Log: "timeout CCR-Initial"

Possíveis Causas:

- PCRF sobrecarregado
- Latência de rede
- PCRF não respondendo a este Session-ID

Resolução:

1. Verificar logs do PCRF para erros
2. Verificar se o PCRF está processando solicitações
3. Verificar latência da rede: `ping <pcrf_ip>`
4. Aumentar o timeout se a latência da rede for alta

3. Sessões Rejeitadas pelo PCRF

Sintomas:

- CCA-Initial com Result-Code != 2001
- Falha na Resposta de Criação de Sessão

Códigos de Resultado Comuns:

Código de Resultado	Causa Provável	Resolução
5030	IMSI não no banco de dados de assinantes	Provisionar assinante no HSS/SPR
5003	Autorização rejeitada	Verificar permissões do assinante
4010	PCRF muito ocupado	Tentar novamente ou adicionar capacidade ao PCRF

Verificar Logs:

```
# Logs do PGW-C mostram:
[error] Erro Diameter Gx: Result-Code 5030 (DIAMETER_USER_UNKNOWN)
[error] IMSI 310260999999999 rejeitado pelo PCRF
```

4. QoS Não Aplicada

Sintomas:

- Sessão estabelecida, mas QoS errada
- Taxas de bits não correspondem aos valores esperados

Etapas de Depuração:

1. **Verificar CCA-Initial:**
    - Verificar se o AVP `QoS-Information` está presente
    - Verificar os valores de `APN-Aggregate-Max-Bitrate-UL/DL`
2. **Verificar Estabelecimento de Sessão PFCP:**
    - Verificar se o QER foi criado com os valores de MBR corretos
    - Verificar logs do PGW-U para instalação do QER
3. **Verificar Política do PCRF:**
    - Verificar configuração do PCRF
    - Verificar se o perfil do assinante inclui QoS correta

5. Problemas de Roteamento Diameter

Sintomas:

- Mensagens Diameter não chegam ao PCRF
- Log: "Sem rota para Destination-Realm"

Causa:

- Desvio entre configuração e mensagens

Resolução:

Garantir consistência:

```
# Todos devem corresponder
config :pgw_c,
  diameter: %{
    realm: "epc.mnc999.mcc999.3gppnetwork.org",  # Reino do PGW-C
    peer_list: [
      %{
        realm: "epc.mnc999.mcc999.3gppnetwork.org"  # Reino do
PCRF (geralmente o mesmo)
      }
    ]
  }
```

No CCR-Initial:

```
Origin-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
Destination-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
```

Monitorando a Saúde do Gx

Métricas Chave:

```
# Taxas de mensagens Gx
rate(gx_inbound_messages_total{message_type="gx_CCA"}[5m])
rate(gx_outbound_messages_total{message_type="gx_CCR"}[5m])

# Taxas de erro Gx
rate(gx_inbound_errors_total[5m])

# Taxa de sucesso de respostas Gx (nova métrica)
sum(rate(gx_outbound_responses_total{result_code_class="2xxx"}
[5m])) /
sum(rate(gx_outbound_responses_total[5m])) * 100

# Falhas de resposta Gx por host do PCRF
rate(gx_outbound_responses_total{result_code_class!="2xxx"}[5m])
by (diameter_host)

# Contagem de sessões Gx
session_id_registry_count

# Duração do tratamento de mensagens Gx
histogram_quantile(0.95,
rate(gx_inbound_handling_duration_bucket[5m]))
```

Métricas de Resposta por Classe de Código de Resultado:

A métrica `gx_outbound_responses_total` fornece visibilidade detalhada sobre as respostas Diameter enviadas para pares PCRF, categorizadas por:

- **`message_type`**: Tipo de mensagem de resposta (`gx_RAA`, `gx_CCA`)
- **`result_code_class`**: Categoria do código de resultado (`2xxx`, `3xxx`, `4xxx`, `5xxx`)
- **`diameter_host`**: Par PCRF recebendo a resposta

Exemplos de Alertas:

```yaml
# Alerta sobre alta taxa de erro Gx
- alert: GxErrorRateHigh
  expr: rate(gx_inbound_errors_total[5m]) > 0.1
  for: 5m
  annotations:
    summary: "Alta taxa de erro Gx detectada"

# Alerta sobre alta taxa de falhas de resposta Gx
- alert: GxResponseFailureRate
  expr: |

sum(rate(gx_outbound_responses_total{result_code_class!="2xxx"}
[5m])) /
    sum(rate(gx_outbound_responses_total[5m])) > 0.1
  for: 5m
  annotations:
    summary: "Alta taxa de falhas de resposta Gx"
    description: "Mais de 10% das respostas Gx são falhas"

# Alerta sobre falhas específicas do PCRF
- alert: GxPCRFFailures
  expr:
rate(gx_outbound_responses_total{result_code_class=~"4xxx|5xxx"}
[5m]) by (diameter_host) > 0.05
  for: 3m
  annotations:
    summary: "PCRF {{ $labels.diameter_host }} recebendo respostas
de falha"
    description: "Alta taxa de falha para o host PCRF"

# Alerta sobre rejeição de sessão
- alert: GxSessionRejection
  expr: rate(gx_inbound_errors_total{result_code="5030"}[5m]) >
0.01
  for: 5m
  annotations:
    summary: "PCRF rejeitando sessões (USER_UNKNOWN)"
```

Registro de Depuração

Ativar registro verbose do Diameter:

```
# config/runtime.exs
config :logger, level: :debug

# Ou em tempo de execução
iex> Logger.configure(level: :debug)
```

Procurar por:

- `[debug] Enviando CCR-Initial para Session-ID: ...`
- `[debug] Recebido CCA-Initial: Result-Code 2001`
- `[error] Erro Diameter: ...`

Monitoramento de Pares Diameter

A conectividade Diameter Gx com o PCRF é monitorada através da **API OAM REST**. A API é servida sobre HTTPS/TLS em `https://<host>:8443` com todas as rotas sob o prefixo `/api`. A documentação interativa da API (Swagger UI) está disponível em `https://<host>:8443/api/docs`.

Endpoints de Par Diameter

```
GET /api/diameter                      # todos os pares
configurados + estado ao vivo
GET /api/diameter?status=disconnected # filtrar por estado de
conexão
GET /api/diameter/<origin-host>       # um par por Origin-Host
```

Exemplo:

```
curl -k https://localhost:8443/api/diameter
```

`GET /api/diameter` retorna cada par Diameter configurado e seu estado de conexão atual. Adicione `?status=connected` ou `?status=disconnected` para

filtrar por estado de conexão, ou anexe o Origin-Host do par a `GET /api/diameter/<origin-host>` para buscar um único par.

Para cada par, os seguintes campos são retornados:

Campo	Descrição
`host`	Identidade Diameter (Origin-Host) do par
`status`	Estado da conexão (por exemplo, Conectado / Desconectado / Conectando)
`ip`	Endereço IP do par
`port`	Porta do par
`transport`	TCP ou SCTP
`connection_initiation`	Se o OmniPGW inicia (saída) ou aceita (entrada) a conexão
`realm`	Reino Diameter reportado pelo par no CEA
`product_name`	String do produto do fornecedor reportada pelo par no CEA
`application_ids`	IDs de aplicação Diameter negociados no CEA (Gx = 16777238)

Uma implantação saudável mostra todos os pares PCRF configurados como **Conectados**. A perda de todos os pares Gx impede a instalação de políticas e bloqueia a criação de novas sessões.

Casos de Uso Operacionais

Monitorar Conectividade do PCRF:

```
curl -k https://localhost:8443/api/diameter
```

Verifique se todos os pares PCRF mostram `Conectado`, se `connection_initiation` corresponde à configuração, e se `application_ids` incluem Gx (16777238).

Solução de Problemas de Falhas na Criação de Sessões (Problemas Gx): Quando sessões de usuários falham com erros de "timeout do PCRF", consulte o par afetado:

```
curl -k https://localhost:8443/api/diameter/<origin-host>
```

- `status` é `Disconnected` → verifique a conectividade de rede, verifique se o PCRF está em execução e confirme se as regras de firewall permitem TCP 3868.
- `status` é `Connected`, mas as sessões ainda falham → o problema está no nível da aplicação (verifique os logs); o PCRF pode estar rejeitando assinantes.

Verificar Configuração Diameter: Após configurar um novo par PCRF, confirme se ele aparece e alcança `Connected`:

```
curl -k https://localhost:8443/api/diameter/<origin-host>
```

Verifique se `realm` corresponde à configuração, se `application_ids` incluem Gx, e se `product_name` mostra o identificador esperado do PCRF.

Monitorar Failover: Quando um PCRF primário falha, `GET /api/diameter?status=disconnected` lista-o como `Disconnected` enquanto o PCRF de backup permanece `Connected`. Novas sessões usam automaticamente o backup; quando o primário se recupera, seu `status` retorna a `Connected`.

Detectar Problemas de Roteamento Diameter:

- O par mostra `Connected`, mas `realm` está errado
- `application_ids` não inclui Gx (16777238)

- `product_name` não corresponde ao esperado do PCRF

Identificar Incompatibilidades de Configuração: Se `connection_initiation` reporta que o par está iniciando enquanto a configuração define `initiate_connection: true`, ambos os lados estão tentando iniciar, o que pode causar condições de corrida de conexão.

Integração com Métricas

A API OAM fornece o estado do par em tempo real; combine-a com o Prometheus para:

- Taxas históricas de erro Gx
- Contagens de mensagens CCR/CCA
- Tendências de latência

`GET /api/diameter` responde "Está funcionando agora?"; métricas respondem "Como tem funcionado ao longo do tempo?"

Documentação Relacionada

Configuração e Política

- **Guia de Configuração** - Configuração Diameter, configuração de pares PCRF
- **Interface PFCP** - Aplicação de QoS via QERs a partir de regras de PCC
- **Gerenciamento de Sessão** - Ciclo de vida da sessão com integração de política
- **QoS & Gerenciamento de Portadores** - Configuração detalhada de QoS e configuração de portadores

Integração de Cobrança

- **Interface Diameter Gy** - Cobrança online acionada por regras de PCC

- **Formato de CDR de Dados** - Registros de cobrança offline com informações de política
- **Configuração PCO** - Entrega P-CSCF para controle de política IMS

Operações

- **Guia de Monitoramento** - Métricas Gx, rastreamento de políticas, alertas de conectividade do PCRF
- **Interface S5/S8** - Integração de gerenciamento de portadores com política

---

Voltar ao Guia de Operações

Cobrança Online Diameter (Interface Gy/Ro)

Interface do Sistema de Cobrança Online (OCS)

Índice

1. Visão Geral
2. Arquitetura de Cobrança 3GPP
3. Fundamentos da Interface Gy/Ro
4. Mensagens de Controle de Crédito
5. Fluxos de Cobrança Online
6. Controle de Cobrança de Bearer
7. Controle de Crédito para Múltiplos Serviços
8. Configuração
9. Fluxos de Mensagens
10. Tratamento de Erros
11. Integração com Gx
12. Solução de Problemas

Visão Geral

A **interface Gy** (também chamada de **interface Ro** em contextos IMS) conecta o PGW-C ao **Sistema de Cobrança Online (OCS)** para controle de crédito em tempo real. Isso possibilita:

- **Cobrança Pré-paga** - Autorização e dedução de crédito em tempo real

- **Controle de Crédito em Tempo Real** - Conceder cota antes da entrega do serviço
- **Cobrança Baseada em Serviço** - Cobrança diferente para voz, dados, SMS, etc.
- **Atualizações Imediatas da Conta** - Atualizações do saldo de crédito em tempo real
- **Negação de Serviço** - Bloquear serviço quando o crédito se esgotar

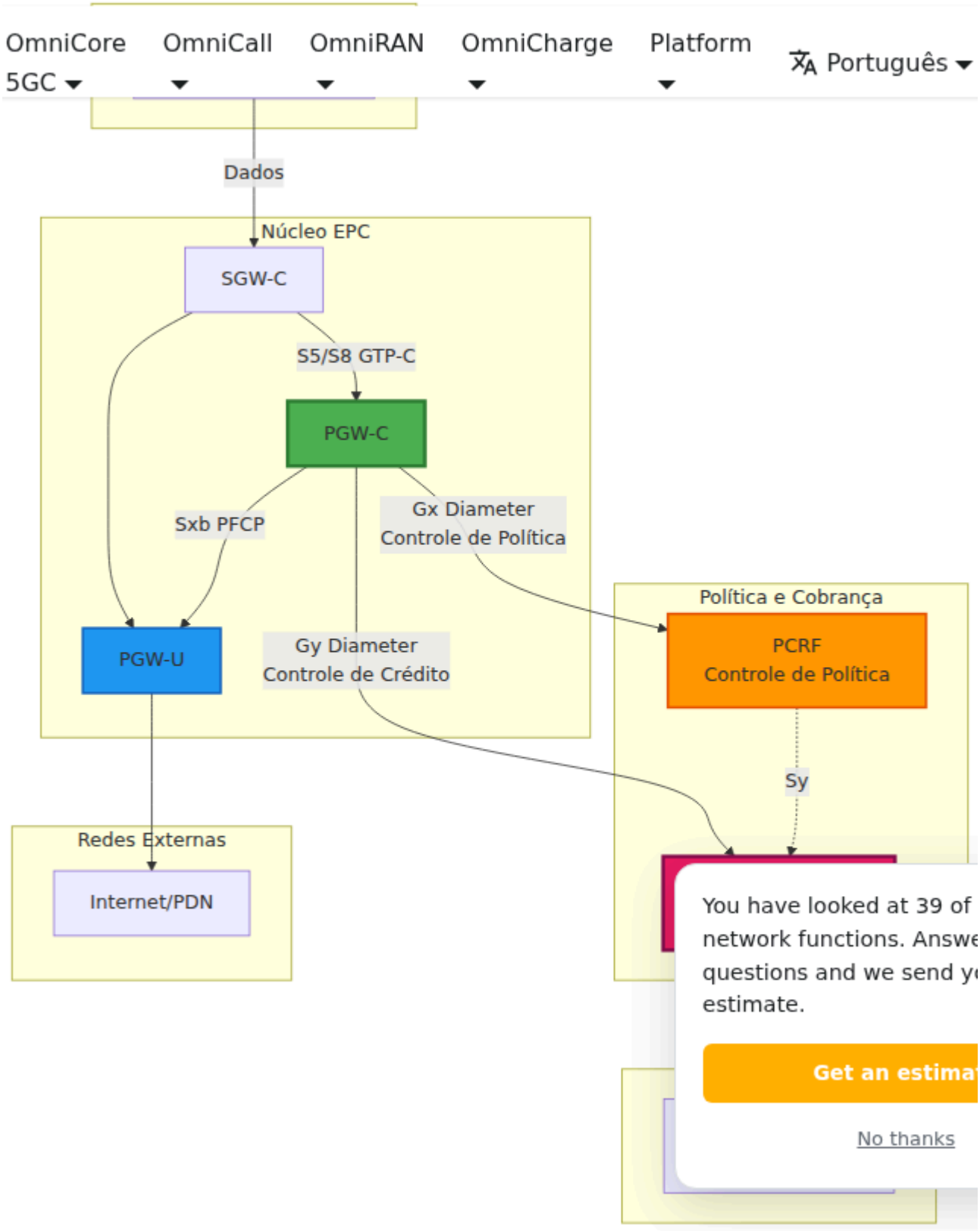
Cobrança Online vs. Offline

Aspecto	Cobrança Online (Gy/Ro)	Cobrança Offline (Gz/Rf)
Tempo	Em tempo real, antes do serviço	Após a entrega do serviço
Caso de Uso	Assinantes pré-pagos	Assinantes pós-pagos
Verificação de Crédito	Sim, antes de conceder o serviço	Não, fatura gerada posteriormente
Sistema	OCS (Sistema de Cobrança Online)	CGF/CDF (Função de Dados de Cobrança)
Risco	Sem perda de receita	Risco de faturas não pagas
Complexidade	Alta (requisitos em tempo real)	Menor (processamento em lote)
Impacto ao Usuário	Serviço negado se não houver crédito	Serviço sempre disponível

Veja também: Formato de CDR de Dados para registros de cobrança offline (faturamento pós-pago)

Veja também: Gerenciamento de Sessões para o ciclo de vida completo da sessão PDN, incluindo integração de cobrança

Gy na Arquitetura de Rede

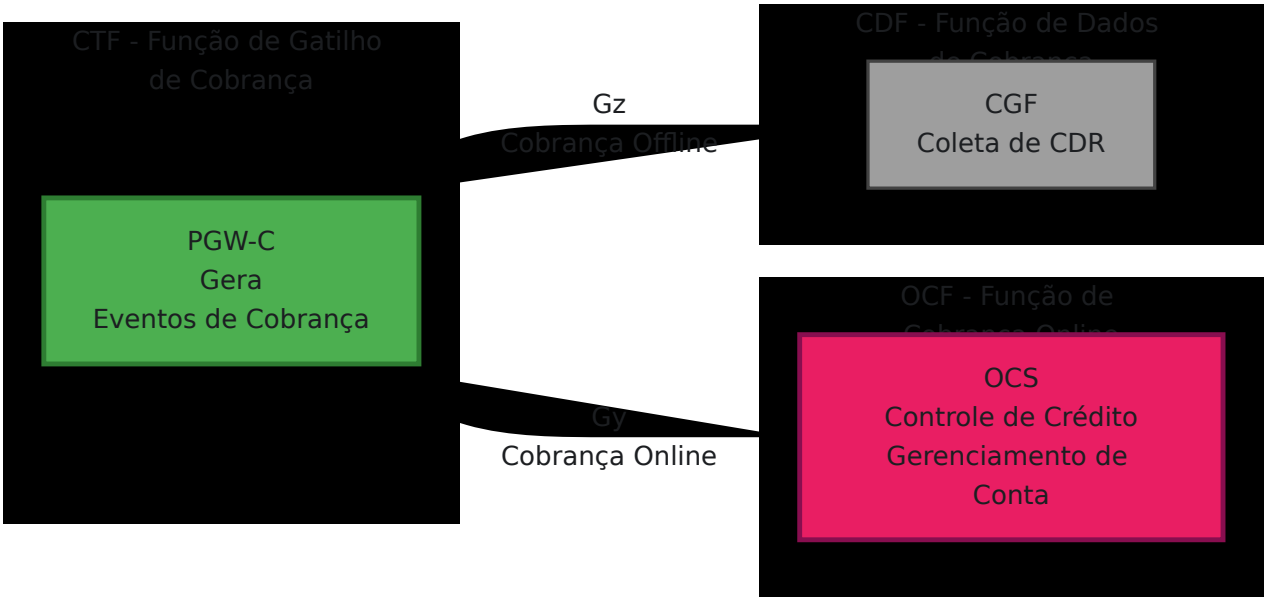


Funções Principais

Função	Descrição
Autorização de Crédito	Solicitar cota ao OCS antes de permitir tráfego
Gerenciamento de Cota	Rastrear unidades concedidas (bytes, tempo, eventos)
Detecção de Depleção de Crédito	Monitorar cota restante
Re-autorização	Solicitar cota adicional quando o limite for atingido
Terminação de Serviço	Parar serviço quando o crédito se esgotar
Liquidação Final	Relatar uso real ao final da sessão

Arquitetura de Cobrança 3GPP

Pontos de Referência de Cobrança



Função de Gatilho de Cobrança (CTF)

O PGW-C atua como um **CTF (Função de Gatilho de Cobrança)**, responsável por:

1. **Detectar eventos cobrados** - Início da sessão, uso de dados, fim da sessão
2. **Solicitar autorização de crédito** - Antes de permitir o serviço
3. **Rastrear consumo de cota** - Monitorar unidades concedidas
4. **Gerar eventos de cobrança** - Acionar solicitações de crédito
5. **Impor controle de crédito** - Bloquear tráfego quando a cota se esgotar

Função de Cobrança Online (OCF)

O OCS implementa a **OCF (Função de Cobrança Online)**:

1. **Gerenciamento de saldo da conta** - Rastrear crédito do assinante
2. **Classificação** - Determinar preço por unidade (por MB, por segundo, etc.)
3. **Reserva de crédito** - Reservar crédito para cota concedida

4. **Dedução de crédito** - Deduzir após relatório de uso
5. **Decisões de política** - Conceder ou negar com base no saldo

---

Fundamentos da Interface Gy/Ro

Referência 3GPP

- **Especificação:** 3GPP TS 32.299 (Arquitetura de cobrança)
- **Protocolo:** 3GPP TS 32.251 (Cobrança de domínio PS)
- **ID da Aplicação Diameter:** 4 (Gy/Ro - Aplicação de Controle de Crédito)
- **Protocolo Base:** RFC 4006 (Aplicação de Controle de Crédito Diameter)

Conceito de Sessão

Cada conexão PDN de UE que requer cobrança online tem uma **sessão Gy/Ro** identificada por um **Session-ID**. Esta sessão:

- Criada quando o bearer requer cobrança online (CCR-Initial)
- Atualizada quando a cota é consumida (CCR-Update)
- Terminada quando a sessão termina (CCR-Termination)

Formato do ID da Sessão

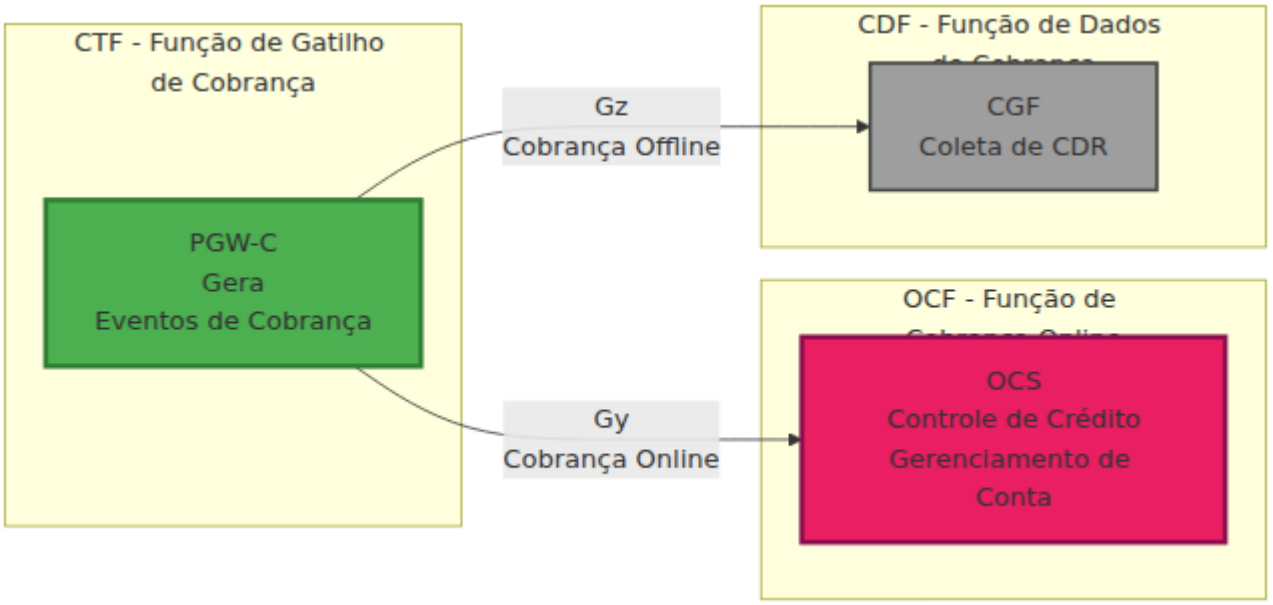
```
Session-ID: <Origin-Host>;<high32>;<low32>[;<optional>]
Exemplo: omni-
pgw_c.epc.mnc999.mcc999.3gppnetwork.org;9876543210;12345;gy
```

Componentes:

- **Origin-Host:** Identidade Diameter do PGW-C
- **high32:** 32 bits altos do identificador único
- **low32:** 32 bits baixos do identificador único
- **optional:** Identificador adicional (por exemplo, "gy" para distinguir do Gx)

Mensagens de Controle de Crédito

Tipos de Mensagens



CCR-Initial (Solicitação de Controle de Crédito - Inicial)

Quando: UE cria uma conexão PDN e o bearer requer cobrança online

Propósito:

- Solicitar autorização de crédito inicial ao OCS
- Reservar cota para entrega do serviço
- Estabelecer sessão Gy/Ro

Principais AVPs Enviados pelo PGW-C:

Nome do AVP	Código do AVP	Tipo	Descrição
Session-Id	263	UTF8String	Identificador único da sessão Gy
Auth-Application-Id	258	Unsigned32	4 (Controle de Crédito)
Origin-Host	264	DiamIdent	Identidade Diameter do PGW-C
Origin-Realm	296	DiamIdent	Reino Diameter do PGW-C
Destination-Realm	283	DiamIdent	Reino do OCS
CC-Request-Type	416	Enumerated	1 = INITIAL_REQUEST
CC-Request-Number	415	Unsigned32	Número de sequência (começa em 0)
Subscription-Id	443	Grouped	Identificador da UE (IMSI/MSISDN)
Service-Context-Id	461	UTF8String	Identificador do contexto de cobrança
Multiple-Services-Credit-Control	456	Grouped	Solicitações de crédito específicas do serviço
Requested-Service-Unit	437	Grouped	Cota solicitada (bytes, tempo, etc.)
Used-Service-Unit	446	Grouped	Cota utilizada (0 para inicial)

Nome do AVP	Código do AVP	Tipo	Descrição
Service-Identifier	439	Unsigned32	Identificador do tipo de serviço
Rating-Group	432	Unsigned32	Identificador da categoria de cobrança

Exemplo de Estrutura CCR-I:

```
CCR (Código do Comando: 272, Solicitação)
├── Session-Id: "pgw_c.example.com;123;456;gy"
├── Auth-Application-Id: 4
├── Origin-Host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org"
├── Origin-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── Destination-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── CC-Request-Type: INITIAL_REQUEST (1)
├── CC-Request-Number: 0
├── Subscription-Id (Grouped)
│   ├── Subscription-Id-Type: END_USER_IMSI (1)
│   └── Subscription-Id-Data: "310260123456789"
├── Subscription-Id (Grouped)
│   ├── Subscription-Id-Type: END_USER_E164 (0)
│   └── Subscription-Id-Data: "15551234567"
├── Service-Context-Id: "32251@3gpp.org"
├── Multiple-Services-Credit-Control (Grouped)
│   ├── Service-Identifier: 1
│   ├── Rating-Group: 100
│   └── Requested-Service-Unit (Grouped)
│       └── CC-Total-Octets: 10000000 (solicitar 10 MB)
└── Used-Service-Unit (Grouped)
    └── CC-Total-Octets: 0 (sem uso ainda)
```

CCA-Initial (Resposta de Controle de Crédito - Inicial)

Enviado por: OCS em resposta ao CCR-I

Propósito:

- Conceder ou negar autorização de crédito
- Fornecer cota para entrega do serviço
- Especificar parâmetros de classificação e cobrança

Principais AVPs Recebidos pelo PGW-C:

Nome do AVP	Código do AVP	Descrição
Result-Code	268	Sucesso (2001) ou código de erro
Multiple-Services-Credit-Control	456	Concessões de crédito específicas do serviço
Granted-Service-Unit	431	Cota concedida (bytes, tempo, etc.)
Validity-Time	448	Período de validade da cota (segundos)
Result-Code	268	Código de resultado por serviço
Final-Unit-Indication	430	Ação quando a cota se esgotar
Volume-Quota-Threshold	-	Limite para re-autorização

Exemplo de Resposta de Sucesso:

```
CCA (Código do Comando: 272, Resposta)
├── Session-Id: "pgw_c.example.com;123;456;gy"
├── Result-Code: DIAMETER_SUCCESS (2001)
├── Origin-Host: "ocs.example.com"
├── Origin-Realm: "example.com"
├── Auth-Application-Id: 4
├── CC-Request-Type: INITIAL_REQUEST (1)
├── CC-Request-Number: 0
└── Multiple-Services-Credit-Control (Grouped)
    ├── Result-Code: DIAMETER_SUCCESS (2001)
    ├── Service-Identifier: 1
    ├── Rating-Group: 100
    ├── Granted-Service-Unit (Grouped)
    │   └── CC-Total-Octets: 10000000 (concedido 10 MB)
    ├── Validity-Time: 3600 (cota válida por 1 hora)
    └── Volume-Quota-Threshold: 8000000 (re-autorização em 8 MB
usados, 80%)
```

CCR-Update (Solicitação de Controle de Crédito - Atualização)

Quando:

- Limite de cota concedida atingido (por exemplo, 80% consumido)
- Tempo de validade expira
- Mudança de serviço requer re-autorização
- Mudança de horário de tarifa

Propósito:

- Solicitar cota adicional
- Relatar uso da cota previamente concedida
- Atualizar parâmetros de cobrança

Principais Diferenças em Relação ao CCR-I:

- `CC-Request-Type: UPDATE_REQUEST (2)`
- `CC-Request-Number` incrementado

- `Used-Service-Unit` contém uso real
- `Requested-Service-Unit` para mais cota

Exemplo de Estrutura CCR-U:

```
CCR (Código do Comando: 272, Solicitação)
├── Session-Id: "pgw_c.example.com;123;456;gy"
├── Auth-Application-Id: 4
├── Origin-Host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org"
├── Origin-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── Destination-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── CC-Request-Type: UPDATE_REQUEST (2)
├── CC-Request-Number: 1
└── Multiple-Services-Credit-Control (Grouped)
    ├── Service-Identifier: 1
    ├── Rating-Group: 100
    ├── Used-Service-Unit (Grouped)
    │   └── CC-Total-Octets: 8000000 (8 MB usados até agora)
    └── Requested-Service-Unit (Grouped)
        └── CC-Total-Octets: 10000000 (solicitar mais 10 MB)
```

CCA-Update (Resposta de Controle de Crédito - Atualização)

Enviado por: OCS em resposta ao CCR-U

Propósito:

- Conceder cota adicional (se crédito disponível)
- Reconhecer uso
- Atualizar parâmetros de cobrança

Possíveis Resultados:

1. Mais Cota Concedida:

```
CCA (Atualização)
└── Multiple-Services-Credit-Control
    ├── Result-Code: DIAMETER_SUCCESS (2001)
    ├── Granted-Service-Unit
    │   └── CC-Total-Octets: 10000000 (mais 10 MB)
    └── Validity-Time: 3600
```

2. Cota Final (Crédito Esgotado):

```
CCA (Atualização)
└── Multiple-Services-Credit-Control
    ├── Result-Code: DIAMETER_SUCCESS (2001)
    ├── Granted-Service-Unit
    │   └── CC-Total-Octets: 1000000 (apenas 1 MB restante)
    └── Final-Unit-Indication
        └── Final-Unit-Action: TERMINATE (0)
```

3. Sem Crédito Disponível:

```
CCA (Atualização)
├── Result-Code: DIAMETER_CREDIT_LIMIT_REACHED (4012)
└── Multiple-Services-Credit-Control
    ├── Result-Code: DIAMETER_CREDIT_LIMIT_REACHED (4012)
    └── Final-Unit-Indication
        └── Final-Unit-Action: TERMINATE (0)
```

CCR-Termination (Solicitação de Controle de Crédito - Terminação)

Quando:

- UE se desliga
- Conexão PDN excluída
- Sessão terminada por qualquer motivo

Propósito:

- Relatório final de uso
- Fechar sessão Gy/Ro
- Liquidação final

Principais Diferenças:

- `CC-Request-Type: TERMINATION_REQUEST (3)`
- `Used-Service-Unit` contém uso final
- Sem `Requested-Service-Unit` (não é mais necessária cota)
- Inclui `Termination-Cause`

Exemplo de Estrutura CCR-T:

```
CCR (Código do Comando: 272, Solicitação)
├── Session-Id: "pgw_c.example.com;123;456;gy"
├── Auth-Application-Id: 4
├── Origin-Host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org"
├── Origin-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── Destination-Realm: "epc.mnc999.mcc999.3gppnetwork.org"
├── CC-Request-Type: TERMINATION_REQUEST (3)
├── CC-Request-Number: 5
├── Termination-Cause: DIAMETER_LOGOUT (1)
└── Multiple-Services-Credit-Control (Grouped)
    ├── Service-Identifier: 1
    ├── Rating-Group: 100
    └── Used-Service-Unit (Grouped)
        └── CC-Total-Octets: 18500000 (18,5 MB de uso total)
```

CCA-Termination (Resposta de Controle de Crédito - Terminação)

Enviado por: OCS em resposta ao CCR-T

Propósito:

- Reconhecer a terminação da sessão
- Completar a contabilidade
- Liberar crédito reservado

Exemplo de CCA-T:

```
CCA (Código do Comando: 272, Resposta)
├── Session-Id: "pgw_c.example.com;123;456;gy"
├── Result-Code: DIAMETER_SUCCESS (2001)
├── Origin-Host: "ocs.example.com"
├── Origin-Realm: "example.com"
├── Auth-Application-Id: 4
├── CC-Request-Type: TERMINATION_REQUEST (3)
└── CC-Request-Number: 5
```

Fluxos de Cobrança Online

Tipos de Unidades de Serviço

O OCS pode conceder cota em diferentes unidades:

Tipo de Unidade	AVP	Descrição	Caso de Uso
Tempo	CC-Time	Segundos	Chamadas de voz, duração da sessão
Volume	CC-Total-Octets	Bytes (total de upload+download)	Serviços de dados
Volume (separado)	CC-Input-Octets, CC-Output-Octets	Bytes (separados)	Cobrança assimétrica
Específico do Serviço	CC-Service-Specific-Units	Unidades personalizadas	SMS, MMS, chamadas de API
Eventos	-	Eventos contados	Serviços pay-per-use

Gerenciamento de Limite de Cota

Problema: Como o PGW-C sabe quando solicitar mais cota?

Solução: O OCS fornece um **Volume-Quota-Threshold** ou **Time-Quota-Threshold**. O PGW-C monitora o uso através de Relatórios de Sessão PFCP do PGW-U (veja Interface PFCP).

Exemplo de Fluxo:

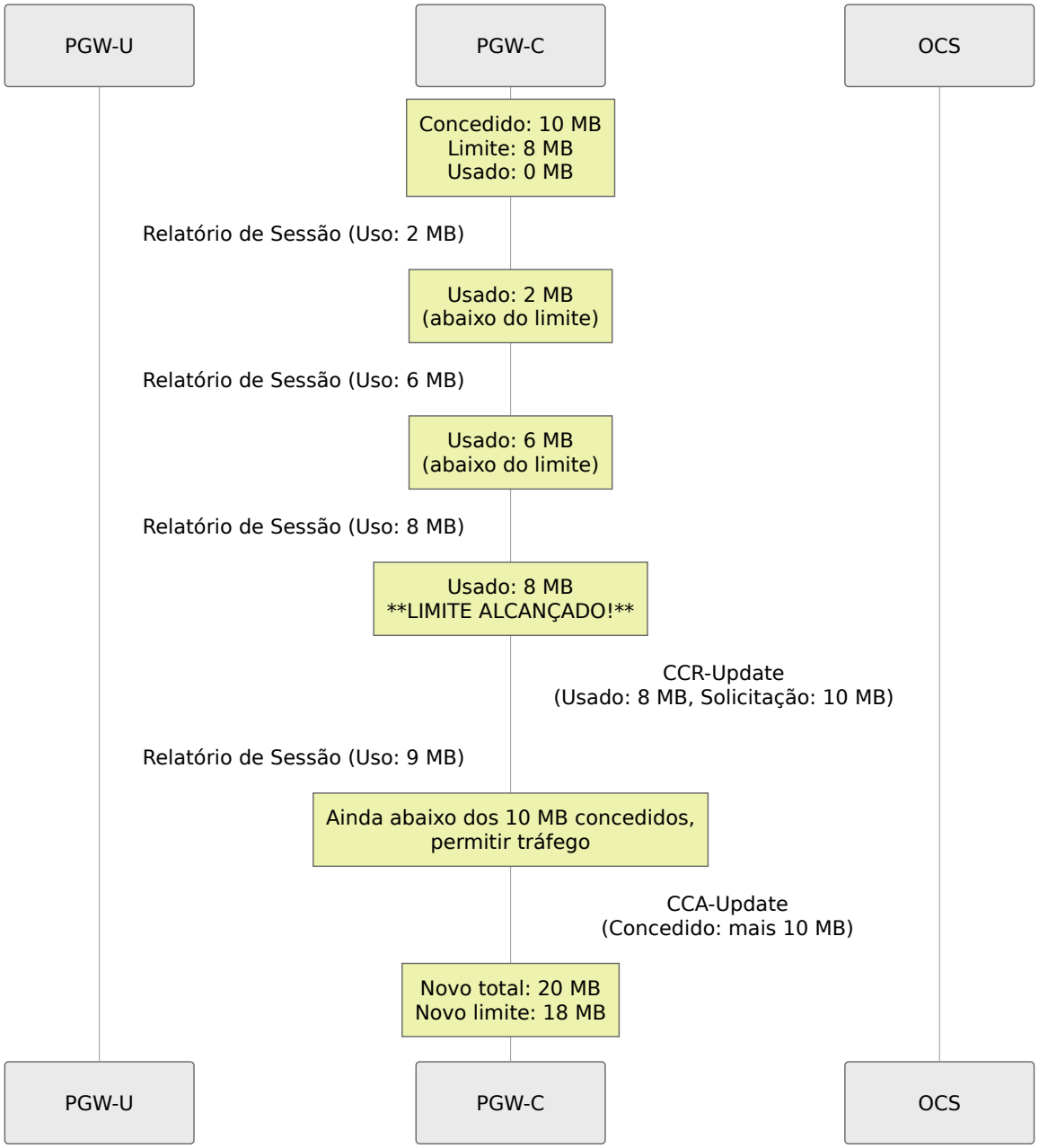
```
1. OCS concede 10 MB de cota com limite de 80% (8 MB)
2. PGW-C monitora o uso através de relatórios de uso do PGW-U
(Relatórios de Sessão PFCP)
3. Quando o uso atinge 8 MB:
   → PGW-C envia CCR-Update
   → Continuar permitindo tráfego (não esperar pela resposta)
4. OCS responde com mais cota
5. Se a cota se esgotar antes do CCR-Update enviado:
   → PGW-C deve bloquear o tráfego
```

Cálculo de Limite:

```
Granted-Service-Unit: 10000000 bytes (10 MB)
Volume-Quota-Threshold: 8000000 bytes (8 MB)

Quando 8 MB consumidos → Acionar CCR-Update
Buffer restante: 2 MB (permite tempo para a resposta do OCS)
```

Monitoramento do PGW-C:

O PGW-C monitora o uso através de **Relatórios de Sessão PFCP** do PGW-U:



Indicação da Unidade Final

O que acontece quando o crédito se esgota?

O OCS inclui o AVP **Final-Unit-Indication** na CCA para especificar a ação:

Ação da Unidade Final	Valor	Comportamento do PGW-C
TERMINAR	0	Bloquear todo o tráfego, iniciar a terminação da sessão
REDIRECIONAR	1	Redirecionar tráfego para o portal (por exemplo, página de recarga)
RESTRINGIR_ACESSO	2	Permitir acesso apenas a serviços específicos (por exemplo, servidor de recarga)

Exemplo: Unidade Final com Redirecionamento

```
CCA (Atualização)
└── Multiple-Services-Credit-Control
    ├── Result-Code: DIAMETER_SUCCESS (2001)
    ├── Granted-Service-Unit
    │   └── CC-Total-Octets: 1000000 (último 1 MB)
    └── Final-Unit-Indication
        ├── Final-Unit-Action: REDIRECT (1)
        └── Redirect-Server (Grouped)
            ├── Redirect-Address-Type: URL (2)
            └── Redirect-Server-Address:
"http://topup.example.com"
```

Ações do PGW-C:

1. **TERMINAR:** Enviar CCR-T, excluir bearer
2. **REDIRECIONAR:** Instalar regra PFCP para redirecionar HTTP para URL de recarga
3. **RESTRINGIR_ACESSO:** Instalar regras PFCP permitindo apenas IPs na lista branca

---

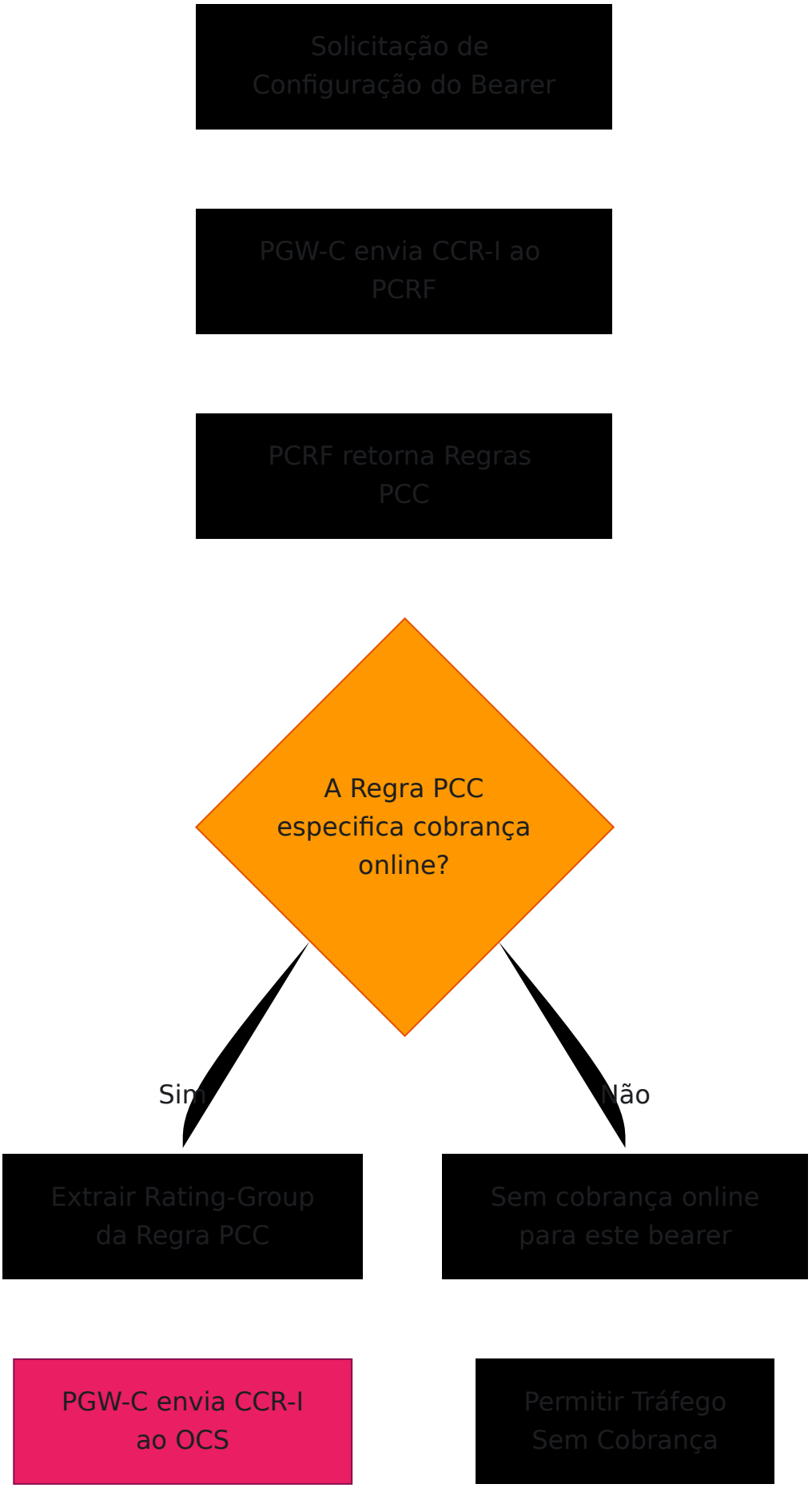
Controle de Cobrança de Bearer

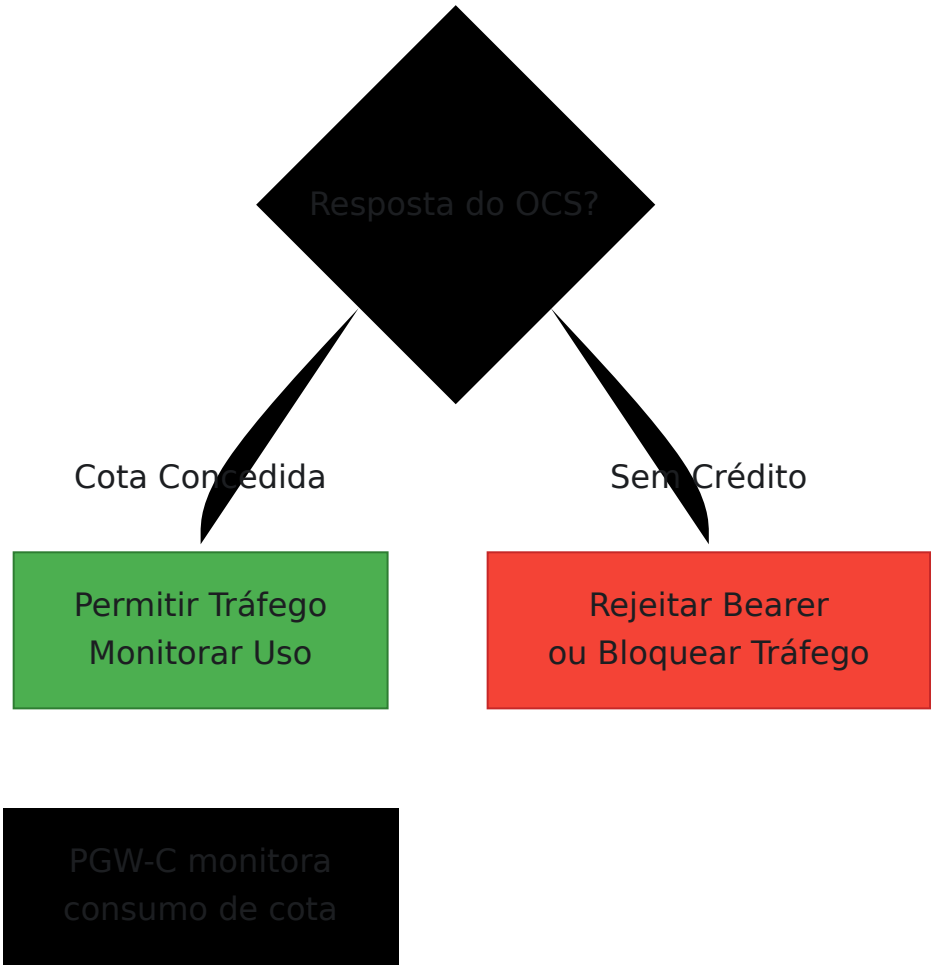
O que controla se um Bearer é cobrado?

Especificação 3GPP: TS 23.203, TS 29.212, TS 32.251

A cobrança de bearer é controlada por **Regras PCC** provisionadas pelo PCRF através da interface Gx. Veja Interface Diameter Gx para a documentação completa das regras PCC.

Fluxo de Decisão de Cobrança:


Solicitação de Configuração do Bearer
PGW-C envia CCR-I ao PCRF
PCRF retorna Regras PCC
A Regra PCC especifica cobrança online?
Sim
Não
Extrair Rating-Group da Regra PCC
Sem cobrança online para este bearer
PGW-C envia CCR-I ao OCS
Permitir Tráfego Sem Cobrança



Regra PCC com Informações de Cobrança

Resposta do PCRF (CCA-I na Gx):

```
CCA (Interface Gx)
└── Charging-Rule-Definition (Grouped)
    ├── Charging-Rule-Name: "prepaid_data_rule"
    ├── Rating-Group: 100
    ├── Online: 1 (habilitar cobrança online)
    ├── Offline: 0 (desabilitar cobrança offline)
    ├── Metering-Method: VOLUME (1)
    ├── Precedence: 100
    ├── Flow-Information: [...]
    └── QoS-Information: [...]
```

Principais AVPs de Cobrança nas Regras PCC:

Nome do AVP	Código do AVP	Valores	Descrição
Rating-Group	432	Unsigned32	Categoria de cobrança (mapeia para tarifa no OCS)
Online	1009	0=Desabilitar, 1=Habilitar	Habilitar cobrança online (Gy)
Offline	1008	0=Desabilitar, 1=Habilitar	Habilitar cobrança offline (Gz)
Metering-Method	1007	0=Duração, 1=Volume, 2=Ambos	O que medir
Reporting-Level	1011	0=Serviço, 1=Grupo de Classificação	Granularidade dos relatórios de uso

Matriz de Decisão de Cobrança de Bearer

Online	Offline	Rating-Group	Comportamento
1	0	Presente	Cobrança online apenas (pré-pago)
0	1	Presente	Cobrança offline apenas (pós-pago)
1	1	Presente	Cobrança online e offline (convergente)
0	0	-	Sem cobrança (serviço gratuito)

Múltiplos Grupos de Avaliação

Uma única conexão PDN pode ter **múltiplos portadores com diferentes grupos de avaliação**:

Exemplo de Cenário:

```
Portador Padrão (Internet)
├── Grupo de Avaliação: 100 (Dados Padrão)
└── Online: 1

Portador Dedicado 1 (Streaming de Vídeo)
├── Grupo de Avaliação: 200 (Serviço de Vídeo)
└── Online: 1

Portador Dedicado 2 (Voz IMS)
├── Grupo de Avaliação: 300 (Voz)
└── Online: 1
```

Comportamento do PGW-C Gy:

- **Único CCR-I** com múltiplas seções MSCC (Controle de Crédito de Múltiplos Serviços):

```
CCR-Inicial
├── Session-Id: "..."
└── Controle de Crédito de Múltiplos Serviços
    ├── [Grupo de Avaliação: 100] → Dados Padrão
    ├── [Grupo de Avaliação: 200] → Serviço de Vídeo
    └── [Grupo de Avaliação: 300] → Voz
```

Resposta do OCS:

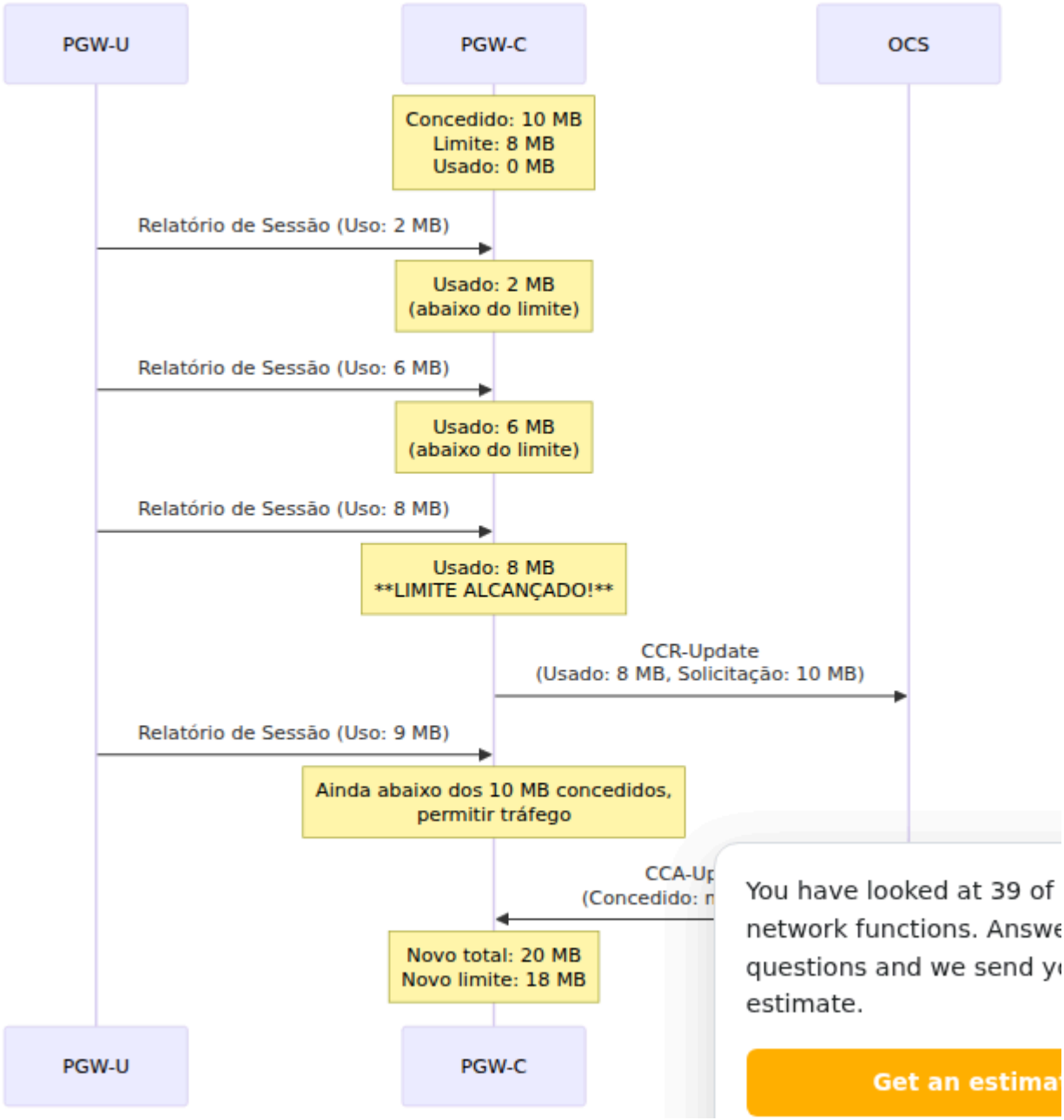
```
CCA-Inicial
└── Controle de Crédito de Múltiplos Serviços
    ├── [Grupo de Avaliação: 100] → Concedido: 10 MB
    ├── [Grupo de Avaliação: 200] → Concedido: 5 MB (vídeo mais
caro)
    └── [Grupo de Avaliação: 300] → Concedido: 60 segundos
```

Aplicação de Cobrança por Serviço

PGW-C rastreia a cota por Grupo de Avaliação:

```
# Pseudocódigo
state.charging_quotas = %{
  100 => %{granted: 10_000_000, used: 0, threshold: 8_000_000},
  200 => %{granted: 5_000_000, used: 0, threshold: 4_000_000},
  300 => %{granted: 60_000, used: 0, threshold: 48_000}  #
milissegundos
}
```

Monitoramento de Uso por Portador:



Controle de Crédito de Múltiplos Serviços

AVP MSCC (Controle de Crédito de Múltiplos Serviços)

Propósito: Agrupar informações de cobrança para um serviço/grupo de avaliação específico

Estrutura:

```
Controle de Crédito de Múltiplos Serviços (Agrupado, AVP 456)
├── Identificador de Serviço (Unsigned32, AVP 439)
├── Grupo de Avaliação (Unsigned32, AVP 432)
├── Unidade de Serviço Solicitada (Agrupada, AVP 437)
│   ├── CC-Tempo (Unsigned32, AVP 420)
│   ├── CC-Octetos-Total (Unsigned64, AVP 421)
│   ├── CC-Octetos-Entrada (Unsigned64, AVP 412)
│   └── CC-Octetos-Saída (Unsigned64, AVP 414)
├── Unidade de Serviço Usada (Agrupada, AVP 446)
│   └── [Mesma estrutura que Unidade de Serviço Solicitada]
├── Unidade de Serviço Concedida (Agrupada, AVP 431)
│   └── [Mesma estrutura que Unidade de Serviço Solicitada]
├── Tempo de Validade (Unsigned32, AVP 448)
├── Código de Resultado (Unsigned32, AVP 268)
└── Indicação de Unidade Final (Agrupada, AVP 430)
    └── Ação de Unidade Final (Enumerada, AVP 449)
```

Identificador de Serviço vs. Grupo de Avaliação

Atributo	Identificador de Serviço	Grupo de Avaliação
Propósito	Identifica o tipo de serviço	Identifica a categoria de cobrança
Exemplo	1=Dados, 2=Voz, 3=SMS	100=Regular, 200=Premium
Granularidade	Classificação ampla	Tarifa específica
Requerido	Opcional	Requerido para cobrança
Mapeamento	Pode mapear para múltiplos RGs	Tarifa única no OCS

Exemplo:

```
Identificador de Serviço: 1 (Serviço de Dados)
├── Grupo de Avaliação: 100 (Dados Padrão - $0.01/MB)
└── Grupo de Avaliação: 200 (Dados Premium - $0.05/MB)

Identificador de Serviço: 2 (Voz)
└── Grupo de Avaliação: 300 (Chamadas de Voz - $0.10/min)
```

Configuração

Configuração Básica do Gy

Edite `config/runtime.exs`:

```
config :pgw_c,
  gy: %{
    # Habilitar ou desabilitar cobrança online globalmente
    enabled: true,

    # Tempo limite de conexão OCS (milissegundos)
    timeout_ms: 5000,

    # Solicitação de cota padrão (bytes) se não especificado pelo
PCRF
    default_requested_quota: 10_000_000,  # 10 MB

    # Porcentagem de limite para re-autorização
    # (0.8 = acionar CCR-Atualização a 80% da cota consumida)
    quota_threshold_percentage: 0.8,

    # Ação quando ocorre timeout no OCS
    # Opções: :block, :allow
    timeout_action: :block,

    # Ação quando OCS retorna sem crédito
    # Opções: :terminate, :redirect
    no_credit_action: :terminate,

    # URL de redirecionamento para recarga (usada se
no_credit_action: :redirect)
    topup_redirect_url: "http://topup.example.com"
  },
  diameter: %{
    listen_ip: "0.0.0.0",
    host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org",
    realm: "epc.mnc999.mcc999.3gppnetwork.org",

    # Configuração do par OCS
    peer_list: [
      # PCRF para controle de política (Gx)
      %{
        host: "pcrf.epc.mnc999.mcc999.3gppnetwork.org",
        realm: "epc.mnc999.mcc999.3gppnetwork.org",
        ip: "10.0.0.30",
        initiate_connection: true
      },
      # OCS para cobrança online (Gy)
```

```
      %{
        host: "ocs.epc.mnc999.mcc999.3gppnetwork.org",
        realm: "epc.mnc999.mcc999.3gppnetwork.org",
        ip: "10.0.0.40",
        initiate_connection: true
      }
    ]
  }
```

Parâmetros de Configuração Explicados

`enabled`

- `true`: Cobrança online ativa, mensagens CCR enviadas para OCS
- `false`: Cobrança online desativada, sem mensagens Gy

`timeout_ms`

- Tempo para aguardar a resposta CCA do OCS
- Recomendado: 3000-5000 ms

`default_requested_quota`

- Cota padrão a ser solicitada se o PCRF não especificar
- Valores típicos: 1-100 MB

`quota_threshold_percentage`

- Acionar CCR-Atualização quando essa % da cota for consumida
- Recomendado: 0.75-0.85 (75%-85%)
- Maior = menos mensagens, mas risco de exaustão da cota
- Menor = mais mensagens, mas mais seguro

`timeout_action`

- **`:block`** - Bloquear tráfego se OCS não responder (mais seguro, previne perda de receita)
- **`:allow`** - Permitir tráfego se OCS não responder (melhor UX, risco de receita)

`no_credit_action`

- `:terminate` - Deletar portador quando crédito esgotado
- `:redirect` - Redirecionar para portal de recarga

Configuração Específica do Ambiente

Produção (assinantes pré-pagos):

```
config :pgw_c,
  gy: %{
    enabled: true,
    timeout_action: :block,
    no_credit_action: :terminate,
    quota_threshold_percentage: 0.8
  }
```

Teste/Desenvolvimento:

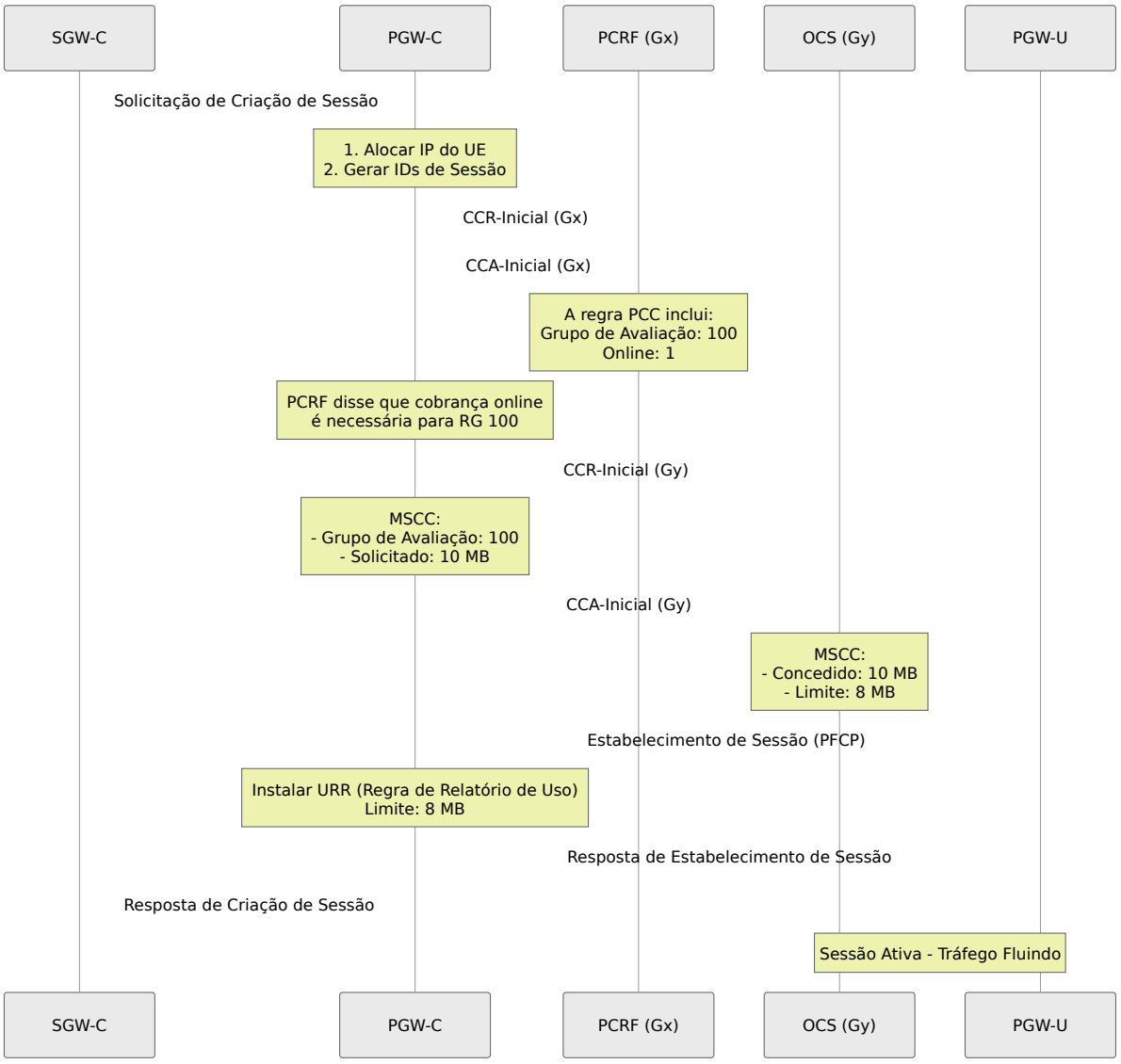
```
config :pgw_c,
  gy: %{
    enabled: false  # Desativar para testes
  }
```

Híbrido (alguns pré-pagos, alguns pós-pagos):

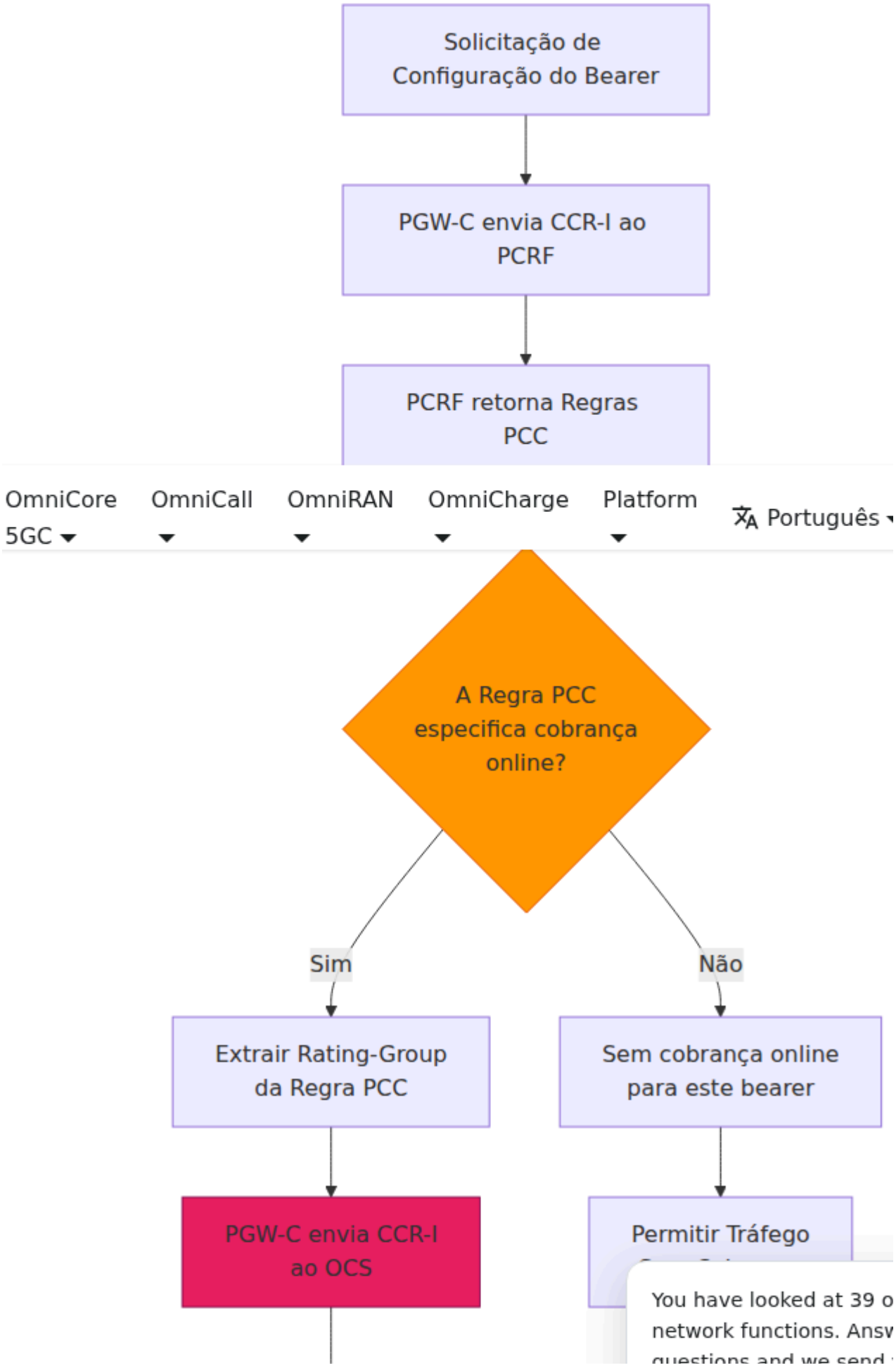
```
config :pgw_c,
  gy: %{
    enabled: true,  # Controlado por assinante pelo PCRF
    timeout_action: :allow,  # Não bloquear pós-pagos em falha do
OCS
    no_credit_action: :terminate
  }
```

Fluxos de Mensagens

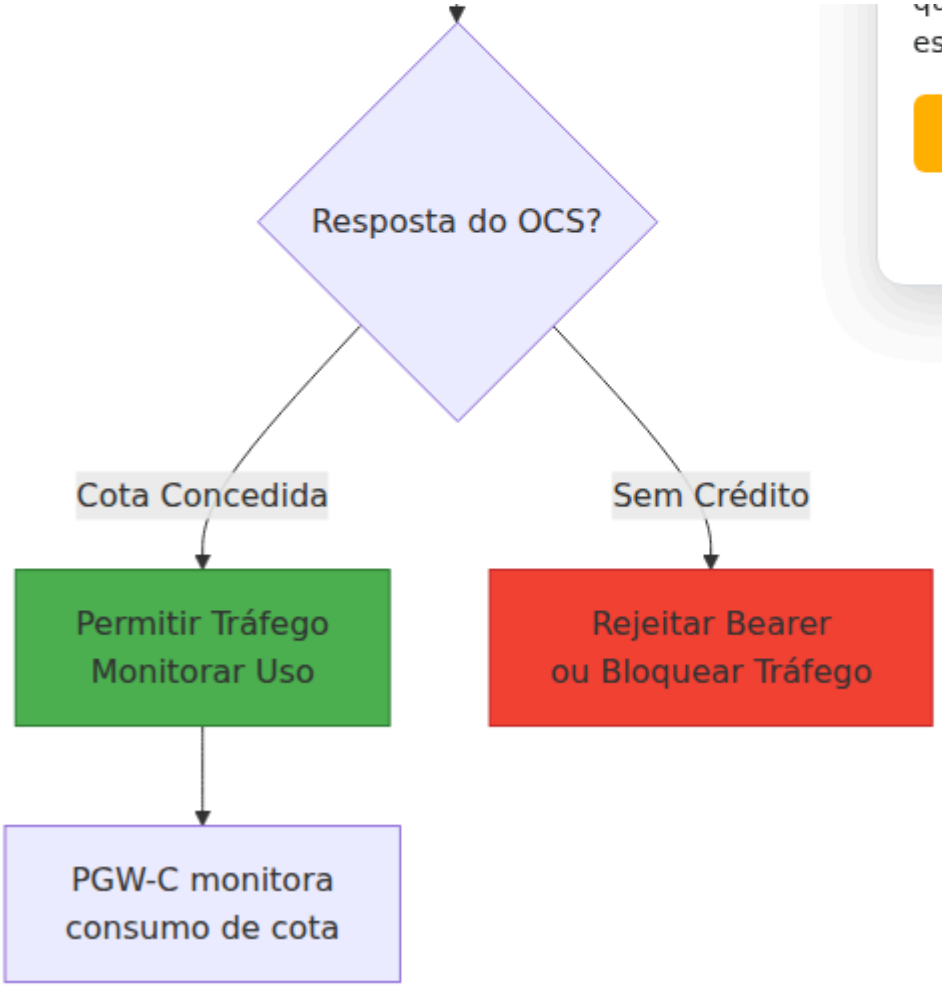
Sessão Bem-Sucedida com Cobrança Online



Re-autorização de Cota (CCR-Atualização)

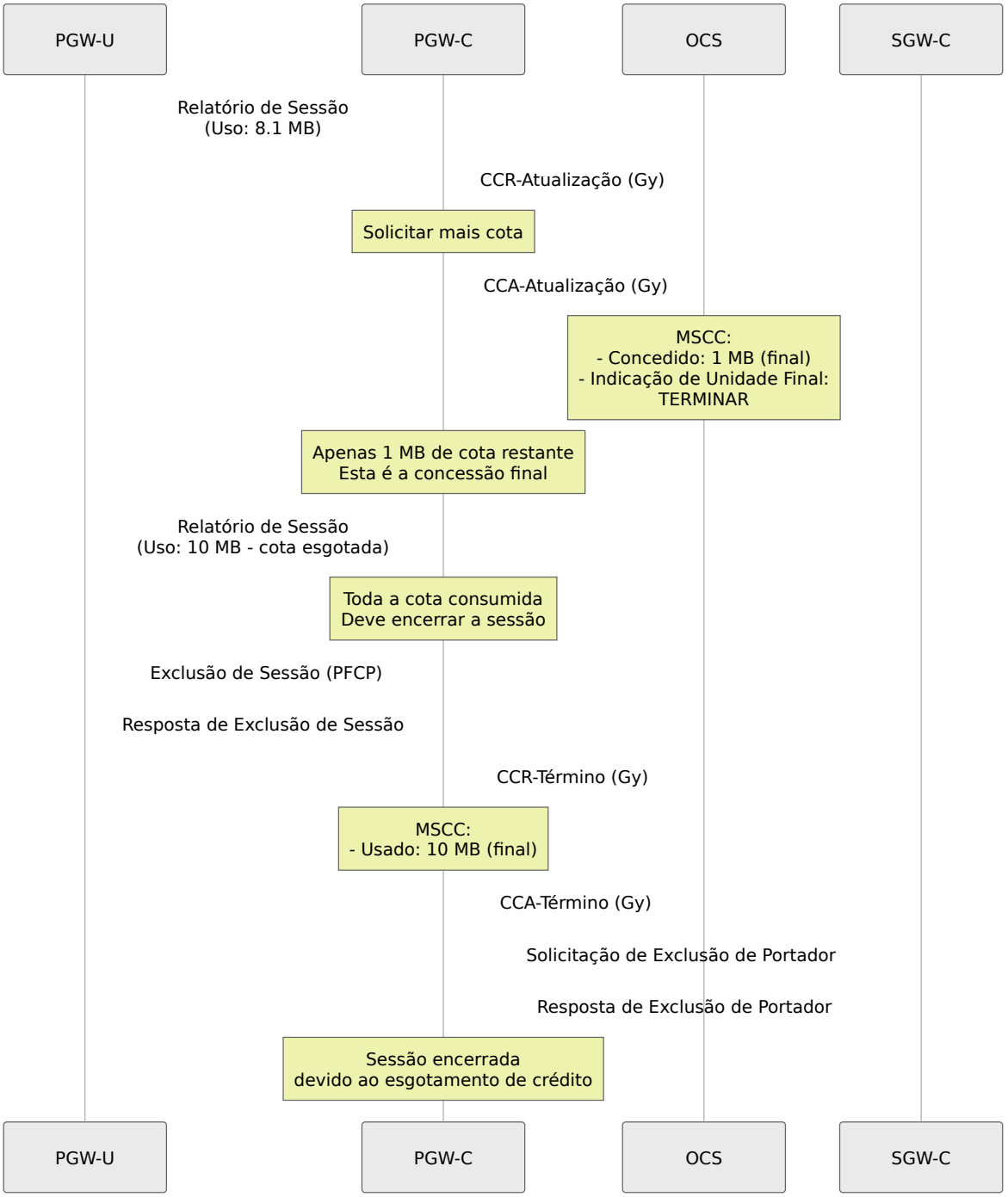


OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
Português
You have looked at 39 o
network functions. Ansv
questions and we send

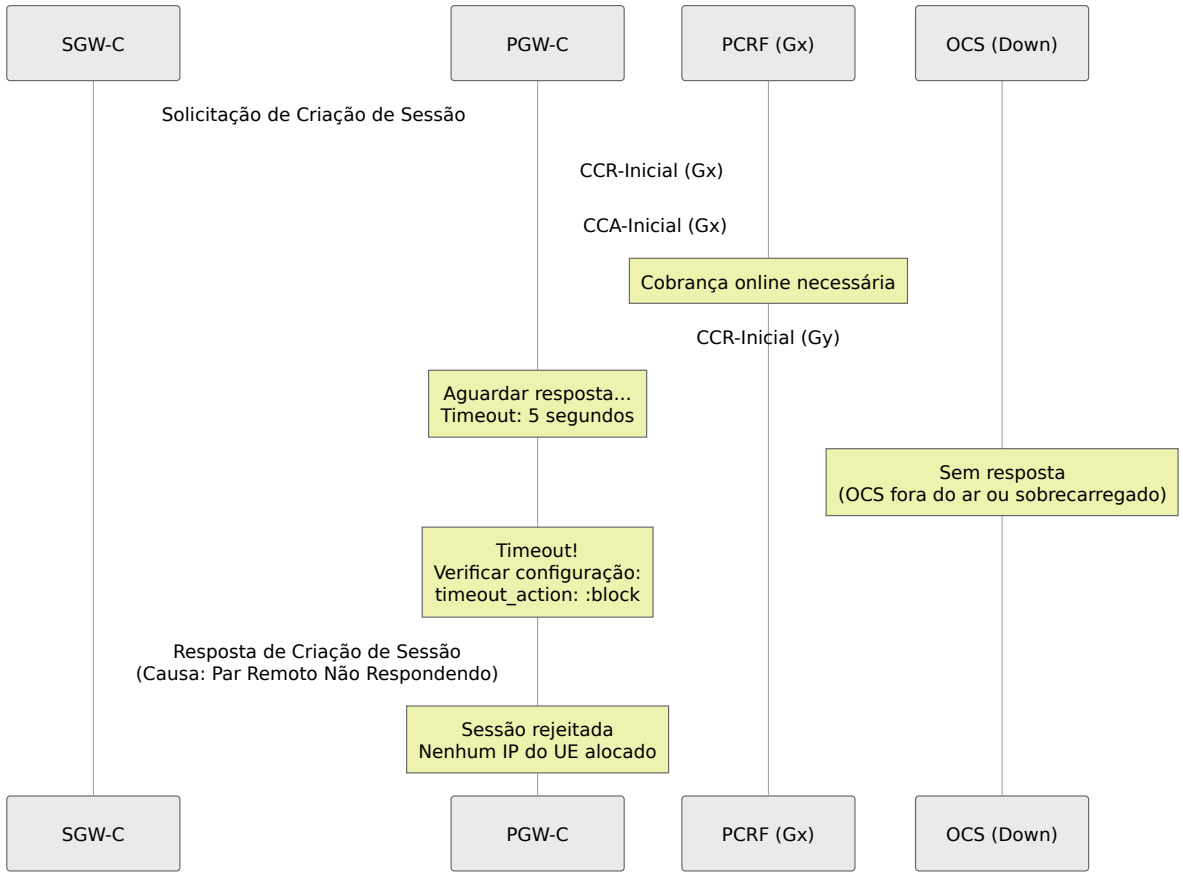


estimate.
Get an estim
No thanks

Esgotamento de Crédito (Unidade Final)


PGW-U
PGW-C
OCS
SGW-C
Relatório de Sessão
(Uso: 8.1 MB)
CCR-Atualização (Gy)
Solicitar mais cota
CCA-Atualização (Gy)
MSCC:
- Concedido: 1 MB (final)
- Indicação de Unidade Final:
TERMINAR
Apenas 1 MB de cota restante
Esta é a concessão final
Relatório de Sessão
(Uso: 10 MB - cota esgotada)
Toda a cota consumida
Deve encerrar a sessão
Exclusão de Sessão (PFCP)
Resposta de Exclusão de Sessão
CCR-Término (Gy)
MSCC:
- Usado: 10 MB (final)
CCA-Término (Gy)
Solicitação de Exclusão de Portador
Resposta de Exclusão de Portador
Sessão encerrada
devido ao esgotamento de crédito
PGW-U
PGW-C
OCS
SGW-C

Tratamento de Timeout do OCS



Tratamento de Erros

Códigos de Resultado

Códigos de Sucesso:

Código	Nome	Ação
2001	DIAMETER_SUCCESS	Continuar com cota concedida

Falhas Transitórias (4xxx):

Código	Nome	Ação do PGW-C
4010	DIAMETER_TOO_BUSY	Tentar novamente com backoff
4011	DIAMETER_UNABLE_TO_COMPLY	Registrar erro, pode tentar novamente
4012	DIAMETER_CREDIT_LIMIT_REACHED	Encerrar ou redirecionar

Falhas Permanentes (5xxx):

Código	Nome	Ação do PGW-C
5003	DIAMETER_AUTHORIZATION_REJECTED	Rejeitar sessão
5031	DIAMETER_USER_UNKNOWN	Rejeitar sessão (assinante inválido)

Códigos de Resultado por Serviço

Importante: O Código de Resultado pode aparecer em **dois níveis**:

1. **Nível de Mensagem** - Resultado geral
2. **Nível MSCC** - Resultado por serviço

Exemplo:

```
CCA-Inicial
├── Código de Resultado: DIAMETER_SUCCESS (2001)  ← Nível de
Mensagem: OK
└── Controle de Crédito de Múltiplos Serviços
    ├── [Grupo de Avaliação: 100]
    │   └── Código de Resultado: DIAMETER_SUCCESS (2001)  ← RG
100: OK
    └── [Grupo de Avaliação: 200]
        └── Código de Resultado: DIAMETER_CREDIT_LIMIT_REACHED
(4012)  ← RG 200: Sem crédito
```

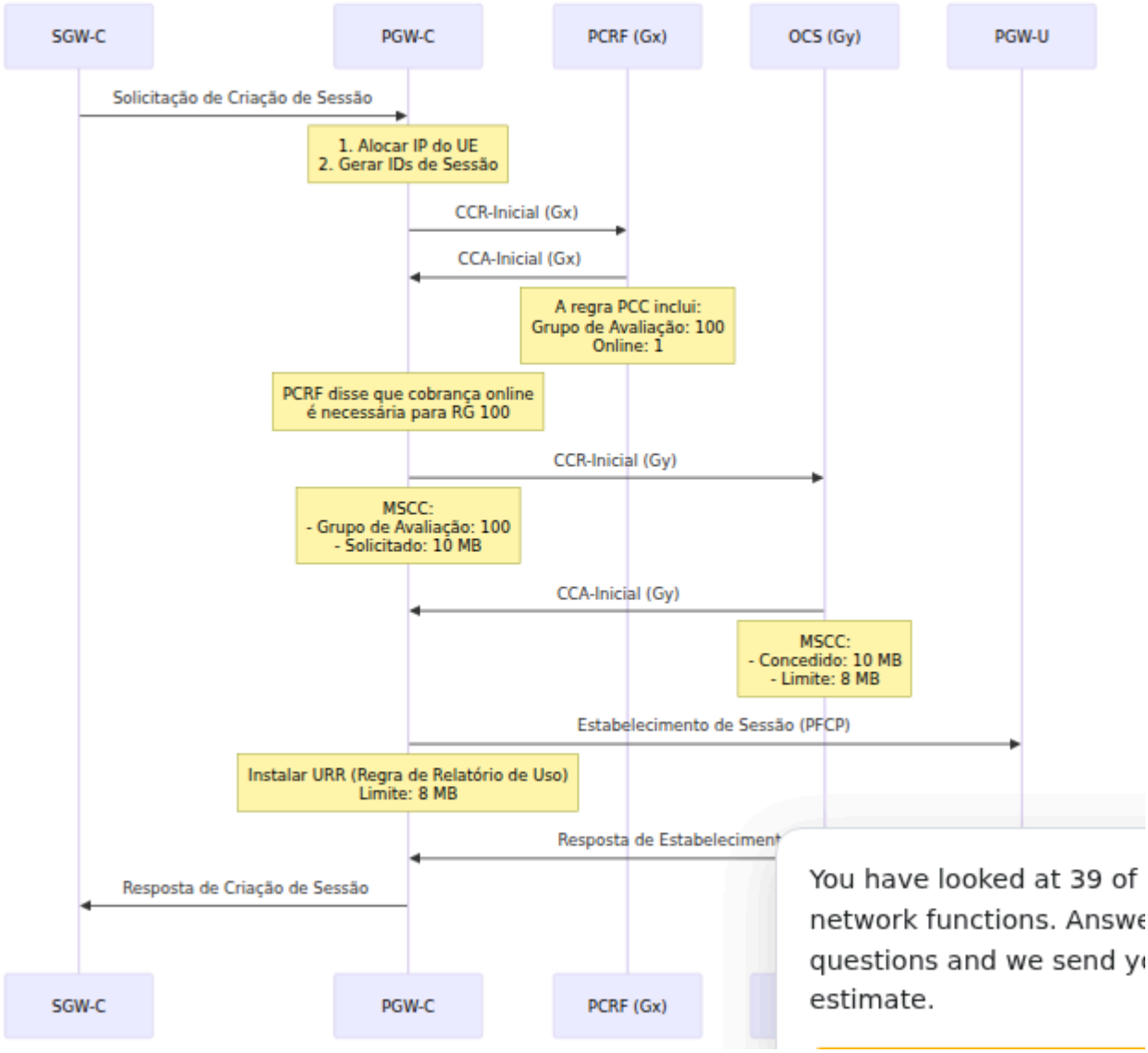
Comportamento do PGW-C:

- Permitir tráfego para o Grupo de Avaliação 100
- Bloquear tráfego para o Grupo de Avaliação 200

Integração com Gx

A interface Gx (controle de política PCRF) determina se a cobrança online é necessária e fornece o Grupo de Avaliação que direciona a cobrança Gy. Veja Interface Diameter Gx para a documentação completa de controle de política.

Relação entre Gx e Gy



You have looked at 39 of
network functions. Answe
questions and we send y
estimate.

Fluxo de Integração

1. Configuração do Portador:

```
PGW-C recebe Solicitação de Criação de Sessão
    ↓
Enviar CCR-I para PCRF (Gx)
    ↓
Receber CCA-I com Regras PCC
    ↓
Analisar Regras PCC:
    - A regra tem Grupo de Avaliação?
    - É Online = 1?
    ↓
Se SIM:
    Enviar CCR-I para OCS (Gy) com Grupo de Avaliação
    ↓
    Receber CCA-I com cota
    ↓
    Se cota concedida: Prosseguir
    Se sem crédito: Rejeitar portador
Se NÃO:
    Prosseguir sem cobrança online
```

2. Atualização de Política Dinâmica (RAR do PCRF):

```
PCRF envia RAR (Re-Auth-Request) no Gx
    ↓
Nova Regra PCC adicionada com Online=1, Grupo de Avaliação=200
    ↓
PGW-C envia CCR-U para OCS (Gy)
    - Adicionar MSCC para Grupo de Avaliação 200
    ↓
OCS concede cota para novo serviço
    ↓
Instalar portador dedicado com cobrança online
```

Solução de Problemas

Problemas Comuns

1. Timeouts de CCR-Inicial para OCS

Sintomas:

- Sessões falham com "timeout OCS"
- Log: "CCR-Inicial (Gy) timeout"

Possíveis Causas:

- OCS não acessível
- IP do OCS incorreto na configuração
- Firewall bloqueando a porta Diameter (3868)
- OCS sobrecarregado

Resolução:

```
# Testar conectividade de rede
ping <ocs_ip>

# Testar porta Diameter (TCP 3868)
telnet <ocs_ip> 3868

# Verificar configuração
# Garantir que o par OCS esteja configurado na peer_list
```

2. Sessões Rejeitadas pelo OCS

Sintomas:

- CCA-I com Código de Resultado != 2001
- Falha na Resposta de Criação de Sessão

Códigos de Resultado Comuns:

Código de Resultado	Causa Provável	Resolução
4012	Limite de crédito alcançado	Assinante precisa recarregar
5003	Autorização rejeitada	Verificar permissões do assinante
5031	Usuário desconhecido	Provisionar assinante no OCS

Passos de Depuração:

1. Verificar logs do OCS para razão da rejeição
2. Verificar saldo do assinante no OCS
3. Verificar se IMSI/MSISDN no CCR-I corresponde ao registro do assinante

3. Esgotamento de Cota Não Detectado

Sintomas:

- Usuário continua usando dados após saldo esgotado
- Nenhuma CCR-Atualização enviada

Possíveis Causas:

- URR (Regra de Relatório de Uso) não instalada no PGW-U
- Limite não configurado corretamente
- Relatórios de Sessão PFCP não recebidos

Passos de Depuração:

1. Verificar URR na Estabelecimento de Sessão PFCP:

```
Criar URR
├── URR-ID: 1
├── Método de Medição: VOLUME
├── Limite de Volume: 8000000 (8 MB)
└── Gatilhos de Relatório: VOLUME_THRESHOLD
```

2. Verificar logs do PGW-U para relatórios de uso
3. Verificar `quota_threshold_percentage` na configuração

4. Grupo de Avaliação Incorreto

Sintomas:

- OCS rejeita com "Grupo de Avaliação Desconhecido"
- Sessões falham

Causa:

- Grupo de Avaliação no CCR-I n��o corresponde à configuração do OCS
- PCRF provisionou Grupo de Avaliação inválido

Resolução:

1. Verificar Grupo de Avaliação na Regra PCC do PCRF
2. Verificar configuração do OCS para Grupos de Avaliação válidos
3. Garantir mapeamento entre Regras PCC e tarifas do OCS

Monitoramento

Métricas Chave

A lista completa, com rótulos, está no Guia de Monitoramento.

```
# Taxas de mensagens Gy (message_type é ro_CCR, ro_CCA, ro_RAR ou
ro_RAA)
rate(gy_outbound_messages_total{message_type="ro_CCR"}[5m])
rate(gy_inbound_messages_total{message_type="ro_CCA"}[5m])

# Resultados de CCA por tipo de solicitação (inicial, atualização,
término) e classe de código de resultado
sum by (cc_request_type, result_code_class)
(rate(gy_inbound_responses_total[5m]))

# Taxas de erro Gy
rate(gy_inbound_errors_total[5m])

# CCRs enviados sem CCA de volta (timeouts do OCS ou respostas
perdidas)
sum(rate(gy_outbound_messages_total{message_type="ro_CCR"}[5m]))
  - sum(rate(gy_inbound_responses_total[5m]))

# Tempo de ida e volta do OCS (CCR para CCA)
histogram_quantile(0.95, sum by (le)
(rate(gy_outbound_transaction_duration_bucket[5m])))

# Sessões Gy abertas
gy_session_id_registry_count
```

Alertas

```
# Alerta sobre uma alta taxa de erro Gy
- alert: GyErrorRateHigh
  expr: rate(gy_inbound_errors_total[5m]) > 0.1
  for: 5m
  annotations:
    summary: "Alta taxa de erro Gy detectada"

# Alerta quando o OCS não responde CCRs (timeouts ou respostas perdid
- alert: OcsNotAnswering
  expr: |
    sum by (instance) (rate(gy_outbound_messages_total{message_type="
      - sum by (instance) (rate(gy_inbound_responses_total[5m])) > 0.
  for: 2m
  annotations:
    summary: "CCRs para o OCS não estão sendo respondidos"

# Alerta quando o OCS rejeita novas sessões
- alert: OcsRejectingCcrI
  expr:
sum(rate(gy_inbound_responses_total{cc_request_type="initial",result_
[5m])) > 0.1
  for: 5m
  annotations:
    summary: "O OCS está rejeitando CCR-I (novas sessões Gy)"
```

Controle de Crédito Gy (API OAM)

A cobrança online é inspecionada e exercida através da **API REST OAM**, servida sobre HTTPS/TLS em `https://<host>:8443`. A documentação interativa da API (Swagger UI) está disponível em `https://<host>:8443/api/docs`.

Status da Cobrança Online

```
GET /api/charging                # todas as sessões de cobrança
online + estado de crédito ao vivo
GET /api/charging?search=<imsi>  # filtrar sessões por IMSI
GET /api/charging/<imsi>         # estado de controle de crédito
de uma sessão por IMSI
```

Exemplo:

```
curl -k https://localhost:8443/api/charging/310170123456789
```

`GET /api/charging` retorna o estado da cobrança online (cota concedida vs. usada, validade, a Ação de Unidade Final do OCS, e estado de redirecionamento de jardim murado / sem crédito) para cada sessão ativa. Adicione `?search=<imsi>` para restringir a lista, ou anexe o IMSI a `GET /api/charging/<imsi>` para buscar o estado de controle de crédito de um único assinante.

Simulador de Controle de Crédito Gy

```
POST /api/gy_simulator
```

Envia uma Solicitação de Controle de Crédito sintética para o OCS configurado e retorna a resposta analisada — Código de Resultado em nível de comando, cota concedida, tempo de validade e qualquer Indicação de Unidade Final — para que a cobrança online possa ser exercida **sem tráfego de assinante ao vivo**. Esta é a substituição da API OAM para o antigo simulador Gy baseado em navegador. Exige que Gy esteja habilitado (`gy.enabled: true`); a solicitação é enviada do próprio host de origem Gy do PGW-C para o domínio OCS configurado.

Corpo da solicitação (`application/json`):

Campo	Tipo	Descrição
`imsi`	string	**Requerido.** IMSI do assinante.
`msisdn`	string	MSISDN opcional (E.164). Preenche `User-Name` como `<msisdn>@<apn>`.
`apn`	string	APN opcional.
`requested_units`	integer	Cota solicitada opcional em octetos (Unidade de Serviço Solicitada → CC-Octetos-Total).
`rating_group`	integer	Grupo de Avaliação / chave de cobrança opcional (padrão `1`).
`service_id`	integer	Identificador de Serviço opcional.
`cc_request_type`	string	`initial` (padrão), `update`, ou `terminate` (ou o numérico `1`/`2`/`3`).

Exemplo — solicitar 1 MB de cota inicial para um assinante:

```
curl -k -X POST https://localhost:8443/api/gy_simulator \
  -H 'content-type: application/json' \
  -d
'{"imsi":"310170123456789","msisdn":"14155551234","apn":"internet","r
```

Exemplo de resposta (CCA-Inicial concedendo 1 MB, válido por 1 hora):

```
{
  "session_id": "omni-pgw_c.epc...;734825199;app_gy_sim",
  "cc_request_type": 1,
  "cc_request_type_name": "initial",
  "result_code": "diameter_success",
  "accepted": true,
  "ocs_host": "ocs.epc.example.org",
  "ocs_realm": "epc.example.org",
  "granted": { "total_octets": 1000000, "input_octets": null,
"output_octets": null, "time": null },
  "validity_time": 3600,
  "final_unit_action": null,
  "final_unit_redirect": null,
  "mscc_result_code": "diameter_success"
}
```


Em caso de falha, o endpoint retorna `422` (`imsi` ausente) ou `503` (Gy desativado, ou o OCS não respondeu — o campo `reason` carrega o erro subjacente, por exemplo, `:timeout`).

Casos de Uso Operacionais

- **Inspecionar a cota de um assinante** — `GET /api/charging/<imsi>` retorna a cota concedida atual e o status de crédito da sessão.
- **Testar a integração do OCS sem tráfego** — `POST /api/gy_simulator` para confirmar que o OCS concede a cota esperada / Ação de Unidade Final para um determinado IMSI, grupo de avaliação ou serviço.
- **Resolver problemas de cobrança** — localizar um assinante afetado com `GET /api/charging?search=<imsi>` e revisar seu estado de controle de crédito ao vivo.

Para tendências históricas (taxas de mensagens, esgotamento de cota, timeouts do OCS), use as métricas do Prometheus descritas em Monitoring acima.

Documentação Relacionada

Cobrança e Política

- **Interface Diameter Gx** - Controle de política PCRF, regras PCC que acionam a cobrança online
- **Formato de CDR de Dados** - Registros de cobrança offline para faturamento pós-pago
- **Guia de Configuração** - Parâmetros completos de configuração de cobrança online

Gerenciamento de Sessão

- **Gerenciamento de Sessão** - Ciclo de vida da sessão PDN, gerenciamento de bearer
- **Interface PFCP** - Relatório de uso do PGW-U via URRs
- **Interface S5/S8** - Configuração e desmontagem do bearer GTP-C

Operações

- **Guia de Monitoramento** - Métricas Gy, rastreamento de cota, alertas de timeout do OCS
- **Alocação de IP do UE** - Configuração do pool de IP para sessões cobradas

---

Voltar ao Guia de Operações

Documentação da Interface Gn/Gp

Comunicação GTP-C com SGSN (Redes 2G/3G)

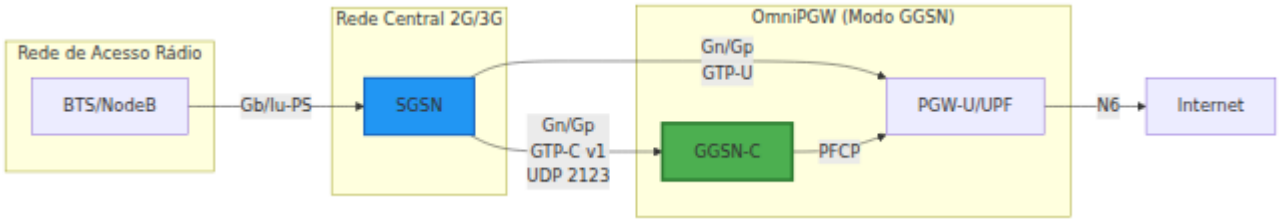
OmniPGW da Omnitouch Network Services

Visão Geral

A **interface Gn/Gp** permite que o OmniPGW funcione como um **GGSN** (Gateway GPRS Support Node) para redes 2G/3G usando o protocolo **GTP-C v1**. Esta interface gerencia o contexto PDP (Packet Data Protocol) entre o GGSN e os SGSNs.

- **Interface Gn**: Conecta-se aos SGSNs dentro da mesma PLMN (rede doméstica)
- **Interface Gp**: Conecta-se aos SGSNs em outras PLMNs (cenários de roaming)

Ambas as interfaces usam protocolos idênticos - a distinção é puramente topológica.



Principais Diferenças em Relação ao PGW (S5/S8)

Aspecto	GGSN (Gn/Gp)	PGW (S5/S8)
Versão do Protocolo	GTP-C v1	GTP-C v2
Especificação 3GPP	TS 29.060	TS 29.274
Conceito de Sessão	Contexto PDP	Sessão PDN + Bearers
Identificador de Bearer	NSAPI (0-15)	EBI (5-15)
Interface de Política	Nenhuma (política estática)	Gx (PCRF)
Modelo de QoS	Perfil de QoS R99	QoS de Bearer EPS (QCI)
Porta	2123 (compartilhada)	2123

Detalhes do Protocolo

GTP-C Versão 1

- **Protocolo**: GTP-C v1 (3GPP TS 29.060)
- **Transporte**: UDP
- **Porta**: 2123 (compartilhada com S5/S8 GTP-C v2)
- **Tipo de Interface**: Plano de Controle

TEID (Identificador de Ponto de Extremidade do Túnel)

Cada contexto PDP tem TEIDs únicos para controle e plano de usuário:

- **TEID de Controle do GGSN**: Alocado pelo OmniPGW para mensagens de controle recebidas
- **TEID de Dados do GGSN**: Alocado pelo UPF para dados de usuário recebidos
- **TEID de Controle do SGSN**: Fornecido pelo SGSN para mensagens de controle enviadas
- **TEID de Dados do SGSN**: Fornecido pelo SGSN para dados de usuário enviados

```
Fluxo de Mensagem:
  SGSN → GGSN: TEID de Destino = TEID de Controle do GGSN
  GGSN → SGSN: TEID de Destino = TEID de Controle do SGSN
```

Suporte a CUPS (Separação de Controle e Plano de Usuário)

O OmniPGW suporta a arquitetura CUPS para 2G/3G:

- **Endereço GSN para Sinalização**: IP do plano de controle do OmniPGW (de `gn.local_ipv4_address`)
- **Endereço GSN para Tráfego de Usuário**: Endereço IP do UPF (da Resposta de Estabelecimento de Sessão PFCP)

Isso permite que o GGSN-C e o PGW-U/UPF operem em nós separados com endereços IP diferentes, o que é padrão para implantações CUPS.

NSAPI (Identificador de Ponto de Acesso ao Serviço de Rede)

O NSAPI identifica contextos PDP individuais dentro de uma sessão de assinante:

- **Faixa**: 5-15 (0-4 reservados)
- **Uso**: Múltiplos contextos PDP por assinante são possíveis
- **Equivalente**: Semelhante ao EBI em LTE/EPC

Configuração

Configuração Básica

```
# config/runtime.exs
config :pgw_c,
  # A interface Gn/Gp é habilitada automaticamente quando a
configuração gn está presente
  gn: %{
    # Endereço IPv4 local para a interface Gn
    local_ipv4_address: "10.0.0.20",

    # Opcional: Endereço IPv6 local
    local_ipv6_address: nil,

    # Porta para GTP-C v1 (compartilhada com S5/S8)
    local_port: 2123
  },

  # Servidores DNS retornados em PCO (compartilhados com PGW)
  dns: %{
    primary_ipv4: {8, 8, 8, 8},
    secondary_ipv4: {8, 8, 4, 4}
  }
```

Parâmetros da Interface Gn

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
`local_ipv4_address`	String	Sim	-	Endereço IPv4 para vincular à interface Gn. Este é o endereço do GGSN anunciado aos SGSNs.
`local_ipv6_address`	String	Não	`nil`	Endereço IPv6 opcional. A maioria das redes 2G/3G é apenas IPv4.
`local_port`	Inteiro	Não	2123	Porta UDP para GTP-C. Porta padrão conforme 3GPP TS 29.060. Compartilhada com S5/S8.

Parâmetros DNS

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
`primary_ipv4`	Tupla	Não	`{8, 8, 8, 8}`	Endereço do servidor DNS primário IPv4 retornado em PCO.
`secondary_ipv4`	Tupla	Não	`{8, 8, 4, 4}`	Endereço do servidor DNS secundário IPv4 retornado em PCO.
`primary_ipv6`	Tupla	Não	`nil`	Endereço do servidor DNS primário IPv6.
`secondary_ipv6`	Tupla	Não	`nil`	Endereço do servidor DNS secundário IPv6.

Requisitos de Rede

Regras de Firewall:

```
# Permitir GTP-C da rede SGSN
iptables -A INPUT -p udp --dport 2123 -s <sgsn_network>/24 -j
ACCEPT

# Permitir GTP-C de saída para SGSN
iptables -A OUTPUT -p udp --dport 2123 -d <sgsn_network>/24 -j
ACCEPT
```

Tipos de Mensagens

Gerenciamento de Contexto PDP

Solicitação de Criação de Contexto PDP

Direção: SGSN → GGSN

Propósito: Estabelecer um novo contexto PDP para conectividade de dados

Principais IEs (Elementos de Informação):

Nome IE	Tipo	Descrição
IMSI	Identidade	Identidade Internacional do Assinante Móvel
NSAPI	Inteiro	Identificador de Ponto de Acesso ao Serviço de Rede (5-15)
TEID Dados I	Inteiro	Ponto de extremidade do túnel do plano de usuário do SGSN
TEID Controle	Inteiro	Ponto de extremidade do túnel do plano de controle do SGSN
Endereço do Usuário Final	Endereço	Tipo de PDN solicitado e IP opcional
Nome do Ponto de Acesso	String	APN para a conexão
Endereço GSN (Sinalização)	IP	Endereço do plano de sinalização do SGSN
Endereço GSN (Usuário)	IP	Endereço do plano de usuário do SGSN
Perfil de QoS	Binário	Parâmetros de QoS solicitados
MSISDN	Identidade	Número de telefone móvel (opcional)
Modo de Seleção	Enum	Modo de seleção de APN
Opções de Configuração de Protocolo	Opções	Opções de protocolo adicionais

Exemplo:

```
Solicitação de Criação de Contexto PDP
├── IMSI: 310260123456789
├── NSAPI: 5
├── TEID Dados I: 0x12345678
├── TEID Controle: 0x87654321
├── Endereço do Usuário Final: IPv4 (dinâmico)
├── APN: internet
├── Endereço GSN (Sinalização): 10.1.1.100
├── Endereço GSN (Usuário): 10.1.1.100
├── Perfil de QoS: [Parâmetros de QoS R99]
└── PCO: [Solicitação de DNS, etc.]
```

Resposta de Criação de Contexto PDP

Direção: GGSN → SGSN

Propósito: Reconhecer a criação do contexto PDP com recursos alocados

Principais IEs:

Nome IE	Tipo	Descrição
Causa	Resultado	Código de sucesso ou erro
Reordenação Necessária	Booleano	Se a reordenação é necessária
Recuperação	Contador	Contador de reinício do GGSN
TEID Dados I	Inteiro	Ponto de extremidade do túnel do plano de usuário do GGSN
TEID Controle	Inteiro	Ponto de extremidade do túnel do plano de controle do GGSN
NSAPI	Inteiro	NSAPI confirmado
ID de Cobrança	Binário	ID de correlação de cobrança
Endereço do Usuário Final	Endereço	Endereço IP do UE alocado
Endereço GSN (Sinalização)	IP	Endereço do plano de sinalização do GGSN
Endereço GSN (Usuário)	IP	Endereço do plano de usuário do GGSN
Perfil de QoS	Binário	Parâmetros de QoS negociados
Opções de Configuração de Protocolo	Opções	Servidores DNS, etc.

Resposta de Sucesso:

```
Resposta de Criação de Contexto PDP
├── Causa: Solicitação Aceita (128)
├── TEID Dados I: 0xAABBCCDD
├── TEID Controle: 0xDDCCBBAA
├── NSAPI: 5
├── ID de Cobrança: 0x11223344
├── Endereço do Usuário Final
│   └── IPv4: 100.64.1.42
├── Endereço GSN (Sinalização): 10.0.0.20
├── Endereço GSN (Usuário): 10.0.0.20
├── Perfil de QoS: [QoS Negociado]
└── PCO
    ├── DNS Primário: 8.8.8.8
    └── DNS Secundário: 8.8.4.4
```

Solicitação de Atualização de Contexto PDP

Direção: SGSN → GGSN

Propósito: Modificar um contexto PDP existente (por exemplo, realocação de SGSN, alteração de QoS)

Cenários comuns:

- Transferência inter-SGSN (atualização da área de roteamento)
- Renegociação de QoS
- Alteração de endereço do SGSN

Principais IEs:

Nome IE	Descrição
NSAPI	Identificador do contexto PDP
TEID Dados I	Novo TEID do plano de usuário do SGSN
Endereço GSN	Novos endereços do SGSN
Perfil de QoS	QoS atualizada (opcional)

Resposta de Atualização de Contexto PDP

Direção: GGSN → SGSN

Propósito: Reconhecer a atualização do contexto PDP

Principais IEs:

Nome IE	Descrição
Causa	Código de sucesso ou erro
TEID Dados I	TEID do plano de usuário do GGSN
Endereço GSN	Endereços do GGSN
Perfil de QoS	QoS negociada

Solicitação de Exclusão de Contexto PDP

Direção: SGSN → GGSN

Propósito: Terminar um contexto PDP

Principais IEs:

Nome IE	Descrição
NSAPI	Contexto PDP a ser excluído
Indicação de Desmontagem	Excluir todos os contextos para este assinante

Resposta de Exclusão de Contexto PDP

Direção: GGSN → SGSN

Propósito: Reconhecer a exclusão do contexto PDP

Principais IEs:

Nome IE	Descrição
Causa	Código de sucesso ou erro

Gerenciamento de Caminho

Solicitação / Resposta de Eco

Direção: Bidirecional

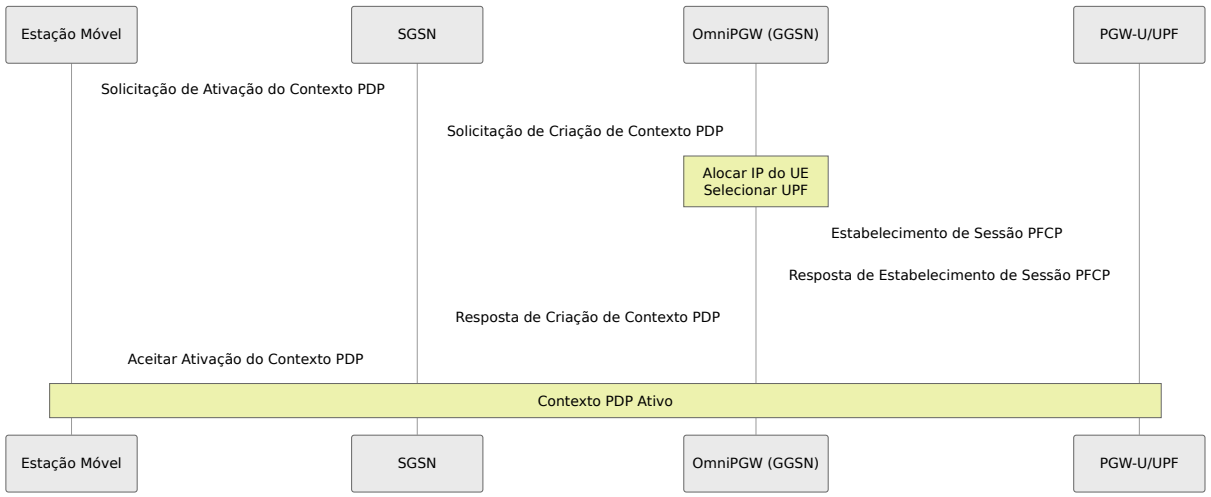
Propósito: Manter o caminho e detectar reinicializações

Principais IEs:

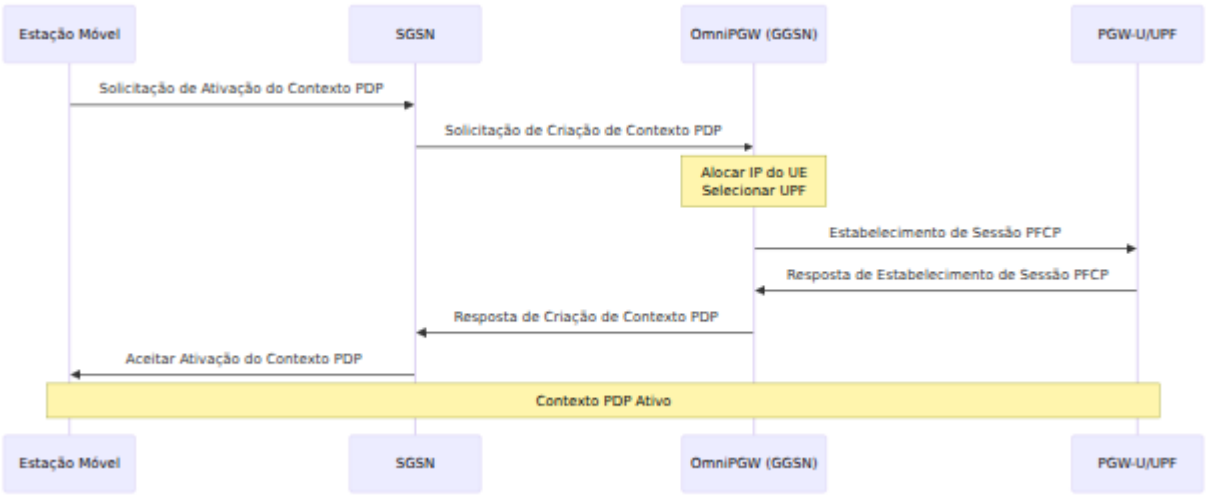
Nome IE	Descrição
Recuperação	Contador de reinício para detectar reinicializações de nó

Fluxos de Mensagem

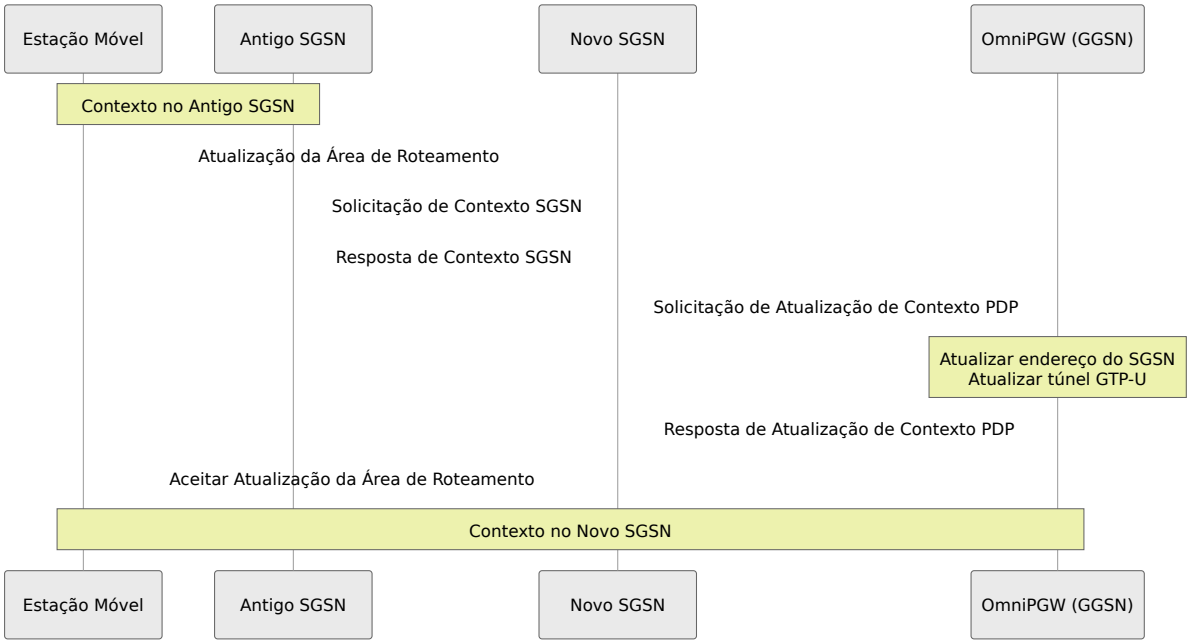
Ativação do Contexto PDP



Desativação do Contexto PDP



Transferência Inter-SGSN



Infraestrutura Compartilhada

A funcionalidade GGSN do OmniPGW compartilha infraestrutura com o modo PGW:

Alocação de Endereço IP do UE

Usa o mesmo AddressRegistry e configuração de sub-rede que o PGW. Veja Alocação de IP do UE para detalhes de configuração.

Gerenciamento de Sessão UPF/PFCP

Usa o mesmo estabelecimento de sessão PFCP que o PGW:

- Regras PDR/FAR/QER criadas por contexto PDP
- Seleção de UPF via a mesma configuração upf_selection
- Veja Interface PFCP para detalhes

Cobrança Online (Gy)

Integração opcional Diameter Gy disponível:

- Controle de crédito para assinantes pré-pagos
- Mesma configuração Gy que o modo PGW
- Veja Interface Diameter Gy para configuração

Geração de CDR

Usa a mesma geração de CDR que o PGW. Veja Formato de CDR de Dados para detalhes.

Códigos de Causa

Sucesso

Código	Nome	Descrição
128	Solicitação Aceita	Operação bem-sucedida

Causas de Rejeição

Código	Nome	Quando Usado
192	Não existente	Contexto PDP não encontrado
193	Formato de mensagem inválido	Mensagem GTP malformada
194	IMSI não conhecido	Assinante não encontrado
195	MS está Desconectado do GPRS	MS não anexado
196	MS não está Respondendo ao GPRS	MS inacessível
197	MS Recusa	MS rejeitou a solicitação
198	Versão não suportada	Incompatibilidade de versão GTP
199	Nenhum recurso disponível	Exaustão de recursos
200	Serviço não suportado	Serviço não suportado
201	IE obrigatório incorreto	Validação de IE falhou
202	IE obrigatório ausente	IE requerido ausente
203	IE opcional incorreto	IE opcional inválido
204	Falha no sistema	Erro interno
205	Restrição de roaming	Roaming não permitido
206	Desvio de Assinatura P-TMSI	Falha de segurança
207	Conexão GPRS suspensa	Conexão suspensa

Código	Nome	Quando Usado
208	Falha de autenticação	Autenticação falhou
209	Falha na autenticação do usuário	Autenticação do usuário falhou

---

Monitoramento

Métricas da Interface Gn

```
# Contadores de mensagens
gn_inbound_messages_total{message_type="create_pdp_context_request"}
gn_inbound_messages_total{message_type="update_pdp_context_request"}
gn_inbound_messages_total{message_type="delete_pdp_context_request"}

gn_outbound_messages_total{message_type="create_pdp_context_response"
gn_outbound_messages_total{message_type="update_pdp_context_response"
gn_outbound_messages_total{message_type="delete_pdp_context_response"

# Latência da transação
gn_outbound_transaction_duration_bucket
```

Consultas Úteis

Taxa de Criação de Contexto PDP:

```
rate(gn_inbound_messages_total{message_type="create_pdp_context_reque
[5m])
```

Taxa de Sucesso:

```
sum(rate(gn_outbound_messages_total{message_type="create_pdp_context_
[5m]))
  /
sum(rate(gn_inbound_messages_total{message_type="create_pdp_context_r
[5m]))
```

Contextos PDP Ativos:

```
gn_inbound_messages_total{message_type="create_pdp_context_request"}
  -
gn_inbound_messages_total{message_type="delete_pdp_context_request"}
```

Resolução de Problemas

Problema: Sem Resposta à Solicitação de Criação de Contexto PDP

Sintomas:

- SGSN envia Solicitação de Criação de Contexto PDP
- Nenhuma resposta recebida
- Tempo limite no SGSN

Possíveis causas:

1. Problema de conectividade de rede
2. OmniPGW não está ouvindo no IP configurado
3. Firewall bloqueando UDP 2123
4. Falha no estabelecimento de UPF/PFCP

Depuração:

```
# Verificar se o OmniPGW está ouvindo
netstat -ulnp | grep 2123

# Verificar pacotes recebidos
tcpdump -i any -n port 2123

# Verificar configuração
grep "local_ipv4_address" config/runtime.exs

# Verificar conectividade PFCP
curl http://pgw:9090/metrics | grep pfcp
```

Problema: Criação de Contexto PDP Rejeitada

Sintomas:

- Resposta de Criação de Contexto PDP com código de erro
- Contexto PDP não estabelecido

Causas comuns:

```
Causa 199 (Nenhum recurso disponível):
  → Pool de IP esgotado
  → Verificar: curl http://pgw:9090/metrics | grep
address_registry_count
  → Expandir pool de IP em ue.subnet_map

Causa 202 (IE obrigatório ausente):
  → SGSN não enviando IEs requeridos
  → Verificar configuração do SGSN
  → Revisar mensagem no tcpdump

Causa 204 (Falha no sistema):
  → Falha no UPF/PFCP
  → Verificar status da associação PFCP
  → Revisar logs do OmniPGW
```

Problema: Falha na Atualização do Contexto PDP

Sintomas:

- Resposta de Atualização de Contexto PDP com código de erro
- Transferência falha

Possíveis causas:

- Contexto PDP não encontrado (TEID diferente)
- Incompatibilidade de NSAPI
- Sessão já excluída

Depuração:

```
# Verificar sessões ativas
curl http://pgw:9090/metrics | grep session_count

# Revisar logs para busca de contexto
journalctl -u omnipgw | grep "PDP Context"
```

Melhores Práticas

Design de Rede

1. **Porta Compartilhada com S5/S8**
    - Gn e S5/S8 compartilham a porta UDP 2123
    - OmniPGW detecta automaticamente a versão GTP a partir da mensagem
    - Um único IP pode atender tanto 2G/3G quanto 4G
2. **Conectividade do SGSN**
    - Garantir que todos os SGSNs possam alcançar o IP do GGSN

- Considerar VLAN dedicada para tráfego Gn
- Planejar para roaming (Gp) se aplicável

3. **Planejamento de Endereço IP**

   - Os mesmos pools de IP atendem tanto os modos PGW quanto GGSN
   - Considerar pools baseados em APN separados para 2G/3G

Desempenho

1. **Capacidade de Sessão**

   - Monitorar a métrica `session_count`
   - Planejar para o número esperado de assinantes 2G/3G
   - Considerar que 2G/3G normalmente tem menos sessões simultâneas

2. **Tamanhos de Buffer UDP**

   - As mesmas recomendações que S5/S8
   - Típico: 4-8 MB por socket

Considerações de Migração

Ao migrar de um GGSN autônomo para o OmniPGW:

1. **Configuração de APN**

   - Garantir que os APNs correspondam à configuração existente do GGSN
   - Verificar configurações do servidor DNS em PCO

2. **Migração de Pool de IP**

   - Planejar cuidadosamente a transição do pool de IP
   - Considerar migração gradual por APN

3. **Mapeamento de QoS**

   - Perfis de QoS R99 aceitos como estão
   - Nenhuma tradução de QCI necessária para 2G/3G

Documentação Relacionada

Funções Principais

- **Guia de Configuração** - Referência completa de configuração
- **Gerenciamento de Sessão** - Detalhes do ciclo de vida da sessão
- **Alocação de IP do UE** - Gerenciamento de endereços IP
- **Configuração de PCO** - Opções de Configuração de Protocolo

Interfaces Relacionadas

- **Interface S5/S8** - Interface equivalente 4G/LTE
- **Interface PFCP** - Coordenação do plano de usuário
- **Interface Diameter Gy** - Cobrança online

Operações

- **Guia de Monitoramento** - Métricas e alertas
- **Resolução de Problemas** - Problemas comuns
- **Formato de CDR de Dados** - Geração de CDR

Referências

- 3GPP TS 29.060 - Protocolo de Tunelamento GPRS (GTP) através da interface Gn e Gp
- 3GPP TS 23.060 - Serviço Geral de Pacote de Rádio (GPRS); Descrição do serviço
- 3GPP TS 24.008 - Especificação da Camada 3 da interface de rádio móvel (perfis de QoS)

Voltar ao Índice da Documentação

Interface Gn/Gp do OmniPGW - *por Omnitouch Network Services*

Suporte a PDN IPv6 / Dual-Stack (IPv4v6)

Guia completo para operar conexões PDN compatíveis com IPv6 através do OmniPGW (plano de controle) e OmniUPF (plano do usuário). Esta página cobre como o tipo de PDN é negociado e sinalizado, a configuração necessária em ambos os nós e como a alcance IPv6 do UE é alcançado com roteamento dinâmico.

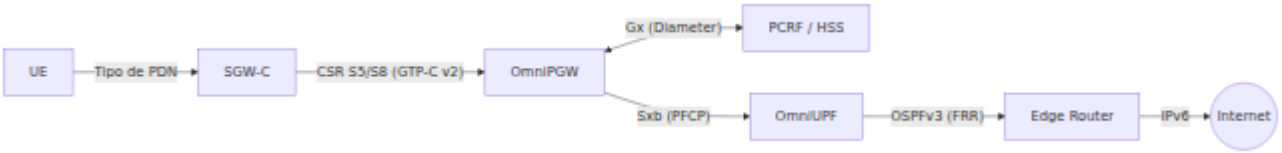
Para o modelo detalhado de pool de endereços do UE (correspondência de APN, alocação, manipulação de colisões), consulte Alocação de IP do UE. Para as interfaces PFCP e Gx em geral, consulte PFCP/Sxb e Diameter Gx.

Índice

- Visão Geral
- Tipos de PDN
- Fluxo de Ponta a Ponta
- Configuração do PGW-C
- Plano do Usuário (OmniUPF)
- Tabelas de Referência
- Solução de Problemas

Visão Geral

Uma conexão PDN pode ser **IPv4**, **IPv6** ou **IPv4v6** (dual-stack). O tipo de PDN é solicitado pelo UE/SGW na Solicitação de Criação de Sessão GTPv2 e confirmado pelo PGW-C na resposta. Fornecer uma PDN IPv6 de ponta a ponta requer comportamento coordenado em três interfaces e ambos os nós:



Nó / Interface	Responsabilidade pelo IPv6
OmniPGW — S5/S8	Negociar tipo de PDN; alocar endereço(s) do UE; retornar a Alocação de Endereço PDN (PAA)
OmniPGW — Gx	Transmitir o endereço do UE para o PCRF: `Framed-IP-Address` (IPv4) para PDNs IPv4/IPv4v6, `Framed-IPv6-Prefix` para PDNs apenas IPv6
OmniPGW — Sxb	Sinalizar o endereço IPv6 do UE e o tipo de PDN para o UPF via PFCP
OmniUPF — plano de dados	Programar a busca de downlink IPv6; encaminhar/encapsular tráfego de usuário IPv6
OmniUPF — FRR	Anunciar cada rota de host IPv6 `/128` do UE para o roteador de borda via OSPFv3

Referências: 3GPP TS 29.274 (GTPv2-C), 3GPP TS 29.212 (Gx), 3GPP TS 29.244 (PFCP).

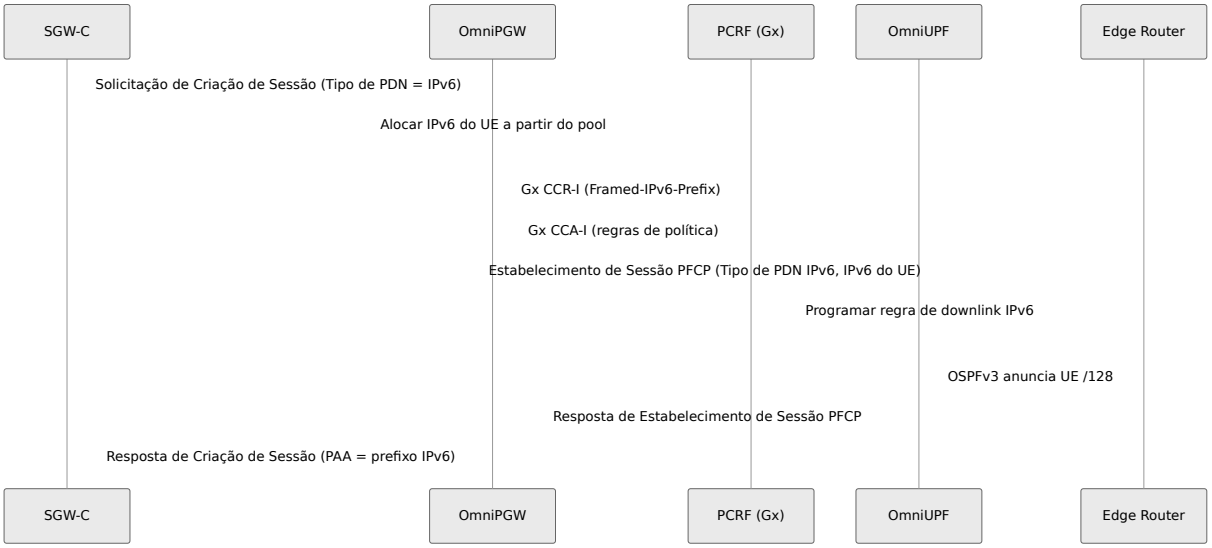
Tipos de PDN

Valor	Tipo de PDN	UE recebe	Conteúdo do PAA (TS 29.274 §8.14)
1	IPv4	Endereço IPv4	Endereço IPv4 de 4 bytes
2	IPv6	Prefixo IPv6 (/64)	octeto de comprimento do prefixo + endereço de 16 bytes
3	IPv4v6	Endereço IPv4 + Prefixo IPv6	octeto de comprimento do prefixo + 16 bytes IPv6 + 4 bytes IPv4

O PGW-C deriva o tipo efetivo de PDN do **IE Tipo de PDN**, voltando para o tipo do **IE PAA** se o IE Tipo de PDN estiver ausente. O mesmo tipo negociado determina o PAA retornado ao SGW, o IE Tipo de PDN PFCP enviado ao UPF, e qual família/famílias de endereços são alocadas.

> **IPv4v6 sem pool IPv6.** Se uma PDN IPv4v6 for solicitada, mas o APN correspondente não tiver faixa IPv6 configurada, o PGW-C **não** falha; ele degrada a PDN para apenas IPv4 ou a rejeita, dependendo se o APN é um APN de emergência (SOS). Veja solicitação IPv4v6 quando o APN não tem pool IPv6.

Fluxo de Ponta a Ponta



Uma Solicitação de Criação de Sessão retransmitida (mesmo número de sequência) é tratada idempotentemente e produz a mesma resposta.

Configuração do PGW-C

Pool de Endereços IPv6 do UE

Os endereços do UE são extraídos de pools por APN sob a chave `ue.subnet_map` em `runtime.exs`. Adicione um CIDR IPv6 junto com o CIDR IPv4 para qualquer APN que deva oferecer IPv6 ou dual-stack. O pool é analisado automaticamente pela família de endereços.

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      # Nome do APN => lista de strings CIDR (IPv4 e/ou IPv6)
      default: [
        "100.64.16.0/24",
        "2001:db8:6f85:70::/64"
      ]
    }
  }
```

Parâmetro	Tipo	Obrigatório	Padrão	Descrição
`ue.subnet_map`	Mapa	Sim	-	Mapeia um nome de AP (ou `:default`) para um lista de strings CIDR. `:default` é o fallback quando nenhuma entrada específica de APN corresponde.
(itens da lista)	String (CIDR)	Sim	-	Um prefixo IPv4 (por exemplo, `100.64.16.0/24`) ou IPv (por exemplo, `2001:db8:6f85:70::/6` A família de endereços detectada a partir da string. Endereços IPv6 d UE são extraídos dentro do prefixo configurado um `/64` é sinalizado pa o UE.

> Para correspondência de regex de APN, múltiplas sub-redes por APN, endereços estáticos e o algoritmo de alocação, consulte Alocação de IP do UE.

Um APN com **ambos** um CIDR IPv4 e um CIDR IPv6 suporta PDNs IPv4, IPv6 e IPv4v6; o PGW-C aloca de qual família/famílias o tipo de PDN negociado requer (para IPv4v6, um endereço de cada).

Gx (Framed-IPv6-Prefix)

Nenhuma configuração adicional é necessária para Gx. O PGW-C define o AVP de endereço framed Gx a partir do endereço do UE por família:

- **PDN apenas IPv6** — o prefixo IPv6 do UE é transmitido usando o **`Framed-IPv6-Prefix`** AVP (RFC 3162 §2.3), não `Framed-IP-Address` (que carrega apenas IPv4).
- **PDNs IPv4 e IPv4v6** — o endereço IPv4 do UE é transmitido em `Framed-IP-Address`. Para IPv4v6, o prefixo IPv6 **não** é carregado no Gx (os AVPs de endereço framed do CCR são derivados do endereço IPv4); o prefixo IPv6 ainda é sinalizado para o UPF via PFCP e para o UE no PAA.

Para um assinante apenas IPv6, o PCRF deve ser provisionado e capaz de processar `Framed-IPv6-Prefix`; caso contrário, o Credit-Control-Request é descartado e a sessão falha — veja Solução de Problemas.

Plano do Usuário (OmniUPF)

Para que uma PDN IPv6 carregue tráfego, o OmniUPF deve estar habilitado para IPv6, programar a busca de downlink IPv6 a partir da sessão PFCP que o PGW-C envia e anunciar cada rota de host IPv6 `/128` para o roteador de borda (via OSPFv3) para que o tráfego de downlink seja atraído para o UPF.

Essas são preocupações do plano do usuário configuradas no OmniUPF, não no PGW-C. Consulte o **guia OmniUPF IPv6 / Dual-Stack** para configurações de kernel, o pool de UE do UPF e anúncio de rotas OSPFv3. Da perspectiva do PGW-C, o único requisito é que o UPF selecionado suporte IPv6 — o PGW-C sinaliza o endereço IPv6 do UE e o tipo de PDN via PFCP (veja PFCP/Sxb).

Tabelas de Referência

Alocação de Endereço PDN (PAA) — TS 29.274 §8.14

Tipo de PDN	Layout do campo de endereço
IPv4	4 octetos: endereço IPv4
IPv6	1 octeto de comprimento do prefixo + 16 octetos de prefixo/endereço IPv6
IPv4v6	1 octeto de comprimento do prefixo + 16 octetos IPv6 + 4 octetos IPv4

AVPs de Endereço Framed Gx

AVP	Código	Família	Formato	Referência
`Framed-IP-Address`	8	IPv4	endereço de 4 bytes	RFC 4005
`Framed-IPv6-Prefix`	97	IPv6	octeto reservado + comprimento do prefixo + prefixo	RFC 3162 §2.3

IE Tipo de PDN PFCP — TS 29.244

Valor	Significado
1	IPv4
2	IPv6
3	IPv4v6

Solução de Problemas

Gx CCR expira para sessões IPv6

Sintomas: Sessões IPv4 para um assinante são bem-sucedidas, mas sessões IPv6/IPv4v6 retornam a causa `remote_peer_not_responding`; o Gx Credit-Control-Request expira.

Causas possíveis:

- O PCRF não pode processar o AVP `Framed-IPv6-Prefix` e descarta silenciosamente a solicitação.
- O assinante não está provisionado para IPv6 no PCRF/HSS.

Resolução:

1. Confirme que o assinante está provisionado e habilitado no PCRF/HSS.
2. Verifique se o PCRF aceita `Framed-IPv6-Prefix` para Gx CCR-I.
3. Compare com uma sessão IPv4 para o mesmo assinante para isolar o comportamento específico de IPv6.

A sessão IPv6 é aceita, mas o tráfego de downlink não chega ao UE

Sintomas: A Resposta de Criação de Sessão carrega um PAA IPv6, o uplink funciona, mas o tráfego da rede para o UE é descartado.

Esta é uma preocupação do plano do usuário (OmniUPF) — o PGW-C já alocou o endereço e o sinalizou via PFCP. A causa usual é que a rota de host `/128` do UE não está alcançando o roteador de borda (OSPFv3 não está em execução/adjacente no UPF, ou atraso de convergência). Consulte o guia OmniUPF IPv6 / Dual-Stack para configuração de anúncio de rotas e solução de problemas.

Nenhum endereço IPv6 é alocado

Sintomas: Uma solicitação de PDN IPv6 é rejeitada ou nenhum PAA IPv6 é retornado.

Causas possíveis:

- O APN correspondente (ou `:default`) não tem CIDR IPv6 em `ue.subnet_map`.
- O pool IPv6 configurado está esgotado.

Resolução:

1. Adicione um CIDR IPv6 à entrada do APN em `ue.subnet_map` (veja Configuração do PGW-C).
2. Confirme que o pool tem endereços livres.

Solicitação IPv4v6 quando o APN não tem pool IPv6

Sintomas: Um UE solicita uma PDN **IPv4v6** contra um APN que tem apenas um CIDR IPv4 configurado em `ue.subnet_map` (sem faixa IPv6). Dependendo do APN, a sessão pode ser estabelecida **apenas IPv4** ou ser **rejeitada**.

Comportamento: O alocador IPv6 relata que não há pool IPv6 quando o APN correspondente tem uma lista IPv6 vazia, em vez de falhar o processo da sessão. O procedimento da sessão então escolhe pelo tipo de APN (per TS 29.274 / TS 23.401 tratamento de bearer de endereço único / downgrade de tipo de PDN):

Tipo de APN	Resultado	O que o UE / PCRF / UPF vê
APN de Emergência (SOS)	**Degradar para apenas IPv4** — atribuir o endereço IPv4, limpar o endereço IPv6, definir o tipo de PDN negociado como `IPv4`.	PAA, Tipo de PDN PFCP e IP do UE PFCP são apenas IPv4; Gx carrega `Framed-IP-Address`. O **P-CSCF ainda é retornado no PCO** para que o registro de emergência IMS possa prosseguir. Um `warning` é registrado.
APN Regular	**Rejeitar** — a sessão não é silenciosamente degradada; `{:error, :no_ipv6_pool}` é exposto para que a má configuração (ou solicitação genuinamente não suportada) seja visível. Um `error` é registrado.	A Criação de Sessão falha com uma causa não-sucedida.

Um APN de emergência é identificado pelo seu nome: o APN contém `"sos"` (por exemplo, `sos`, `sos.mnc410.mcc310.gprs`). Solicitações apenas IPv6 contra um APN sem pool IPv6 também são rejeitadas.

> Na prática, o MME normalmente degrada a solicitação IPv4v6 de um assinante regular para seu IPv4 assinado antes que a solicitação chegue ao PGW-C, então o caminho de rejeição é um respaldo para uma solicitação IPv4v6 que realmente chega sem um pool IPv6. O caminho de degradação

> SOS mantém chamadas de emergência (por exemplo, 911/112) conectadas em uma implantação apenas IPv4.

Resolução:

1. Para um APN regular que deve oferecer dual-stack, adicione um CIDR IPv6 à entrada do APN em `ue.subnet_map` (veja Configuração do PGW-C).
2. Para um APN de emergência (SOS) em uma implantação apenas IPv4, nenhuma ação é necessária — a degradação apenas IPv4 é intencional. Adicione um CIDR IPv6 apenas se você pretende que PDNs SOS sejam dual-stack.

Guia de Monitoramento e Métricas do OmniPGW

Integração com Prometheus e Monitoramento Operacional

por Omnitouch Network Services

Índice

1. Visão Geral
2. Endpoint de Métricas
3. Métricas Disponíveis
4. Configuração do Prometheus
5. Painéis do Grafana
6. Alertas
7. Monitoramento de Desempenho
8. Solução de Problemas de Métricas

Visão Geral

O OmniPGW fornece duas abordagens complementares de monitoramento:

1. API REST OAM em Tempo Real (coberta brevemente aqui, detalhada nos respectivos documentos de interface)

- Visualizador de sessões ao vivo (`GET /api/sessions`)
- Status do par PFCP (`GET /api/upf`)
- Conectividade do par Diameter (`GET /api/diameter`)

- Inspeção de sessão individual (`GET /api/sessions/<imsi>`)

2. Métricas do Prometheus (foco principal deste documento)

- Tendências históricas e análise
- Alertas e notificações
- Métricas de desempenho
- Planejamento de capacidade

Este documento se concentra nas **métricas do Prometheus**. Para a API REST OAM (servida sobre HTTPS na porta `8443`, sob o prefixo `/api`), veja:

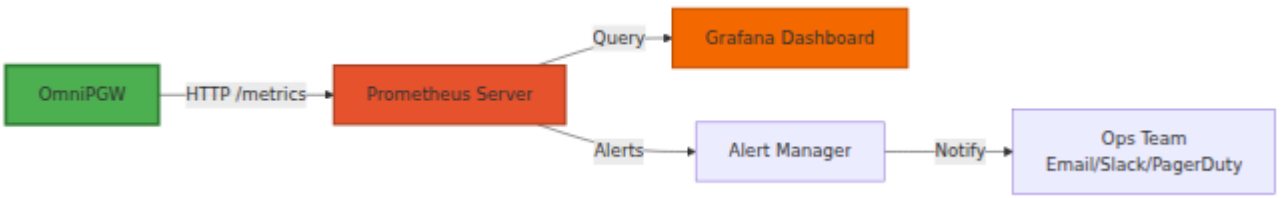
- Gerenciamento de Sessões - API de Sessões
- Interface PFCP - API de Monitoramento
- Diameter Gx - API de Monitoramento de Par

Visão Geral das Métricas do Prometheus

O OmniPGW expõe **métricas compatíveis com Prometheus** para monitoramento abrangente da saúde do sistema, desempenho e capacidade. Isso permite que as equipes de operações:

- **Monitorar a Saúde do Sistema** - Acompanhar sessões ativas, alocações e erros
- **Planejamento de Capacidade** - Compreender tendências de utilização de recursos
- **Análise de Desempenho** - Medir a latência de manipulação de mensagens
- **Alertas** - Notificação proativa de problemas
- **Depuração** - Identificar causas raízes de problemas

Arquitetura de Monitoramento



Endpoint de Métricas

Configuração

Ative as métricas em `config/runtime.exs`:

```
config :pgw_c,
  metrics: %{
    enabled: true,
    ip_address: "127.0.0.42",       # Endereço de ligação padrão
(use "0.0.0.0" para expor em todas as interfaces)
    port: 9090,                      # Porta HTTP padrão
    registry_poll_period_ms: 10_000  # Intervalo de polling para
gauges polled (padrão 10s)
  }
```

> **Nota:** Os padrões enviados ligam o endpoint de métricas a `127.0.0.42:9090`. Para raspá-lo de um Prometheus remoto, defina `ip_address` para um endereço acessível (por exemplo, `"0.0.0.0"`).

Acessando Métricas

Endpoint HTTP:

```
http://<omnipgw_ip>:<port>/metrics
```

Exemplo:

```
curl http://10.0.0.20:9090/metrics
```

Formato de Saída

As métricas são expostas no **formato de texto do Prometheus**:

```
# HELP teid_registry_count The number of TEID registered to
sessions
# TYPE teid_registry_count gauge
teid_registry_count 150

# HELP address_registry_count The number of addresses registered
to sessions
# TYPE address_registry_count gauge
address_registry_count 150

# HELP s5s8_inbound_messages_total The total number of messages
received from S5/S8 peers
# TYPE s5s8_inbound_messages_total counter
s5s8_inbound_messages_total{message_type="create_session_request"}
1523
s5s8_inbound_messages_total{message_type="delete_session_request"}
1487
```

Métricas Disponíveis

O OmniPGW expõe as seguintes categorias de métricas:

Métricas de Sessão

Contagens de Sessões Ativas:

Nome da Métrica	Tipo	Descrição
`teid_registry_count`	Gauge	Sessões S5/S8 ativas (contagem de TEID)
`seid_registry_count`	Gauge	Sessões PFCP ativas (contagem de SEID)
`gx_session_id_registry_count`	Gauge	Sessões Gx ativas (contagem de Diameter Session-ID)
`gy_session_id_registry_count`	Gauge	Sessões Gy ativas (contagem de Diameter Session-ID)
`session_registry_count`	Gauge	Sessões ativas (pares IMSI, EBI)
`address_registry_count`	Gauge	Endereços IP de UE alocados
`charging_id_registry_count`	Gauge	IDs de cobrança ativos (veja Formato de CDR de Dados para registros de cobrança CDR)
`sxb_sequence_number_registry_count`	Gauge	Respostas PFCP pendentes (aguardando resposta)
`s5s8_sequence_number_registry_count`	Gauge	Respostas S5/S8 pendentes (aguardando resposta)

Nome da Métrica	Tipo	Descrição
`sxb_peer_registry_count`	Gauge	Número de processos de pares PFCP registrados

Uso:

```
# Sessões ativas atuais
teid_registry_count

# Taxa de criação de sessões (por segundo)
rate(teid_registry_count[5m])

# Sessões de pico na última hora
max_over_time(teid_registry_count[1h])
```

Ciclo de Vida de Sessão e Bearer:

Contadores incrementados à medida que sessões PDN e bearers são criados, modificados e encerrados.

Nome da Métrica	Tipo	Rótulos	Descrição
`pgw_session_create_total`	Counter	`result`, `cause`, `pdn_type`, `rat_type`	Criações de sessões PDN, por resultado (sucesso/falha), causa GTP, tipo de PDN e tipo de RAT
`pgw_session_delete_total`	Counter	`pdn_type`	Exclusões de sessões PDN, por tipo de PDN
`pgw_session_modify_total`	Counter	-	Solicitações de Modificação de Bearer tratadas com sucesso
`pgw_bearer_create_total`	Counter	`type`, `qci`	Bearers criados, por tipo (padrão/dedicado) e QCI
`pgw_bearer_delete_total`	Counter	`type`	Bearers excluídos, por tipo (padrão/dedicado)

Uso:

```
# Taxa de sucesso de configuração de sessão (últimos 5m)
sum(rate(pgw_session_create_total{result="success"}[5m])) /
sum(rate(pgw_session_create_total[5m]))

# Sessões de configuração falhadas por causa
sum(rate(pgw_session_create_total{result="failure"}[5m])) by
(cause)
```

Pool de Endereços IP de UE

Os gauges de utilização do pool são polled periodicamente; falhas de alocação são contadas à medida que ocorrem. Veja Alocação de IP de UE.

Nome da Métrica	Tipo	Rótulos	Descriç
`ue_pool_addresses_allocated`	Gauge	`pool`, `ip_version`	Endereç IP de UE atualme alocados de um p
`ue_pool_addresses_total`	Gauge	`pool`, `ip_version`	Total de endereç IP de UE disponív em um pool
`ue_pool_addresses_utilization_ratio`	Gauge	`pool`, `ip_version`	Fração (0.0–1.0 de um p atualme alocado
`ue_ip_allocation_failures_total`	Counter	`ip_version`, `reason`	Falhas d alocaçã de IP de (por exemplo exaustã do pool, colisão estática

Uso:

```
# Pools acima de 80% de utilização
ue_pool_addresses_utilization_ratio > 0.8

# Taxa de falha de alocação por razão
sum(rate(ue_ip_allocation_failures_total[5m])) by (reason)
```

Contadores de Mensagens

Mensagens S5/S8 (GTP-C):

Nome da Métrica	Tipo	Rótulos	Descrição
`s5s8_inbound_messages_total`	Counter	`message_type`	Total de mensagens S5/S8 recebidas
`s5s8_outbound_messages_total`	Counter	`message_type`	Total de mensagens S5/S8 enviadas
`s5s8_inbound_errors_total`	Counter	`message_type`	Erros de processamento S5/S8 recebidos
`s5s8_outbound_errors_total`	Counter	`message_type`	Erros de processamento S5/S8 enviados

Tipos de Mensagens:

- `create_session_request`
- `create_session_response`

- `delete_session_request`
- `delete_session_response`
- `create_bearer_request`
- `delete_bearer_request`

Mensagens Sxb (PFCP):

Nome da Métrica	Tipo	Rótulos	Descri
`sxb_inbound_messages_total`	Counter	`message_type`	Total de mensage PFCP rece
`sxb_outbound_messages_total`	Counter	`message_type`	Total de mensage PFCP env
`sxb_inbound_errors_total`	Counter	`message_type`	Erros de processa PFCP rece
`sxb_outbound_errors_total`	Counter	`message_type`	Erros de processa PFCP env
`sxb_outbound_responses_total`	Counter	`message_type`, `cause_code_class`, `peer_ip`	Resposta PFCP env categoriz por class código de causa PF par UPF

Tipos de Mensagens:

- `association_setup_request`
- `association_setup_response`

- `heartbeat_request`
- `heartbeat_response`
- `session_establishment_request`
- `session_establishment_response`
- `session_modification_request`
- `session_deletion_request`

Mensagens Gx (Diameter):

Nome da Métrica	Tipo	Rótulos	Descri
`gx_inbound_messages_total`	Counter	`message_type`	Total de mensage Diameter recebidas
`gx_outbound_messages_total`	Counter	`message_type`	Total de mensage Diameter enviadas
`gx_inbound_errors_total`	Counter	`message_type`	Erros de processar Diameter recebidos
`gx_outbound_errors_total`	Counter	`message_type`	Erros de processar Diameter enviados
`gx_outbound_responses_total`	Counter	`message_type`, `result_code_class`, `diameter_host`	Resposta Diameter enviadas categoriz por classe código de resultado host par

Tipos de Mensagens:

- `gx_CCA` (Credit-Control-Answer)
- `gx_CCR` (Credit-Control-Request)
- `gx_RAA` (Re-Auth-Answer)
- `gx_RAR` (Re-Auth-Request)

Classes de Código de Resultado (para `gx_outbound_responses_total`):

- `2xxx` - Respostas de sucesso (por exemplo, 2001 DIAMETER_SUCCESS)
- `3xxx` - Erros de protocolo (por exemplo, 3001 DIAMETER_COMMAND_UNSUPPORTED)
- `4xxx` - Falhas transitórias (por exemplo, 4001 DIAMETER_AUTHENTICATION_REJECTED)
- `5xxx` - Falhas permanentes (por exemplo, 5012 DIAMETER_UNABLE_TO_COMPLY)

Exemplos de Uso:

```
# Monitorar taxa de sucesso de respostas Gx
sum(rate(gx_outbound_responses_total{result_code_class="2xxx"}[5m]))
sum(rate(gx_outbound_responses_total[5m])) * 100

# Acompanhar falhas por host PCRF
rate(gx_outbound_responses_total{result_code_class!="2xxx"}[5m]) by (

# Contar total de mensagens Re-Auth-Answer bem-sucedidas
gx_outbound_responses_total{message_type="gx_RAA",result_code_class="

# Alertar sobre alta taxa de falhas para PCRF específico
rate(gx_outbound_responses_total{result_code_class=~"4xxx|5xxx",diame
[5m]) > 0.1
```

Mensagens Gy (Diameter):

Contadores de mensagens de cobrança online Gy (OCS). Veja Diameter Gy para a interface.

Nome da Métrica	Tipo	Rótulos	Descri
`gy_inbound_messages_total`	Counter	`message_type`	Total de mensage recebidas
`gy_outbound_messages_total`	Counter	`message_type`	Total de mensage enviadas
`gy_inbound_errors_total`	Counter	`message_type`	Erros de processar Gy receb
`gy_outbound_errors_total`	Counter	`message_type`	Erros de processar Gy enviad
`gy_inbound_responses_total`	Counter	`cc_request_type`, `result_code_class`, `diameter_host`	Credit-Co Answers recebidos OCS, pelo pedido qu responde classe de código de resultado host OCS
`gy_outbound_responses_total`	Counter	`message_type`, `result_code_class`, `diameter_host`	Resposta enviadas OCS (Re-A Answer p um OCS F por classe código de

Nome da Métrica	Tipo	Rótulos	Descri
			resultado host OCS
`gy_usage_clamped_total`	Counter	-	CCR-Ts cu Used-Ser Unit foi limitado a porque o relatado f maior qu uso medi Um valor diferente zero sign que o OC está sub-faturado

Tipos de Mensagens (`message_type`):

- `ro_CCR` (Credit-Control-Request)
- `ro_CCA` (Credit-Control-Answer)
- `ro_RAR` (Re-Auth-Request)
- `ro_RAA` (Re-Auth-Answer)

Tipos de Solicitação CC (`cc_request_type`, para `gy_inbound_responses_total`):

- `initial` - resposta a um CCR-I (início da sessão, primeira concessão de cota)
- `update` - resposta a um CCR-U (relatório de uso, próxima concessão de cota)
- `termination` - resposta a um CCR-T (fim da sessão, uso final)
- `event` - resposta a um CCR baseado em evento

Classes de Código de Resultado (para `gy_inbound_responses_total` e `gy_outbound_responses_total`): as mesmas classes que para Gx (`2xxx`, `3xxx`, `4xxx`, `5xxx`, `other`). No Gy, `4012` DIAMETER_CREDIT_LIMIT_REACHED é um `4xxx` e `5030` DIAMETER_USER_UNKNOWN é um `5xxx`.

A `result_code_class` é o Código de Resultado de nível superior do CCA. Um OCS pode responder `2001` no nível superior e ainda assim negar ou redirecionar um único serviço em seu Controle de Crédito de Múltiplos Serviços.

Exemplos de Uso:

```
# Taxa de sucesso de CCA por tipo de solicitação
sum by (cc_request_type) (rate(gy_inbound_responses_total{result_code
[5m]))
  / sum by (cc_request_type) (rate(gy_inbound_responses_total[5m])) *

# CCR-Is falhados (novas sessões que o OCS recusou), por host OCS
sum by (diameter_host)
(rate(gy_inbound_responses_total{cc_request_type="initial",result_coc
[5m]))

# CCRs enviados sem CCA de volta (timeouts do OCS ou respostas perdic
sum by (instance) (rate(gy_outbound_messages_total{message_type="ro_C
  - sum by (instance) (rate(gy_inbound_responses_total[5m]))

# Re-Auth-Answers enviadas ao OCS que não foram bem-sucedidas
rate(gy_outbound_responses_total{message_type="ro_RAA",result_code_cl
[5m])
```

Mensagens Gn/Gp (GTPv1-C):

Contadores de mensagens de controle Gn/Gp GGSN (interoperabilidade SGSN 2G/3G). Veja Interface Gn.

Nome da Métrica	Tipo	Rótulos	Descrição
`gn_inbound_messages_total`	Counter	`message_type`	Total de mensagens Gn/Gp recebidas (do SGSN)
`gn_outbound_messages_total`	Counter	`message_type`	Total de mensagens Gn/Gp enviadas (para SGSN)
`gn_inbound_errors_total`	Counter	`message_type`	Erros de processamento Gn/Gp recebidos
`gn_outbound_errors_total`	Counter	`message_type`	Respostas de erro Gn/Gp enviadas ao SGSN

Tratamento de Erros:

Nome da Métrica	Tipo	Rótulos	Descrição
`rescues_total`	Counter	`module`, `function`	Total de blocos de resgate atingidos (tratamento de exceções)

Métricas de Latência

Duração do Processamento de Mensagens Recebidas:

Nome da Métrica	Tipo	Rótulos	
`s5s8_inbound_handling_duration`	Histogram	`request_message_type`	
`gn_inbound_handling_duration`	Histogram	`request_message_type`	
`sxb_inbound_handling_duration`	Histogram	`request_message_type`	
`gx_inbound_handling_duration`	Histogram	`request_message_type`	

Nome da Métrica	Tipo	Rótulos	
			r ((c c e r
`gy_inbound_handling_duration`	Histogram	`request_message_type`	T r c r ((c c e r

Duração da Transação de Saída:

Nome da Métrica	Tipo	Rótulos
`s5s8_outbound_transaction_duration`	Histogram	`request_message_type`
`gn_outbound_transaction_duration`	Histogram	`request_message_type`
`sxb_outbound_transaction_duration`	Histogram	`request_message_type`
`gx_outbound_transaction_duration`	Histogram	`request_message_type`
`gy_outbound_transaction_duration`	Histogram	`request_message_type`

Buckets (segundos):

- Valores: 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0, 5.0
- (100µs, 500µs, 1ms, 5ms, 10ms, 50ms, 100ms, 500ms, 1s, 5s)

Uso:

```
# Latência S5/S8 percentil 95
histogram_quantile(0.95,
  rate(s5s8_inbound_handling_duration_bucket[5m])
)

# Latência média PFCP
rate(sxb_inbound_handling_duration_sum[5m]) /
rate(sxb_inbound_handling_duration_count[5m])
```

Monitoramento de Saúde do UPF

Métricas de Par UPF:

Nome da Métrica	Tipo	Rótulos	Descrição
`upf_peers_total`	Gauge	-	Total de pares UPF registrados
`upf_peers_healthy`	Gauge	-	Número de pares UPF saudáveis (associados + batimentos de coração OK)
`upf_peers_unhealthy`	Gauge	-	Número de pares UPF não saudáveis
`upf_peers_associated`	Gauge	-	Número de pares UPF com associação PFCP ativa
`upf_peers_unassociated`	Gauge	-	Número de pares UPF sem associação PFCP
`upf_peer_associated`	Gauge	`peer_ip`	Se um UPF específico tem uma associação PFCP ativa (1/0)
`upf_peer_healthy`	Gauge	`peer_ip`	Status de saúde de um UPF específico (1=saudável, 0=não saudável)
`upf_peer_missed_heartbeats`	Gauge	`peer_ip`	Batimentos de coração perdidos consecutivos para um UPF específico

Nome da Métrica	Tipo	Rótulos	Descrição
`upf_heartbeat_rtt`	Histogram	`peer_ip`	Tempo de ida e volta de solicitação/resposta de batimento de coração PFCP (segundos)

Uso:

```
# Monitorar saúde do pool UPF
upf_peers_healthy / upf_peers_total

# Alertar sobre UPFs não saudáveis
upf_peers_unhealthy > 0

# Acompanhar saúde de um UPF específico
upf_peer_healthy{peer_ip="10.98.0.20"}

# Identificar UPFs com problemas de batimento de coração
upf_peer_missed_heartbeats > 2
```

Exemplos de Alertas:

```yaml
# Alertar quando UPF ficar fora do ar
- alert: UPF_Peer_Down
  expr: upf_peer_healthy == 0
  for: 1m
  labels:
    severity: critical
  annotations:
    summary: "UPF {{ $labels.peer_ip }} está fora do ar"
    description: "Par UPF não respondendo a batimentos de coração
PFCP"

# Alertar quando múltiplos UPFs estiverem fora do ar
- alert: UPF_Pool_Degraded
  expr: (upf_peers_healthy / upf_peers_total) < 0.5
  for: 2m
  labels:
    severity: critical
  annotations:
    summary: "Pool UPF degradado"
    description: "Apenas {{ $value | humanizePercentage }} dos
UPFs estão saudáveis"

# Aviso sobre batimentos de coração perdidos
- alert: UPF_Heartbeat_Issues
  expr: upf_peer_missed_heartbeats > 2
  for: 30s
  labels:
    severity: warning
  annotations:
    summary: "UPF {{ $labels.peer_ip }} problemas de batimento de
coração"
    description: "{{ $value }} batimentos de coração perdidos
consecutivos"
```

Monitoramento de Saúde do P-CSCF

Métricas do Servidor P-CSCF:

Nome da Métrica	Tipo	Rótulos	Descrição
`pcscf_fqdns_total`	Gauge	-	Total de FQDNs P-CSCF sendo monitorados
`pcscf_fqdns_resolved`	Gauge	-	FQDNs P-CSCF resolvidos com sucesso via DNS
`pcscf_fqdns_failed`	Gauge	-	FQDNs P-CSCF que falharam na resolução DNS
`pcscf_servers_total`	Gauge	-	Total de servidores P-CSCF descobertos
`pcscf_servers_healthy`	Gauge	`fqdn`	Servidores P-CSCF saudáveis por FQDN
`pcscf_servers_unhealthy`	Gauge	`fqdn`	Servidores P-CSCF não saudáveis por FQDN

Veja: Guia de Monitoramento P-CSCF para rastreamento detalhado da saúde IMS.

Conectividade de Par Diameter

Polled periodicamente da pilha Diameter.

Nome da Métrica	Tipo	Rótulos	Descrição
`diameter_peers_connected`	Gauge	`application`	Número de pares Diameter atualmente conectados, por aplicação (`gx`/`gy`/`all`)

Uso:

```
# Alertar quando nenhum par Gx (PCRF) estiver conectado
diameter_peers_connected{application="gx"} == 0
```

Métricas de Licença

Status da Licença:

Nome da Métrica	Tipo	Descrição
`license_status`	Gauge	Status atual da licença (1 = válido, 0 = inválido)

Uso:

```
# Verificar se a licença é válida
license_status == 1

# Alertar sobre licença inválida
license_status == 0
```

Exemplo de Alerta:

```
- alert: PGW_C_License_Invalid
  expr: license_status == 0
  for: 1m
  labels:
    severity: critical
  annotations:
    summary: "Licença PGW-C inválida ou expirada"
    description: "O status da licença é inválido - solicitações de criação de sessão estão sendo bloqueadas"
```

Impacto de Licença Inválida:

Quando a licença é inválida ou o servidor de licença está inacessível, **Solicitações de Criação de Sessão serão rejeitadas** com o código de causa GTP-C **"Sem recursos disponíveis" (73)**. Isso é visível nas capturas de pacotes conforme mostrado abaixo:

Captura do Wireshark mostrando Resposta de Criação de Sessão com causa "Sem recursos disponíveis" quando a licença é inválida

Notas:

- Nome do produto registrado no servidor de licença: `omnipgwc`
- A URL do servidor de licença está configurada em `config/runtime.exs` sob `:license_client`
- Quando a licença é inválida (`license_status == 0`), as solicitações de criação de sessão são bloqueadas com o código de causa GTP-C 73 (Sem recursos disponíveis)
- A interface do usuário e a monitorização permanecem acessíveis independentemente do status da licença
- Os pares Diameter, GTP-C e PFCP continuam a manter conexões
- Sessões existentes não são afetadas - apenas a criação de novas sessões é bloqueada

Métricas do Sistema

Métricas da VM Erlang:

Nome da Métrica	Tipo	Descrição
`vm_memory_total`	Gauge	Memória total da VM (bytes)
`vm_memory_processes`	Gauge	Memória usada pelos processos
`vm_memory_system`	Gauge	Memória usada pelo sistema
`vm_system_process_count`	Gauge	Total de processos Erlang
`vm_system_port_count`	Gauge	Total de portas abertas

Configuração do Prometheus

Configuração de Scrape

Adicione OmniPGW ao `prometheus.yml` do Prometheus:

```yaml
# prometheus.yml
global:
  scrape_interval: 15s
  evaluation_interval: 15s

scrape_configs:
  - job_name: 'omnipgw'
    static_configs:
      - targets: ['10.0.0.20:9090']
        labels:
          instance: 'omnipgw-01'
          environment: 'production'
          site: 'datacenter-1'
```

Múltiplas Instâncias do OmniPGW

```yaml
scrape_configs:
  - job_name: 'omnipgw'
    static_configs:
      - targets:
          - '10.0.0.20:9090'
          - '10.0.0.21:9090'
          - '10.0.0.22:9090'
        labels:
          environment: 'production'
```

Descoberta de Serviços

Kubernetes:

```yaml
scrape_configs:
  - job_name: 'omnipgw'
    kubernetes_sd_configs:
      - role: pod
    relabel_configs:
      - source_labels: [__meta_kubernetes_pod_label_app]
        action: keep
        regex: omnipgw
      - source_labels: [__meta_kubernetes_pod_ip]
        target_label: __address__
        replacement: '${1}:9090'
```

Verificação

Teste de scrape:

```
# Verifique os alvos do Prometheus
curl http://prometheus:9090/api/v1/targets

# Consultar uma métrica
curl 'http://prometheus:9090/api/v1/query?
query=teid_registry_count'
```

Painéis do Grafana

Configuração do Painel

1. Adicione a Fonte de Dados do Prometheus:

```
Configuração → Fontes de Dados → Adicionar fonte de dados →
Prometheus
URL: http://prometheus:9090
```

2. Importar Painel:

Crie um novo painel ou importe de JSON.

Painéis Principais

Painel 1: Sessões Ativas

```
# Consulta
teid_registry_count

# Tipo de Painel: Gauge
# Limiares:
#   Verde: < 5000
#   Amarelo: 5000-8000
#   Vermelho: > 8000
```

Painel 2: Taxa de Sessão

```
# Consulta
rate(s5s8_inbound_messages_total{message_type="create_session_request
[5m])

# Tipo de Painel: Gráfico
# Unidade: solicitações/segundo
```

Painel 3: Utilização do Pool de IP

```
# Consulta (para sub-rede /24 com 254 IPs)
(address_registry_count / 254) * 100

# Tipo de Painel: Gauge
# Unidade: porcentagem (0-100)
# Limiares:
#   Verde: < 70%
#   Amarelo: 70-85%
#   Vermelho: > 85%
```

Painel 4: Latência da Mensagem (Percentil 95)

```
# Consulta
histogram_quantile(0.95,

rate(s5s8_inbound_handling_duration_bucket{request_message_type="crea
[5m])
)

# Tipo de Painel: Gráfico
# Unidade: milissegundos
```

Painel 5: Taxa de Erros

```
# Consulta
rate(s5s8_inbound_errors_total[5m])

# Tipo de Painel: Gráfico
# Unidade: erros/segundo
# Limite de Alerta: > 0.1
```

Painel 6: Taxa de Sucesso da Resposta Gx

```
# Consulta: Calcular a porcentagem de respostas Gx bem-sucedidas
sum(rate(gx_outbound_responses_total{result_code_class="2xxx"}
[5m])) /
sum(rate(gx_outbound_responses_total[5m])) * 100

# Tipo de Painel: Gauge
# Unidade: porcentagem (0-100)
# Limiares:
#   Verde: > 95%
#   Amarelo: 90-95%
#   Vermelho: < 90%
```

Alternativa - Divisão por Classe de Código de Resultado:

```
# Consulta: Mostrar contagens de resposta por classe de código de
resultado
sum(rate(gx_outbound_responses_total[5m])) by (result_code_class)

# Tipo de Painel: Gráfico de Pizza ou Gráfico de Barras
# Legenda: {{ result_code_class }}
```

Alternativa - Por Status de Resposta do PCRF:

```
# Consulta: Mostrar respostas por host PCRF
sum(rate(gx_outbound_responses_total[5m])) by (diameter_host,
result_code_class)

# Tipo de Painel: Gráfico de Barras Empilhadas
# Legenda: {{ diameter_host }} - {{ result_code_class }}
```

Painel 7: Status de Saúde do UPF

```
# Consulta: Porcentagem geral de saúde do pool
(upf_peers_healthy / upf_peers_total) * 100

# Tipo de Painel: Gauge
# Unidade: porcentagem (0-100)
# Limiares:
#   Verde: 100%
#   Amarelo: 50-99%
#   Vermelho: < 50%
```

Alternativa - Status por UPF:

```
# Consulta: Saúde individual do UPF
upf_peer_healthy

# Tipo de Painel: Estatística
# Mapeamentos:
#   1 = "UP" (Verde)
#   0 = "DOWN" (Vermelho)
```

Exemplo Completo de Painel

```
{
  "dashboard": {
    "title": "OmniPGW - Painel de Operações",
    "panels": [
      {
        "title": "Sessões Ativas",
        "targets": [
          {
            "expr": "teid_registry_count",
            "legendFormat": "Sessões Ativas"
          }
        ],
        "type": "graph"
      },
      {
        "title": "Taxa de Criação de Sessão",
        "targets": [
          {
            "expr":
"rate(s5s8_inbound_messages_total{message_type=\"create_session_reque
[5m])",
            "legendFormat": "Sessões/segundo"
          }
        ],
        "type": "graph"
      },
      {
        "title": "Utilização do Pool de IP",
        "targets": [
          {
            "expr": "(address_registry_count / 254) * 100",
            "legendFormat": "Uso do Pool %"
          }
        ],
        "type": "gauge"
      },
      {
        "title": "Latência da Mensagem (p95)",
        "targets": [
          {
            "expr": "histogram_quantile(0.95,
```

```
rate(s5s8_inbound_handling_duration_bucket[5m]))",
            "legendFormat": "S5/S8 p95"
          },
          {
            "expr": "histogram_quantile(0.95,
rate(sxb_inbound_handling_duration_bucket[5m]))",
            "legendFormat": "PFCP p95"
          }
        ],
        "type": "graph"
      }
    ]
  }
}
```

Alertas

Regras de Alerta

Crie `omnipgw_alerts.yml`:

```yaml
groups:
  - name: omnipgw
    interval: 30s
    rules:
      # Alertas de Contagem de Sessões
      - alert: OmniPGW_HighSessionCount
        expr: teid_registry_count > 8000
        for: 5m
        labels:
          severity: warning
        annotations:
          summary: "Contagem de sessões alta no OmniPGW"
          description: "{{ $value }} sessões ativas (limite:
8000)"

      - alert: OmniPGW_SessionCountCritical
        expr: teid_registry_count > 9500
        for: 2m
        labels:
          severity: critical
        annotations:
          summary: "Contagem de sessões crítica no OmniPGW"
          description: "{{ $value }} sessões ativas se aproximando
da capacidade"

      # Alertas de Pool de IP
      - alert: OmniPGW_IPPoolUtilizationHigh
        expr: (address_registry_count / 254) * 100 > 80
        for: 10m
        labels:
          severity: warning
        annotations:
          summary: "Utilização alta do pool de IP no OmniPGW"
          description: "Pool de IP {{ $value }}% utilizado"

      - alert: OmniPGW_IPPoolExhausted
        expr: address_registry_count >= 254
        for: 1m
        labels:
          severity: critical
        annotations:
          summary: "Pool de IP esgotado no OmniPGW"
          description: "Nenhum IP disponível para alocação"
```

```
      # Alertas de Taxa de Erros
      - alert: OmniPGW_HighErrorRate
        expr: rate(s5s8_inbound_errors_total[5m]) > 0.1
        for: 5m
        labels:
          severity: warning
        annotations:
          summary: "Taxa de erro alta no OmniPGW"
          description: "{{ $value }} erros/segundo na interface
S5/S8"

      - alert: OmniPGW_GxErrorRate
        expr: rate(gx_inbound_errors_total[5m]) > 0.05
        for: 5m
        labels:
          severity: warning
        annotations:
          summary: "Erros Gx no OmniPGW"
          description: "{{ $value }} erros Diameter/segundo"

      # Alertas de Resposta Gx
      - alert: OmniPGW_GxResponseFailureRate
        expr: |

sum(rate(gx_outbound_responses_total{result_code_class!="2xxx"}
[5m])) /
          sum(rate(gx_outbound_responses_total[5m])) > 0.1
        for: 5m
        labels:
          severity: warning
        annotations:
          summary: "Alta taxa de falha de resposta Gx no OmniPGW"
          description: "{{ $value | humanizePercentage }} das
respostas Gx são falhas (códigos de resultado não-2xxx)"

      - alert: OmniPGW_GxPCRFFailures
        expr:
rate(gx_outbound_responses_total{result_code_class=~"4xxx|5xxx"}
[5m]) by (diameter_host) > 0.05
        for: 3m
        labels:
          severity: warning
        annotations:
```

```
          summary: "PCRF {{ $labels.diameter_host }} recebendo
respostas de falha"
          description: "{{ $value }} respostas de falha/segundo
para PCRF {{ $labels.diameter_host }}"

      # Alertas de Saúde do UPF
      - alert: OmniPGW_UPF_PeerDown
        expr: upf_peer_healthy == 0
        for: 1m
        labels:
          severity: critical
        annotations:
          summary: "Par UPF {{ $labels.peer_ip }} está fora"
          description: "UPF não respondendo aos batimentos
cardíacos do PFCP"

      - alert: OmniPGW_UPF_PoolDegraded
        expr: (upf_peers_healthy / upf_peers_total) < 0.5
        for: 2m
        labels:
          severity: critical
        annotations:
          summary: "Pool de UPF degradado"
          description: "{{ $value | humanizePercentage }} dos UPFs
estão saudáveis (< 50%)"

      - alert: OmniPGW_UPF_HeartbeatFailures
        expr: upf_peer_missed_heartbeats > 2
        for: 30s
        labels:
          severity: warning
        annotations:
          summary: "Falhas de batimento cardíaco do UPF {{
$labels.peer_ip }}"
          description: "{{ $value }} batimentos cardíacos perdidos
consecutivos"

      - alert: OmniPGW_UPF_AllDown
        expr: upf_peers_healthy == 0 and upf_peers_total > 0
        for: 30s
        labels:
          severity: critical
        annotations:
          summary: "Todos os pares UPF estão fora"
```

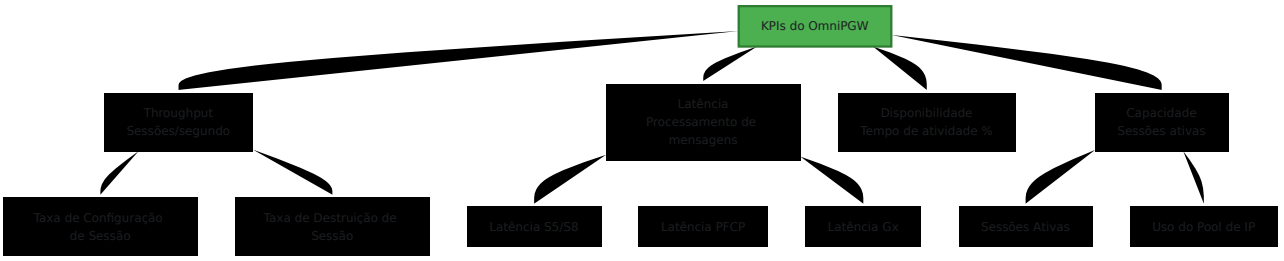
```
          description: "Nenhum UPF saudável disponível para
criação de sessão"

      # Alertas de Latência
      - alert: OmniPGW_HighLatency
        expr: |
          histogram_quantile(0.95,
            rate(s5s8_inbound_handling_duration_bucket[5m])
          ) > 100000
        for: 5m
        labels:
          severity: warning
        annotations:
          summary: "Alta latência de mensagem no OmniPGW"
          description: "latência p95 {{ $value }}µs (> 100ms)"

      # Alertas do Sistema
      - alert: OmniPGW_HighMemoryUsage
        expr: vm_memory_total > 2000000000
        for: 10m
        labels:
          severity: warning
        annotations:
          summary: "Alto uso de memória no OmniPGW"
          description: "VM usando {{ $value | humanize }}B de
memória"

      - alert: OmniPGW_HighProcessCount
        expr: vm_system_process_count > 100000
        for: 10m
        labels:
          severity: warning
        annotations:
          summary: "Alta contagem de processos no OmniPGW"
          description: "{{ $value }} processos Erlang (vazamento
potencial)"
```

Configuração do AlertManager

```yaml
# alertmanager.yml
global:
  resolve_timeout: 5m

route:
  receiver: 'ops-team'
  group_by: ['alertname', 'instance']
  group_wait: 10s
  group_interval: 10s
  repeat_interval: 12h

  routes:
    - match:
        severity: critical
      receiver: 'pagerduty'

    - match:
        severity: warning
      receiver: 'slack'

receivers:
  - name: 'ops-team'
    email_configs:
      - to: 'ops@example.com'

  - name: 'slack'
    slack_configs:
      - api_url:
'https://hooks.slack.com/services/YOUR/SLACK/WEBHOOK'
        channel: '#omnipgw-alerts'
        title: 'Alerta OmniPGW: {{ .GroupLabels.alertname }}'
        text: '{{ range .Alerts }}{{ .Annotations.description }}{{
end }}'

  - name: 'pagerduty'
    pagerduty_configs:
      - service_key: 'YOUR_PAGERDUTY_KEY'
```

Monitoramento de Desempenho

Indicadores-Chave de Desempenho (KPIs)



Consultas de Throughput

Taxa de Configuração de Sessão:

```
rate(s5s8_inbound_messages_total{message_type="create_session_request
[5m])
```

Taxa de Destruição de Sessão:

```
rate(s5s8_inbound_messages_total{message_type="delete_session_request
[5m])
```

Crescimento Líquido de Sessões:

```
rate(s5s8_inbound_messages_total{message_type="create_session_request
[5m]) -
rate(s5s8_inbound_messages_total{message_type="delete_session_request
[5m])
```

Análise de Latência

Latência de Processamento de Mensagens (Percentis):

```
# p50 (Mediana)
histogram_quantile(0.50,
  rate(s5s8_inbound_handling_duration_bucket[5m])
)

# p95
histogram_quantile(0.95,
  rate(s5s8_inbound_handling_duration_bucket[5m])
)

# p99
histogram_quantile(0.99,
  rate(s5s8_inbound_handling_duration_bucket[5m])
)
```

Divisão de Latência por Tipo de Mensagem:

```
histogram_quantile(0.95,
  rate(s5s8_inbound_handling_duration_bucket[5m])
) by (request_message_type)
```

Tendência de Capacidade

Tendência de Crescimento de Sessões (24h):

```
teid_registry_count -
teid_registry_count offset 24h
```

Capacidade Restante:

```
# Para capacidade máxima de 10.000 sessões
10000 - teid_registry_count
```

Tempo até o Esgotamento da Capacidade:

```
# Dias até a capacidade esgotada (com base na taxa de crescimento
de 1h)
(10000 - teid_registry_count) /
(rate(teid_registry_count[1h]) * 86400)
```

Métricas de Solução de Problemas

Identificando Problemas

Problema: Alta Taxa de Rejeição de Sessões

Consulta:

```
rate(s5s8_inbound_errors_total[5m]) by (message_type)
```

Ação:

- Verifique os logs de erro
- Verifique a conectividade do PCRF (erros Gx)
- Verifique o esgotamento do pool de IP

Problema: Configuração de Sessão Lenta

Consulta:

```
histogram_quantile(0.95,

rate(s5s8_inbound_handling_duration_bucket{request_message_type="crea
[5m])
)
```

Ação:

- Verifique a latência Gx (tempo de resposta do PCRF)
- Verifique a latência PFCP (tempo de resposta do PGW-U)

- Revise o uso de recursos do sistema

Problema: Falhas de Política do PCRF

Consultas:

```
# Taxa geral de falha de resposta Gx
sum(rate(gx_outbound_responses_total{result_code_class!="2xxx"}
[5m])) /
sum(rate(gx_outbound_responses_total[5m])) * 100

# Divisão por host PCRF
sum(rate(gx_outbound_responses_total[5m])) by (diameter_host,
result_code_class)

# Classes específicas de código de resultado
rate(gx_outbound_responses_total{result_code_class="5xxx"}[5m]) by
(diameter_host)
```

Ação:

- Verifique a conectividade e a saúde do PCRF
- Revise os perfis de assinantes no PCRF (erros 5xxx geralmente indicam problemas de política)
- Verifique a configuração do par Diameter
- Verifique os logs do PCRF para erros correspondentes
- Para 5012 (DIAMETER_UNABLE_TO_COMPLY), revise o tratamento de Re-Auth-Request

Problema: Vazamento de Memória Suspeito

Consultas:

```
# Tendência de memória total
rate(vm_memory_total[1h])

# Tendência de memória do processo
rate(vm_memory_processes[1h])

# Tendência de contagem de processos
rate(vm_system_process_count[1h])
```

Ação:

- Verifique se há sessões obsoletas
- Revise as contagens de registro
- Reinicie se o vazamento for confirmado

Consultas de Depuração

Encontrar o Horário de Pico de Sessão:

```
max_over_time(teid_registry_count[24h])
```

Comparar Atual vs. Histórico:

```
teid_registry_count /
avg_over_time(teid_registry_count[7d])
```

Identificar Anomalias:

```
abs(
  teid_registry_count -
  avg_over_time(teid_registry_count[1h])
) > 100
```

Melhores Práticas

Coleta de Métricas

1. **Intervalo de Scrape:** 15-30 segundos (equilibrar granularidade vs. carga)
2. **Retenção:** 15+ dias para análise histórica
3. **Rótulos:** Use rotulagem consistente (instância, ambiente, site)

Design de Painéis

1. **Painel de Visão Geral** - KPIs de alto nível para NOC
2. **Painéis Detalhados** - Análise profunda por interface
3. **Painel de Solução de Problemas** - Métricas de erro e logs

Design de Alertas

1. **Evitar Fadiga de Alertas** - Alertar apenas sobre problemas acionáveis
2. **Escalonamento** - Aviso → Crítico com aumento de severidade
3. **Contexto** - Incluir links de runbook nas descrições de alertas

---

Documentação Relacionada

Configuração e Configuração

- **Guia de Configuração** - Configuração de métricas do Prometheus, configuração da API REST OAM
- **Guia de Solução de Problemas** - Usando métricas para depuração

Métricas de Interface

- **Interface PFCP** - Métricas de sessão PFCP, monitoramento de saúde do UPF

- **Interface Diameter Gx** - Métricas de política Gx, rastreamento de interação do PCRF
- **Interface Diameter Gy** - Métricas de cobrança Gy, rastreamento de cotas, timeouts do OCS
- **Interface S5/S8** - Métricas de mensagem GTP-C, comunicação SGW-C

Monitoramento Especializado

- **Monitoramento P-CSCF** - Métricas de descoberta do P-CSCF, saúde do IMS
- **Gerenciamento de Sessões** - Sessões ativas, métricas do ciclo de vida da sessão
- **Alocação de IP do UE** - Métricas de utilização do pool de IP

Voltar ao Guia de Operações

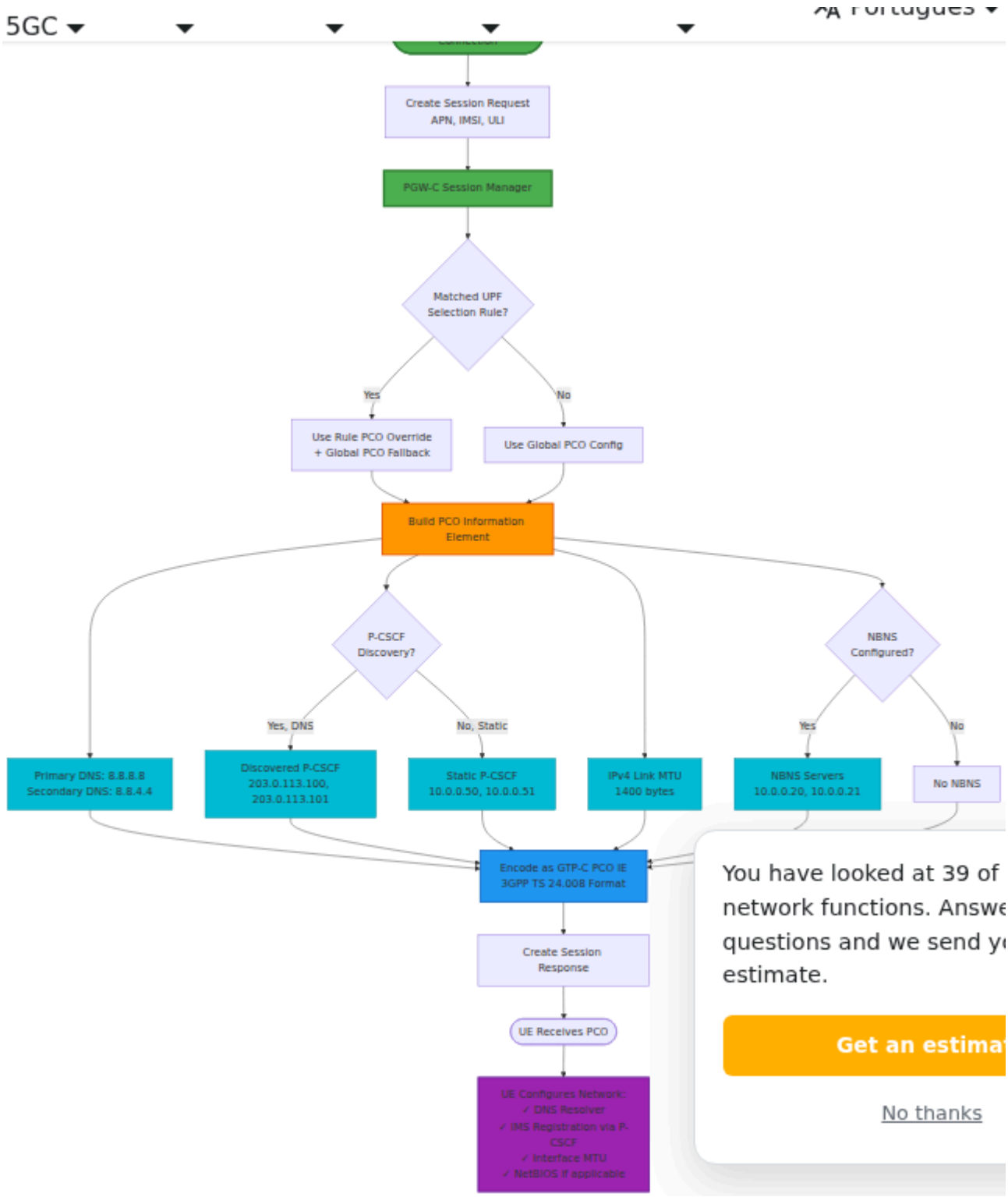
Opções de Configuração de Protocolo (PCO)

Parâmetros de Rede Entregues ao UE

OmniPGW by Omnitouch Network Services

Visão Geral

PCO (Opções de Configuração de Protocolo) são parâmetros de rede enviados ao UE (dispositivo móvel) durante o estabelecimento da conexão PDN. Esses parâmetros permitem que o UE acesse serviços de rede como DNS, IMS e configure as configurações de rede.



You have looked at 39 of
network functions. Answe
questions and we send y
estimate.
Get an estima
No thanks

Contêineres PCO que o OmniPGW emite (rede → MS):

Nome IE	ID do Contêiner	Descrição	Necessário
Endereço IPv4 do Servidor DNS	0x000D	DNS Primário	Sim
Endereço IPv4 do Servidor DNS	0x000D	DNS Secundário	Opcional
Endereço IPv6 do Servidor DNS	0x0003	DNS Primário/Secundário (IPv6)	Opcional
Endereço IPv4 do P-CSCF	0x000C	P-CSCF para IMS (um contêiner por endereço)	Opcional (IMS)
Endereço IPv6 do P-CSCF	0x0001	P-CSCF para IMS (um contêiner por endereço)	Opcional (IMS)
MTU do Link IPv4	0x0010	Unidade máxima de transmissão	Recomendado

> **DNS/NBNS via IPCP.** Quando o UE solicita servidores DNS (e NBNS/WINS) usando um **IPCP** (Protocolo de Controle de Internet, contêiner `0x8021`) Configure-Request em vez de um contêiner DNS simples, o OmniPGW responde dentro de um IPCP **Configure-Nak** carregando os endereços configurados como as opções IPCP `ms_dns1`/`ms_dns2` e `ms_wins1`/`ms_wins2` (RFC 1877). NBNS/WINS é, portanto, entregue **apenas** através deste caminho IPCP — não há contêiner PCO NBNS autônomo (`0x0011`) emitido. Um Configure-Nak (não um Ack) é usado porque o PGW fornece endereços que diferem dos espaços reservados `0.0.0.0` do UE, e a resposta ecoa o Identificador IPCP da solicitação (RFC 1661 §5).

> **P-CSCF / DNS IPv4 e IPv6; sem E-CSCF.** O OmniPGW responde tanto a solicitações de endereços P-CSCF IPv4 quanto IPv6, roteando cada

> endereço descoberto ou configurado para o contêiner correspondente à sua família: endereços IPv4 vão em `0x000C`, endereços IPv6 vão em `0x0001`. Uma solicitação de P-CSCF IPv6 (`p_cscf_ipv6_address_request`) é atendida a partir de `p_cscf_ipv6_address_list` (ou os endereços IPv6 retornados pela descoberta DNS/AAAA), e uma solicitação de DNS IPv6 (`dns_server_ipv6_address_request`) é atendida no contêiner `0x0003` a partir de `primary_dns_server_ipv6_address` / `secondary_dns_server_ipv6_address`. Famílias de endereços nunca são misturadas entre contêineres. Não há manipulação de contêiner E-CSCF. Tudo o que foi mencionado acima se aplica igualmente ao APCO (PCO Adicional).

Configuração

Configuração Básica

```
# config/runtime.exs
config :pgw_c,
  pco: %{
    # Servidores DNS (obrigatório)
    primary_dns_server_address: "8.8.8.8",
    secondary_dns_server_address: "8.8.4.4",

    # Servidores DNS IPv6 (opcional, contêiner 0x0003)
    primary_dns_server_ipv6_address: nil,
    secondary_dns_server_ipv6_address: nil,

    # Servidores NBNS (opcional, para dispositivos Windows)
    primary_nbns_server_address: nil,
    secondary_nbns_server_address: nil,

    # Endereços P-CSCF para IMS/VoLTE (opcional)
    p_cscf_ipv4_address_list: [],   # contêiner 0x000C
    p_cscf_ipv6_address_list: [],   # contêiner 0x0001

    # Descoberta Dinâmica do P-CSCF (opcional; resolve registros A
e AAAA)
    p_cscf_discovery_enabled: false,
    p_cscf_discovery_dns_server: nil,
    p_cscf_discovery_timeout_ms: 5000,

    # Tamanho do MTU IPv4 (bytes)
    ipv4_link_mtu_size: 1400
  }
```

Parâmetros PCO

Endereços do Servidor DNS

DNS Primário e Secundário:

```
pco: %{
  primary_dns_server_address: "8.8.8.8",
  secondary_dns_server_address: "8.8.4.4"
}
```

Provedores de DNS Comuns:

Provedor	Primário	Secundário
Google	8.8.8.8	8.8.4.4
Cloudflare	1.1.1.1	1.0.0.1
Quad9	9.9.9.9	149.112.112.112
OpenDNS	208.67.222.222	208.67.220.220

DNS Privado:

```
pco: %{
  primary_dns_server_address: "10.0.0.10",
  secondary_dns_server_address: "10.0.0.11"
}
```

Endereços P-CSCF (IMS)

Para Serviços IMS/VoLTE:

```
pco: %{
  p_cscf_ipv4_address_list: [
    "10.0.0.50",  # P-CSCF Primário (IPv4, contêiner 0x000C)
    "10.0.0.51"   # P-CSCF Secundário
  ],
  # Endereços P-CSCF IPv6 (contêiner 0x0001) para PDNs IMS
IPv6/IPv4v6
  p_cscf_ipv6_address_list: [
    "2001:db8::50",
    "2001:db8::51"
  ]
}
```

Cada endereço é emitido apenas no contêiner correspondente à sua família, e apenas quando o UE solicita essa família — uma solicitação de P-CSCF IPv4 recebe os contêineres `0x000C`, uma solicitação de P-CSCF IPv6 recebe os contêineres `0x0001`. PDNs IMS de pilha dupla normalmente solicitam ambos.

P-CSCF (Função de Controle de Sessão de Chamada Proxy):

- Ponto de entrada para sinalização IMS
- Necessário para VoLTE, VoWiFi, RCS
- UE usa SIP através deste servidor

Descoberta Dinâmica do P-CSCF

Descoberta do P-CSCF Baseada em DNS:

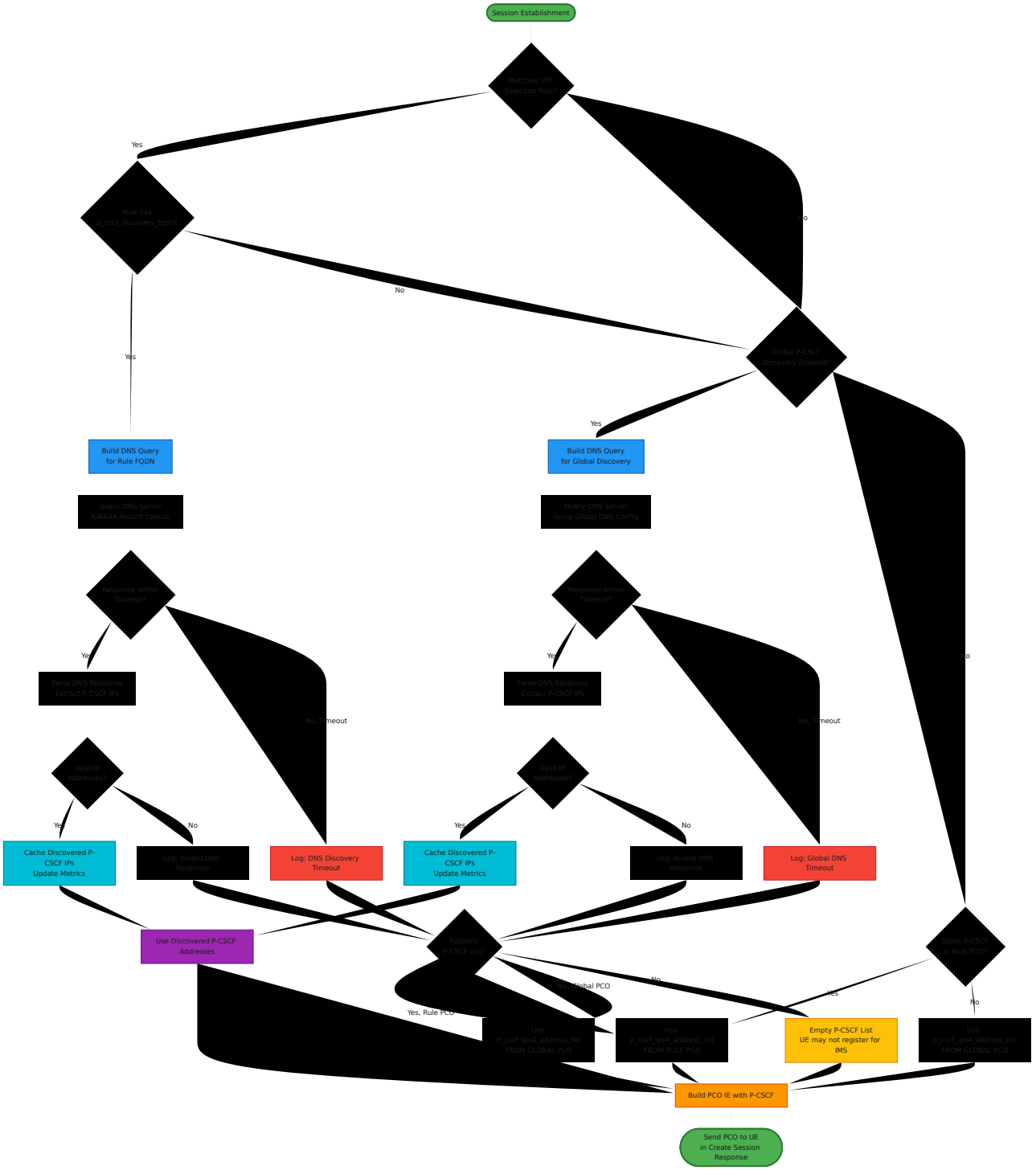
O OmniPGW suporta a descoberta dinâmica do P-CSCF através de consultas DNS conforme definido na 3GPP TS 23.003 e TS 24.229. Quando habilitado, o PGW-C pode consultar o DNS para endereços P-CSCF em vez de usar configuração estática.

```
pco: %{
  # Habilitar descoberta dinâmica do P-CSCF
  p_cscf_discovery_enabled: true,

  # Servidor DNS para consultas P-CSCF (como tupla)
  p_cscf_discovery_dns_server: {10, 179, 2, 177},

  # Tempo limite para consultas DNS (milissegundos)
  p_cscf_discovery_timeout_ms: 5000,

  # Lista estática de P-CSCF (usada como fallback se o DNS falhar)
  p_cscf_ipv4_address_list: ["10.0.0.50"]
}
```

Como Funciona:

1. Quando `p_cscf_discovery_enabled: true`, o PGW-C realiza consultas DNS para endereços P-CSCF
2. A consulta DNS é enviada para o `p_cscf_discovery_dns_server` configurado
3. Se a consulta DNS for bem-sucedida, os endereços P-CSCF descobertos são enviados ao UE via PCO
4. Se a consulta DNS falhar ou expirar, recorre à lista estática `p_cscf_ipv4_address_list`
5. Veja Monitoramento do P-CSCF para monitoramento e métricas detalhadas

Fluxo de Descoberta do P-CSCF



Prioridade de Descoberta:

1. **Descoberta FQDN por Regra** (Maior Prioridade) - `p_cscf_discovery_fqdn` na regra de seleção do UPF
2. **Descoberta DNS Global** - `p_cscf_discovery_enabled: true` na configuração global do PCO

3. **Lista Estática PCO por Regra** - `p_cscf_ipv4_address_list` na substituição do PCO por regra
4. **Lista Estática PCO Global** (Fallback) - `p_cscf_ipv4_address_list` na configuração global do PCO

Monitoramento:

Todas as tentativas de descoberta do P-CSCF são registradas e rastreadas com métricas:

- Taxas de sucesso/falha de consultas DNS
- Latência de descoberta
- Estatísticas de uso de fallback
- Métricas de descoberta por regra e global

Veja Monitoramento do P-CSCF para detalhes completos de monitoramento.

Opções de Configuração:

Parâmetro	Tipo	Padrão	Descrição
`p_cscf_discovery_enabled`	Booleano	`false`	Habilitar descoberta dinâmica do P-CSCF baseada em DNS
`p_cscf_discovery_dns_server`	Tupla (IP)	`nil`	Endereço IP do servidor DNS como tupla 4 (por exemplo, `{10, 179, 2, 177}`)
`p_cscf_discovery_timeout_ms`	Inteiro	`5000`	Tempo limite para consultas DNS em milissegundos

Casos de Uso:

- **Implantações IMS dinâmicas** - Endereços P-CSCF mudam com base na configuração do DNS
- **Balanceamento de carga geográfico** - DNS retorna os servidores P-CSCF mais próximos
- **Alta disponibilidade** - DNS retorna automaticamente servidores P-CSCF disponíveis
- **Ambientes multi-inquilinos** - Diferentes assinantes recebem diferentes servidores P-CSCF

Exemplo: IMS de Produção com Descoberta DNS

```
pco: %{
  primary_dns_server_address: "10.0.0.10",
  secondary_dns_server_address: "10.0.0.11",

  # Habilitar descoberta dinâmica do P-CSCF
  p_cscf_discovery_enabled: true,
  p_cscf_discovery_dns_server: {10, 179, 2, 177},  # Servidor DNS
IMS
  p_cscf_discovery_timeout_ms: 3000,

  # Endereços P-CSCF de fallback (se o DNS falhar)
  p_cscf_ipv4_address_list: [
    "10.0.0.50",  # Fallback primário
    "10.0.0.51"   # Fallback secundário
  ],

  ipv4_link_mtu_size: 1400
}
```

Descoberta P-CSCF por Regra:

A descoberta do P-CSCF também pode ser configurada por regra de seleção do UPF. Isso permite que diferentes APNs usem diferentes servidores DNS para descoberta do P-CSCF:

```
# Na configuração de seleção do upf
rules: [
  %{
    name: "IMS Traffic",
    priority: 20,
    match_field: :apn,
    match_regex: "^ims",
    upf_pool: [...],

    # Descoberta do P-CSCF por regra
    p_cscf_discovery_fqdn: "pcscf.mnc001.mcc001.3gppnetwork.org"
  }
]
```

Veja Configuração de Seleção do UPF para detalhes sobre a descoberta do P-CSCF por regra.

Veja também: Monitoramento do P-CSCF para monitoramento da descoberta do P-CSCF e saúde

Servidores NBNS (NetBIOS)

Para Compatibilidade com Dispositivos Windows:

```
pco: %{
  primary_nbns_server_address: "10.0.0.20",
  secondary_nbns_server_address: "10.0.0.21"
}
```

Quando Usar:

- Redes empresariais com dispositivos Windows
- Suporte a aplicativos legados
- Resolução de nomes NetBIOS necessária

Tamanho do MTU do Link

Unidade Máxima de Transmissão:

```
pco: %{
  ipv4_link_mtu_size: 1400  # bytes
}
```

Valores Comuns de MTU:

MTU	Caso de Uso
1500	Ethernet padrão (sem tunelamento)
1400	Sobrecarga de tunelamento GTP contabilizada
1420	Sobrecarga reduzida
1280	MTU mínima do IPv6
1360	Ambientes VPN/túnel

Recomendação: Use **1400** para LTE para contabilizar a sobrecarga do GTP-U.

Exemplos de Configuração

APN de Internet

```
pco: %{
  primary_dns_server_address: "8.8.8.8",
  secondary_dns_server_address: "8.8.4.4",
  ipv4_link_mtu_size: 1400
}
```

APN de IMS

```
pco: %{
  primary_dns_server_address: "10.0.0.10",
  secondary_dns_server_address: "10.0.0.11",
  p_cscf_ipv4_address_list: [
    "10.0.0.50",
    "10.0.0.51"
  ],
  ipv4_link_mtu_size: 1400
}
```

Veja: Monitoramento do P-CSCF para monitoramento das taxas de sucesso de registro IMS e saúde do P-CSCF

APN Empresarial

```
pco: %{
  primary_dns_server_address: "10.100.0.10",
  secondary_dns_server_address: "10.100.0.11",
  primary_nbns_server_address: "10.100.0.20",
  secondary_nbns_server_address: "10.100.0.21",
  ipv4_link_mtu_size: 1400
}
```

PCO em Mensagens GTP-C

Resposta de Criação de Sessão

O OmniPGW inclui PCO na mensagem **Resposta de Criação de Sessão**:

```
Create Session Response
├── Cause: Request accepted
├── UE IP Address: 100.64.1.42
├── PCO (Protocol Configuration Options)
│   ├── DNS Server IPv4 Address: 8.8.8.8
│   ├── DNS Server IPv4 Address: 8.8.4.4
│   ├── P-CSCF IPv4 Address: 10.0.0.50
│   ├── P-CSCF IPv4 Address: 10.0.0.51
│   └── IPv4 Link MTU: 1400
```

Processamento do UE

O UE recebe PCO e:

1. Configura o resolvedor DNS com os servidores fornecidos
2. Registra-se no P-CSCF para serviços IMS
3. Define o MTU da interface para o valor especificado

Resolução de Problemas

Problema: UE Não Consegue Resolver DNS

Sintomas:

- UE tem endereço IP, mas não consegue acessar a internet
- Consultas DNS falham

Possíveis Causas:

1. Endereços de servidor DNS incorretos na configuração do PCO
2. Servidores DNS não acessíveis a partir do pool de IP do UE
3. Firewall bloqueando o tráfego DNS

Resolução:

```
# Testar a acessibilidade do servidor DNS
ping 8.8.8.8

# Testar a resolução DNS a partir da rede do UE
nslookup google.com 8.8.8.8

# Verificar a configuração do PCO
grep "primary_dns_server_address" config/runtime.exs
```

Problema: Registro IMS Falha

Sintomas:

- Chamadas VoLTE falham
- UE mostra "Sem registro IMS"

Possíveis Causas:

1. Configuração do P-CSCF ausente
2. Endereços IP do P-CSCF incorretos
3. P-CSCF não acessível

Resolução:

```
# Verificar a configuração do P-CSCF
pco: %{
  p_cscf_ipv4_address_list: ["10.0.0.50"]  # Garantir que não
esteja vazio
}
```

Problema: Problemas de MTU

Sintomas:

- Alguns sites carregam, outros não
- Transferências de arquivos grandes falham
- Problemas de fragmentação

Possíveis Causas:

- MTU muito grande para a sobrecarga de tunelamento
- MTU muito pequeno causando fragmentação excessiva

Resolução:

```
# Recomendado: 1400 para tunelamento GTP
pco: %{
  ipv4_link_mtu_size: 1400
}

# Se ainda tiver problemas, tente um valor menor
pco: %{
  ipv4_link_mtu_size: 1360
}
```

Melhores Práticas

Configuração de DNS

1. **Use Servidores DNS Confiáveis**
    - Público: Google (8.8.8.8), Cloudflare (1.1.1.1)
    - Privado: DNS interno para empresas
2. **Sempre Configure o Secundário**
    - Fornece redundância
    - Melhora a confiabilidade
3. **Considere a Segurança do DNS**
    - Resolvedores compatíveis com DNSSEC
    - Filtragem de DNS para segurança

Configuração de IMS

1. **Forneça Múltiplos P-CSCF**
    - Pelo menos 2 para redundância
    - Distribuição geográfica, se possível
2. **Garanta Acessibilidade**
    - O P-CSCF deve ser acessível a partir do pool de IP do UE
    - Testar conectividade SIP

Otimização de MTU

1. **Contabilize a Sobrecarga**
    - GTP-U: 36 bytes (IPv4)
    - IPsec: Variável (50-100 bytes)
2. **MTU Padrão para LTE**
    - Recomendado: **1400 bytes**
    - Equilibra throughput e compatibilidade
3. **Teste de Ponta a Ponta**
    - Descoberta de MTU de caminho
    - Teste com pacotes grandes

Documentação Relacionada

Guias de Configuração

- **Guia de Configuração** - Referência completa do runtime.exs, seleção do UPF com substituições de PCO
- **Alocação de IP do UE** - Gerenciamento de pool de IP, alocação baseada em APN

- **Monitoramento do P-CSCF** - Monitoramento da descoberta do P-CSCF, rastreamento de saúde, métricas

Gerenciamento de Sessão e Interface

- **Gerenciamento de Sessão** - Ciclo de vida da sessão PDN, estabelecimento de bearer
- **Interface S5/S8** - Protocolo GTP-C, codificação e entrega do PCO
- **Interface PFCP** - Estabelecimento de sessão do plano do usuário

IMS e VoLTE

- **Interface Diameter Gx** - Controle de políticas para bearers IMS
- **Guia de Monitoramento** - Métricas e painéis relacionados ao PCO

Voltar ao Guia de Operações

Descoberta e Monitoramento do P-CSCF

Descoberta Dinâmica do Servidor P-CSCF com Monitoramento em Tempo Real

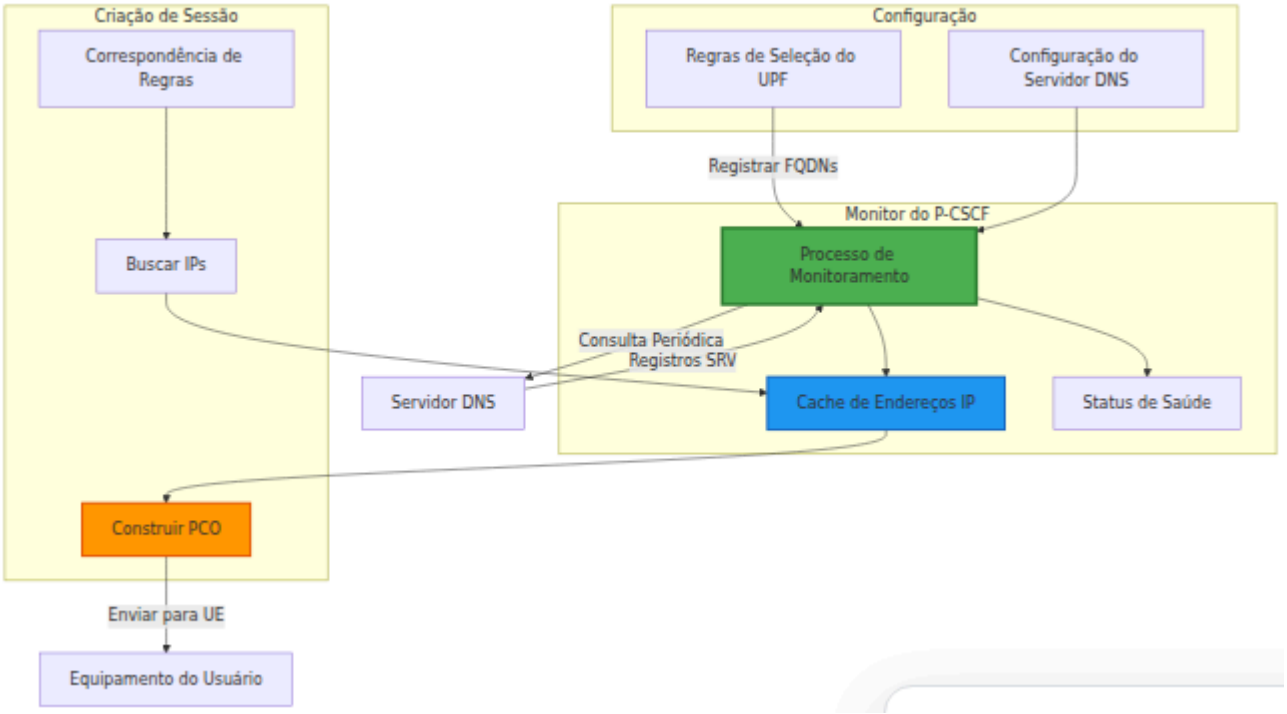
OmniPGW da Omnitouch Network Services

Visão Geral

Descoberta e Monitoramento do P-CSCF (Função de Controle de Sessão de Chamada Proxy) fornece descoberta dinâmica de servidores IMS P-CSCF usando consultas DNS SRV com verificação de saúde SIP OPTIONS em tempo real. Este recurso permite:

- **Descoberta do P-CSCF por Regra**: Diferentes servidores P-CSCF para diferentes tipos de tráfego
- **Monitoramento Automático**: Processo em segundo plano monitora continuamente a resolução DNS (a cada 60 segundos)
- **Verificações de Saúde SIP OPTIONS**: Verifica se os servidores P-CSCF estão ativos por meio de pings SIP OPTIONS
  - **TCP Primeiro**: Tenta SIP OPTIONS via TCP (preferido por confiabilidade)
  - **Fallback para UDP**: Retorna para UDP se o TCP falhar
  - **Rastreamento de Status**: Marca cada servidor como :up ou :down com base na resposta
- **Rastreamento de Saúde em Tempo Real**: A API OAM expõe o status de resolução, IPs descobertos e status de saúde
- **Fallback Graceful**: Estratégia de fallback em três camadas para máxima confiabilidade

- **Métricas Prometheus**: Total observabilidade por meio de métricas Prometheus



Índice

1. Introdução Rápida
2. Configuração
3. Como Funciona
4. Monitoramento da API OAM
5. Métricas e Observabilidade
6. Estratégia de Fallback
7. Configuração do DNS
8. Solução de Problemas
9. Melhores Práticas

Introdução Rápida

Configuração Básica

```
# config/runtime.exs

# Configuração global do PCO (servidor DNS para descoberta do P-
CSCF)
config :pgw_c,
  pco: %{
    p_cscf_discovery_dns_server: "192.0.2.177",
    p_cscf_discovery_enabled: true,
    p_cscf_discovery_timeout_ms: 5000
  },

  upf_selection: %{
    rules: [
      # Tráfego IMS - Descoberta dinâmica do P-CSCF
      %{
        name: "Tráfego IMS",
        priority: 20,
        match_field: :apn,
        match_regex: "^ims",
        upf_pool: [
          %{remote_ip_address: "10.100.2.21", remote_port: 8805,
weight: 80}
        ],
        # FQDN de Descoberta do P-CSCF (veja o Guia de
Configuração para mais regras de seleção do UPF)
        p_cscf_discovery_fqdn:
"pcscf.mnc001.mcc001.3gppnetwork.org",
        # Fallback estático (veja o Guia de Configuração do PCO)
        pco: %{
          p_cscf_ipv4_address_list: ["203.0.113.100",
"203.0.113.101"]
        }
      }
    ]
  }
```

Veja o Guia de Configuração para configuração completa das regras de seleção do UPF e Configuração do PCO para opções de fallback estático do P-CSCF.

Monitoramento de Acesso

1. Inicie o OmniPGW
2. Consulte a API OAM: `curl -k https://localhost:8443/api/pcscf_monitor`
3. Inspecione o status de resolução retornado e os IPs descobertos (execute novamente a solicitação para o estado atual)

Configuração

Configurações Globais de Descoberta do P-CSCF

Configure o servidor DNS usado para a descoberta do P-CSCF na seção PCO:

```
pco: %{
  # Servidor DNS para descoberta do P-CSCF (separado do DNS
fornecido ao UE)
  p_cscf_discovery_dns_server: "192.0.2.177",

  # Habilitar recurso de descoberta DNS do P-CSCF
  p_cscf_discovery_enabled: true,

  # Timeout para consultas DNS SRV (milissegundos)
  p_cscf_discovery_timeout_ms: 5000,

  # Endereços P-CSCF estáticos (fallback global)
  p_cscf_ipv4_address_list: ["203.0.113.146"]
}
```

FQDNs do P-CSCF por Regra

Cada regra de seleção do UPF pode especificar seu próprio FQDN de descoberta do P-CSCF:

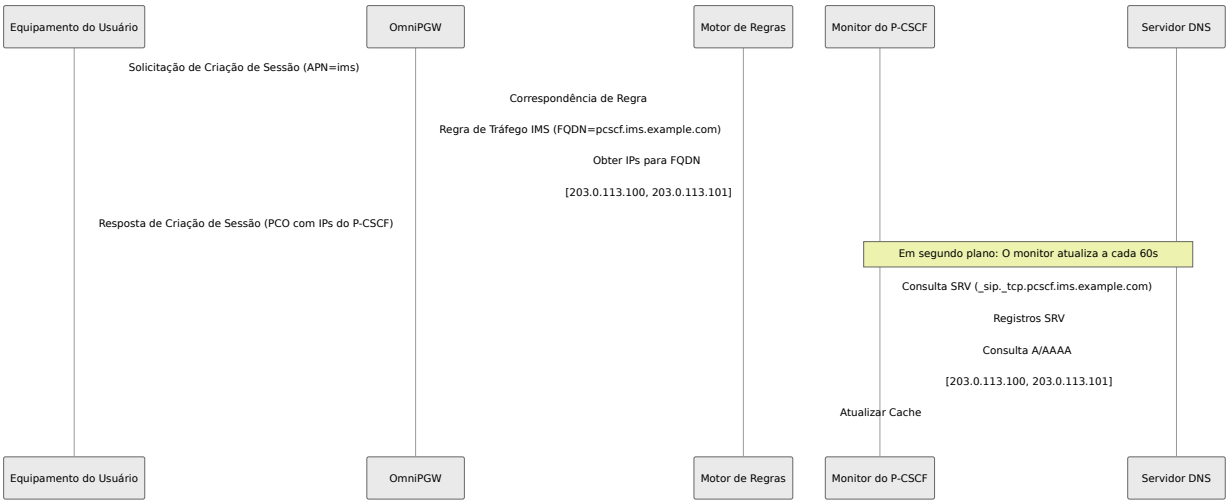
```
upf_selection: %{
  rules: [
    # Tráfego IMS - P-CSCF específico para IMS
    %{
      name: "Tráfego IMS",
      match_field: :apn,
      match_regex: "^ims",
      upf_pool: [...],
      p_cscf_discovery_fqdn:
"pcscf.ims.mnc001.mcc001.3gppnetwork.org",
      pco: %{
        p_cscf_ipv4_address_list: ["203.0.113.100"]  # Fallback
      }
    },

    # Empresa - P-CSCF específico para Empresa
    %{
      name: "Tráfego Empresarial",
      match_field: :apn,
      match_regex: "^enterprise",
      upf_pool: [...],
      p_cscf_discovery_fqdn: "pcscf.enterprise.example.com",
      pco: %{
        p_cscf_ipv4_address_list: ["192.168.1.50"]  # Fallback
      }
    },

    # Internet - Sem descoberta do P-CSCF (usa configuração
global)
    %{
      name: "Tráfego da Internet",
      match_field: :apn,
      match_regex: "^internet",
      upf_pool: [...]
      # Sem p_cscf_discovery_fqdn - usa configuração global do PCO
    }
  ]
}
```

Como Funciona

Processo de Inicialização

1. **Aplicação Inicia**
    - O GenServer do Monitor do P-CSCF é inicializado
    - O analisador de configuração extrai todos os FQDNs únicos do P-CSCF das regras de seleção do UPF
2. **Registro de FQDN**
    - Cada FQDN único é registrado com o monitor
    - O monitor realiza a consulta inicial DNS SRV para cada FQDN
    - **Verificação de Saúde SIP OPTIONS** (em paralelo para todos os servidores descobertos):
        - Tente TCP primeiro (`SIP/2.0/TCP` na porta 5060)
        - Se o TCP falhar, retorne para UDP (`SIP/2.0/UDP` na porta 5060)
        - Marque cada servidor como `:up` (responde) ou `:down` (sem resposta/timeout)
    - Resultados (IPs, status de saúde ou erros) são armazenados em cache com timestamps
3. **Monitoramento Periódico** (A cada 60 segundos)
    - O monitor atualiza todos os FQDNs
    - Consultas DNS são executadas em segundo plano sem bloquear
    - Para cada servidor descoberto:
        - Envie SIP OPTIONS via TCP (timeout: 5 segundos)
        - Se o TCP falhar, tente UDP (timeout: 5 segundos)
        - Atualize o status de saúde com base na resposta
    - O cache é atualizado com os últimos resultados DNS e status de saúde

Fluxo de Criação de Sessão



Processo de Consulta DNS

O monitor usa **registros DNS SRV** para descoberta direta do P-CSCF:

1. **Consulta SRV**: Consulta registros SRV em `_sip._tcp.{fqdn}`
2. **Ordenação por Prioridade**: Ordena por prioridade e peso
3. **Extração de Alvo**: Extrai nomes de host de destino dos registros SRV
4. **Resolução de Nome de Host**: Resolve nomes de host de destino para endereços IP (registros A/AAAA)
5. **Cache**: Armazena IPs resolvidos com status e timestamp

Precedência na Seleção de Endereço do P-CSCF

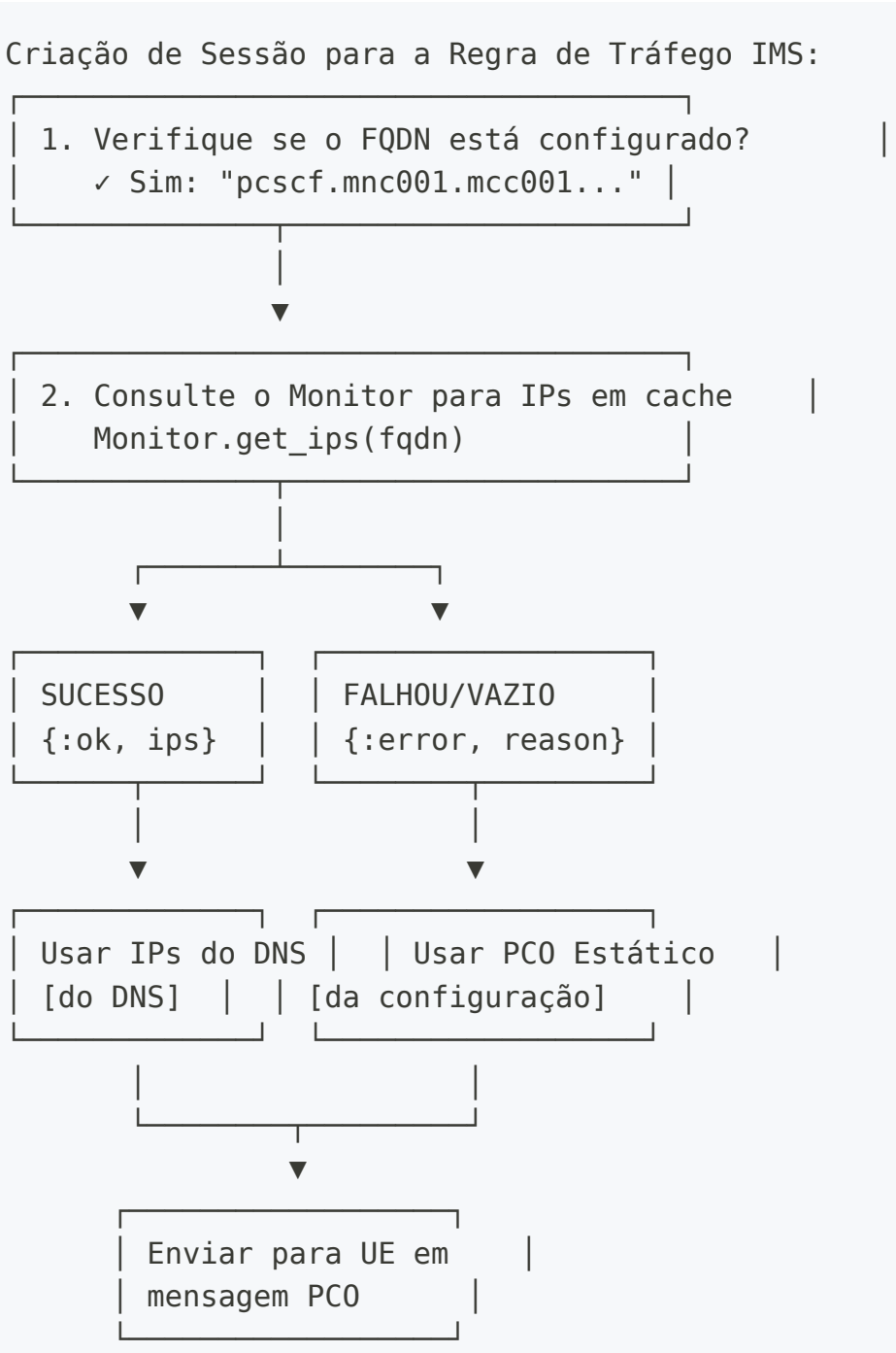
Quando tanto o FQDN quanto o PCO estático estão configurados em uma regra, o FQDN tem precedência:

```
%{
  name: "Tráfego IMS",
  p_cscf_discovery_fqdn: "pcscf.mnc001.mcc001.3gppnetwork.org",  #
← Tentado PRIMEIRO
  pco: %{
    p_cscf_ipv4_address_list: ["203.0.113.100", "203.0.113.101"]
# ← Fallback
  }
}
```

Lógica de Seleção:

Condição	Fonte do P-CSCF	IPs Usados	Mensage
FQDN resolve com sucesso	Descoberta DNS (Monitor)	IPs descobertos do DNS	`"Usando e do P-CSCF pcscf.exa`
FQDN falha ao resolver	Sobrescrição do PCO da Regra	IPs estáticos de `pco.p_cscf_ipv4_address_list`	`"Falha ac IPs do P- FQDN..., retornand configura estática"`
FQDN retorna lista vazia	Sobrescrição do PCO da Regra	IPs estáticos de `pco.p_cscf_ipv4_address_list`	Fallback ac
Monitor indisponível	Sobrescrição do PCO da Regra	IPs estáticos de `pco.p_cscf_ipv4_address_list`	Erro acion
Nenhum FQDN configurado	Sobrescrição do PCO da Regra ou Global	IPs estáticos da regra ou configuração global	Usa config estática di

Fluxo de Exemplo:



Cenários do Mundo Real:

Cenário 1: Descoberta DNS Funciona

```
Config:
  p_cscf_discovery_fqdn: "pcscf.ims.example.com"
  pco.p_cscf_ipv4_address_list: ["203.0.113.100"]

Resultado DNS: [203.0.113.150, 203.0.113.151]
UE Recebe: [203.0.113.150, 203.0.113.151] ← Do DNS
Nota: PCO estático é ignorado quando o DNS tem sucesso
```

Cenário 2: DNS Falha, Fallback Graceful ⚠

```
Config:
  p_cscf_discovery_fqdn: "pcscf.ims.example.com"
  pco.p_cscf_ipv4_address_list: ["203.0.113.100"]

Resultado DNS: ERRO :no_naptr_records
UE Recebe: [203.0.113.100] ← Do PCO estático
Nota: A sessão tem sucesso apesar da falha do DNS
```

Cenário 3: Nenhum FQDN Configurado

```
Config:
  # Nenhum p_cscf_discovery_fqdn
  pco.p_cscf_ipv4_address_list: ["192.168.1.50"]

UE Recebe: [192.168.1.50] ← Do PCO estático
Nota: Descoberta DNS não foi tentada
```

Por que este design?

1. **Preferir Dinâmico**: DNS fornece flexibilidade, balanceamento de carga e roteamento ciente da localização
2. **Garantir Confiabilidade**: Fallback estático garante que as sessões nunca falhem devido a problemas de DNS
3. **Zero Intervenção Manual**: Failover automático sem envolvimento do operador
4. **Seguro para Produção**: O melhor dos dois mundos - agilidade com estabilidade

Recomendação: Sempre configure tanto o FQDN quanto o PCO estático para implantações em produção:

```
# ✓ RECOMENDADO: Dinâmico com fallback
%{
  p_cscf_discovery_fqdn: "pcscf.ims.example.com",  # Preferido
  pco: %{
    p_cscf_ipv4_address_list: ["203.0.113.100"]     # Rede de
segurança
  }
}

# ⚠ ARRISCADO: Apenas dinâmico (retorna para o PCO global)
%{
  p_cscf_discovery_fqdn: "pcscf.ims.example.com"
  # Sem fallback específico para a regra!
}

# ✓ VÁLIDO: Apenas estático (sem sobrecarga de DNS)
%{
  pco: %{
    p_cscf_ipv4_address_list: ["192.168.1.50"]
  }
}
```

Monitoramento da API OAM

Endpoint do Monitor do P-CSCF

O monitoramento é exposto através da API REST OAM (HTTPS/TLS na porta `8443`). Todas as rotas são servidas sob o prefixo `/api`. Consulte o monitor de descoberta do P-CSCF com:

```
GET /api/pcscf_monitor
```

```
curl -k https://localhost:8443/api/pcscf_monitor
```

O endpoint não aceita parâmetros de caminho ou consulta e retorna o estado atual da descoberta em cada chamada. A documentação interativa da API (Swagger UI) está disponível em `https://localhost:8443/api/docs`.

Conteúdos da Resposta:

- **Estatísticas Gerais**
  - Total de FQDNs monitorados
  - FQDNs resolvidos com sucesso
  - Resoluções falhadas
  - Total de IPs do P-CSCF descobertos
- **Detalhes por FQDN**
  - FQDN sendo monitorado
  - Status de resolução (Resolvido / Falhou / Pendente)
  - Número de IPs descobertos
  - Lista de endereços IP resolvidos (com detalhes por servidor)
  - Timestamp da última atualização
  - **Status de Saúde**: Cada servidor descoberto relata:
    - Endereço IP e porta
    - Nome do host (do alvo DNS SRV)
    - Indicador de saúde (Ativo / Inativo)
- **Métricas de Monitoramento**
  - **Total de FQDNs**: Número de FQDNs únicos registrados para monitoramento
  - **Resolvidos com Sucesso**: FQDNs que foram resolvidos com sucesso via DNS
  - **Resoluções DNS Falhadas**: FQDNs que falharam ao resolver
  - **Total de Servidores P-CSCF**: Número total de servidores descobertos em todos os FQDNs

- **Saudáveis (SIP OPTIONS ATIVOS)**: Servidores respondendo a verificações de saúde SIP OPTIONS
- **Não Saudáveis (SIP OPTIONS INATIVOS)**: Servidores não respondendo a SIP OPTIONS
- **Taxa de Sucesso do DNS**: Porcentagem de resoluções DNS bem-sucedidas
- **Intervalo de Verificação de Saúde**: Frequência das verificações de saúde SIP OPTIONS (60s, timeout de 5s)

A resposta fornece visibilidade atual sobre a saúde da resolução DNS e a disponibilidade do servidor P-CSCF via SIP OPTIONS. Para atualizar, basta executar novamente `GET /api/pcscf_monitor` — cada chamada retorna o estado mais recente.

Status de Seleção do UPF

O endpoint de status de seleção do UPF (`GET /api/upf_selection`) relata o status de descoberta do P-CSCF para cada regra:

```
 Tráfego IMS (Prioridade 20)
   Correspondência: Correspondência de APN ^ims
   Pool: UPF-IMS-Primário (10.100.2.21:8805)

    Descoberta do P-CSCF
      FQDN: pcscf.mnc001.mcc001.3gppnetwork.org
      Status: ✓ Resolvido (2 IPs)
      IPs Resolvidos: 203.0.113.100, 203.0.113.101

   ⚙ Sobrescrições do PCO
      DNS Primário: 198.51.100.195
      P-CSCF (fallback estático): 203.0.113.100, 203.0.113.101
```

Métricas e Observabilidade

Métricas Prometheus

O sistema de monitoramento do P-CSCF expõe métricas via Prometheus (porta 42069 por padrão):

Métricas de Gauge

```
# Métricas a nível de FQDN
pcscf_fqdns_total                    # Número total de FQDNs
monitorados
pcscf_fqdns_resolved                 # FQDNs resolvidos com
sucesso (DNS teve sucesso)
pcscf_fqdns_failed                   # Resoluções de FQDN
falhadas (DNS falhou)

# Métricas a nível de servidor (agregado)
pcscf_servers_total                  # Número total de
servidores P-CSCF descobertos via DNS SRV
pcscf_servers_healthy                # Servidores respondendo a
SIP OPTIONS (agregado)
pcscf_servers_unhealthy              # Servidores não
respondendo a SIP OPTIONS (agregado)

# Métricas a nível de servidor (por FQDN com rótulo)
pcscf_servers_healthy{fqdn="..."}    # Servidores saudáveis para
FQDN específico
pcscf_servers_unhealthy{fqdn="..."}  # Servidores não saudáveis
para FQDN específico
```

Detalhes da Verificação de Saúde:

- `healthy`: O servidor respondeu ao ping SIP OPTIONS (TCP ou UDP)
- `unhealthy`: O servidor falhou em responder ao SIP OPTIONS (timeout de 5s por transporte)

Exemplos de Métricas

Métricas de Resolução DNS:

```
# Consultar FQDNs resolvidos com sucesso
pcscf_fqdns_resolved

# Calcular taxa de sucesso do DNS
(pcscf_fqdns_resolved / pcscf_fqdns_total) * 100

# Total de servidores descobertos
pcscf_servers_total
```

Métricas de Saúde SIP OPTIONS:

```
# Total de servidores saudáveis em todos os FQDNs
pcscf_servers_healthy

# Total de servidores não saudáveis
pcscf_servers_unhealthy

# Calcular taxa de sucesso da verificação de saúde
(pcscf_servers_healthy / pcscf_servers_total) * 100

# Servidores saudáveis para um FQDN específico
pcscf_servers_healthy{fqdn="pcscf.mnc001.mcc001.3gppnetwork.org"}

# Alerta quando todos os servidores estão inativos
pcscf_servers_healthy == 0 AND pcscf_servers_total > 0
```

Exemplos de Alertas Prometheus:

```yaml
# Alerta quando todos os servidores P-CSCF estão inativos
- alert: AllPCSCFServersDown
  expr: pcscf_servers_healthy == 0 AND pcscf_servers_total > 0
  for: 5m
  labels:
    severity: critical
  annotations:
    summary: "Todos os servidores P-CSCF estão não saudáveis"
    description: "{{ $value }} servidores saudáveis (0) - todos
falharam nas verificações SIP OPTIONS"

# Alerta quando mais de 50% dos servidores estão inativos
- alert: MajorityPCSCFServersDown
  expr: (pcscf_servers_healthy / pcscf_servers_total) < 0.5
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "A maioria dos servidores P-CSCF estão não saudáveis"
    description: "Apenas {{ $value }}% dos servidores estão
respondendo a SIP OPTIONS"

# Alerta sobre falhas de resolução DNS
- alert: PCSCFDNSResolutionFailed
  expr: pcscf_fqdns_failed > 0
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "Falhas de resolução DNS do P-CSCF"
    description: "{{ $value }} FQDN(s) falhando ao resolver"
```

Registro

O monitor registra eventos-chave:

```
[info] Monitor do P-CSCF iniciado
[info] Registrando 2 FQDNs únicos do P-CSCF para monitoramento:
["pcscf.ims.example.com", "pcscf.enterprise.example.com"]
[info] Monitor do P-CSCF: Registrando FQDN pcscf.ims.example.com
[debug] Monitor do P-CSCF: Resolvido com sucesso
pcscf.ims.example.com para 2 IPs
[warning] Monitor do P-CSCF: Falha ao resolver
pcscf.enterprise.example.com: :nxdomain
[debug] Usando endereços do P-CSCF do FQDN pcscf.ims.example.com:
[{10, 101, 2, 100}, {10, 101, 2, 101}]
```

Estratégia de Fallback

O sistema usa uma **estratégia de fallback em três camadas** para máxima confiabilidade:

Camada 1: Descoberta DNS (Preferida)

```
p_cscf_discovery_fqdn: "pcscf.ims.example.com"
```

- O monitor consulta o DNS e armazena em cache os IPs resolvidos
- A sessão usa os IPs em cache se disponíveis
- **Vantagem**: Dinâmico, balanceado, ciente da localização

Camada 2: PCO Estático Específico da Regra (Fallback)

```
pco: %{
  p_cscf_ipv4_address_list: ["203.0.113.100", "203.0.113.101"]
}
```

- Usado se a descoberta DNS falhar ou retornar nenhum IP
- Configuração estática específica da regra

- **Vantagem**: Fallback específico da regra, previsível

Camada 3: Configuração Global do PCO (Último Recurso)

```
# Configuração global do pco
pco: %{
  p_cscf_ipv4_address_list: ["203.0.113.146"]
}
```

- Usado se nenhuma configuração específica da regra e DNS falhar
- Endereços P-CSCF padrão global
- **Vantagem**: Sempre disponível, evita falha de sessão

Exemplo de Lógica de Fallback

```
A sessão corresponde à regra "Tráfego IMS":

1. Tente a descoberta DNS para "pcscf.ims.example.com"
   ├─ Sucesso → Usar [203.0.113.100, 203.0.113.101] ✓
   └─ Falhou → Tente a próxima camada

2. Tente a sobrescrição do PCO da regra
   ├─ Configurado → Usar [203.0.113.100, 203.0.113.101] ✓
   └─ Não configurado → Tente a próxima camada

3. Use a configuração global do PCO
   └─ Usar [203.0.113.146] ✓ (Sempre tem sucesso)
```

Configuração do DNS

Configuração do Servidor DNS

Configure o servidor DNS com registros SRV e A/AAAA para descoberta do P-CSCF:

```
; Registros SRV para o P-CSCF (_sip._tcp prefix é consultado
automaticamente)
_sip._tcp.pcscf.mnc001.mcc001.3gppnetwork.org. IN SRV 10 50 5060
pcscf1.example.com.
_sip._tcp.pcscf.mnc001.mcc001.3gppnetwork.org. IN SRV 20 50 5060
pcscf2.example.com.

; Registros A
pcscf1.example.com. IN A 203.0.113.100
pcscf2.example.com. IN A 203.0.113.101
```

Importante: O OmniPGW automaticamente adiciona `_sip._tcp.` ao FQDN configurado. Se você configurar `p_cscf_discovery_fqdn: "pcscf.mnc001.mcc001.3gppnetwork.org"`, o sistema consultará `_sip._tcp.pcscf.mnc001.mcc001.3gppnetwork.org`.

Formato do Registro SRV

Os registros SRV seguem este formato:

```
_serviço._proto.dominio. IN SRV prioridade peso porta alvo.
```

- **Prioridade**: Valores mais baixos têm maior prioridade (10 antes de 20)
- **Peso**: Para balanceamento de carga entre a mesma prioridade (maior = mais tráfego)
- **Porta**: Porta SIP (tipicamente 5060 para TCP, 5060 para UDP)
- **Alvo**: Nome do host a ser resolvido para o endereço IP

Testando a Configuração DNS

```
# Consultar registros SRV (note o prefixo _sip._tcp)
dig SRV _sip._tcp.pcscf.mnc001.mcc001.3gppnetwork.org @192.0.2.177

# Saída esperada:
# _sip._tcp.pcscf.mnc001.mcc001.3gppnetwork.org. 300 IN SRV 10 50
5060 pcscf1.example.com.

# Resolver nome do P-CSCF para IP
dig A pcscf1.example.com @192.0.2.177

# Saída esperada:
# pcscf1.example.com. 300 IN A 203.0.113.100
```

Solução de Problemas

Problema: FQDN Mostra Status "Falhou"

Sintomas:

- `GET /api/pcscf_monitor` relata status Falhou
- Erro: `:nxdomain`, `:timeout` ou `:no_naptr_records`

Causas Possíveis:

1. Servidor DNS não acessível
2. FQDN não existe no DNS
3. Nenhum registro NAPTR configurado
4. Timeout do servidor DNS

Resolução:

```
# 1. Testar conectividade do servidor DNS
ping 192.0.2.177

# 2. Testar consulta NAPTR manualmente
dig NAPTR pcscf.mnc001.mcc001.3gppnetwork.org @192.0.2.177

# 3. Verificar logs do OmniPGW
grep "P-CSCF" /var/log/pgw_c.log

# 4. Verificar configuração
grep "p_cscf_discovery_dns_server" config/runtime.exs

# 5. Reconsultar a API OAM para o estado atual
curl -k https://localhost:8443/api/pcscf_monitor
```

Problema: Nenhum IP Retornado

Sintomas:

- `GET /api/pcscf_monitor` relata "0 IPs"
- O status pode ser Resolvido ou Falhado

Causas Possíveis:

1. Registros NAPTR existem, mas FQDNs de substituição não resolvem
2. O campo de serviço não corresponde ao padrão IMS/SIP
3. Registros A/AAAA ausentes

Resolução:

```
# Verificar o campo de serviço do registro NAPTR
dig NAPTR pcscf.example.com @192.0.2.177

# Garantir que o serviço contenha "SIP" ou "IMS":
# CORRETO: "SIP+D2U", "x-3gpp-ims:sip"
# ERRADO: "HTTP", "FTP"

# Verificar se os registros A/AAAA existem
dig pcscf1.example.com A @192.0.2.177
```

Problema: Sessões Usam o P-CSCF Errado

Sintomas:

- UE recebe endereços P-CSCF inesperados
- Fallback estático usado em vez de IPs descobertos

Causas Possíveis:

1. A descoberta DNS falhou, mas o fallback está funcionando
2. Correspondência de regra incorreta
3. FQDN não registrado

Resolução:

```
# 1. Verificar a página do Monitor do P-CSCF
# Verificar se o FQDN está registrado e resolvido

# 2. Verificar logs da sessão
grep "Usando endereços do P-CSCF do FQDN" /var/log/pgw_c.log

# 3. Verificar a página de Seleção do UPF
# Verificar se a regra mostra o FQDN correto e o status

# 4. Testar correspondência de regra
# Criar sessão com APN específico e verificar qual regra
corresponde
```

Problema: Alta Latência de Consulta DNS

Sintomas:

- Criação de sessão lenta
- Métricas mostram alta `pcscf_discovery_query_duration_seconds`

Causas Possíveis:

1. Problemas de desempenho do servidor DNS
2. Latência de rede para o servidor DNS

3. Timeout muito alto

Resolução:

```
# Reduzir o timeout da consulta
pco: %{
  p_cscf_discovery_timeout_ms: 2000  # Reduzir de 5000ms
}

# Considerar usar um servidor DNS mais próximo
pco: %{
  p_cscf_discovery_dns_server: "10.0.0.10"  # DNS local
}
```

Melhores Práticas

1. Seleção do Servidor DNS

Use Servidor DNS Dedicado

```
pco: %{
  # DNS dedicado para descoberta do P-CSCF (não o mesmo que o DNS
do UE)
  p_cscf_discovery_dns_server: "192.0.2.177",

  # Servidores DNS do UE (fornecidos aos dispositivos móveis)
  primary_dns_server_address: "8.8.8.8",
  secondary_dns_server_address: "8.8.4.4"
}
```

Por quê?

- Separar preocupações: DNS do UE vs. DNS interno do IMS
- Diferentes políticas de acesso e segurança
- Escalonamento e confiabilidade independentes

2. Sempre Configure Fallback Estático

```
%{
  p_cscf_discovery_fqdn: "pcscf.ims.example.com",  # Preferido
  pco: %{
    p_cscf_ipv4_address_list: ["203.0.113.100"]  # Fallback
necessário
  }
}
```

Por quê?

- Garante que as sessões tenham sucesso mesmo se o DNS falhar
- Degradação graciosa
- Atende aos requisitos de SLA

3. Use FQDNs Específicos por Tipo de Tráfego

```
rules: [
  # IMS
  %{
    name: "IMS",
    match_regex: "^ims",
    p_cscf_discovery_fqdn:
"pcscf.ims.mnc001.mcc001.3gppnetwork.org"
  },

  # Empresa
  %{
    name: "Empresa",
    match_regex: "^enterprise",
    p_cscf_discovery_fqdn: "pcscf.enterprise.example.com"
  }
]
```

Por quê?

- Diferentes pools de P-CSCF por serviço
- Melhor distribuição de carga

- Roteamento específico para o serviço

4. Monitore o Desempenho da Consulta DNS

```
# Alerta sobre alta latência de consulta P-CSCF
alert: HighPCSCFQueryLatency
expr: histogram_quantile(0.95,
pcscf_discovery_query_duration_seconds_bucket) > 2
for: 5m
labels:
  severity: warning
annotations:
  summary: "Consultas DNS do P-CSCF estão lentas (p95 > 2s)"
```

5. Verificações de Saúde DNS Regulares

- **API OAM**: Consulte `GET /api/pcscf_monitor` diariamente
- **Métricas**: Monitore a métrica `pcscf_monitor_fqdns_failed`
- **Logs**: Fique de olho em erros de DNS
- **Testes**: Verifique periodicamente se os registros DNS existem

6. Configure um Timeout Apropriado

```
# Produção: Equilibre confiabilidade e latência
pco: %{
  p_cscf_discovery_timeout_ms: 5000  # 5 segundos
}

# Alto desempenho: Priorize velocidade, confie no fallback
pco: %{
  p_cscf_discovery_timeout_ms: 2000  # 2 segundos
}
```

7. Use Redundância DNS

Configure DNS primário e secundário:

```
# DNS Primário do P-CSCF
pcscf.mnc001.mcc001.3gppnetwork.org. IN NAPTR 10 50 "s" "SIP+D2U"
"" _sip._udp.pcscf1.example.com.

# DNS Secundário do P-CSCF
pcscf.mnc001.mcc001.3gppnetwork.org. IN NAPTR 20 50 "s" "SIP+D2U"
"" _sip._udp.pcscf2.example.com.
```

Documentação Relacionada

- **Configuração do PCO** - Opções de Configuração de Protocolo, configurações de DNS e P-CSCF
- **Guia de Configuração** - Referência completa de configuração do OmniPGW
- **Monitoramento** - Métricas, registro e observabilidade
- **Gerenciamento de Sessão** - Ciclo de vida da sessão e entrega do PCO
- **Interface PFCP** - Comunicação da Função de Plano de Usuário

---

Voltar à Documentação Principal

Documentação da Interface PFCP/Sxb

Protocolo de Controle de Encaminhamento de Pacotes - Comunicação entre PGW-C e PGW-U

Índice

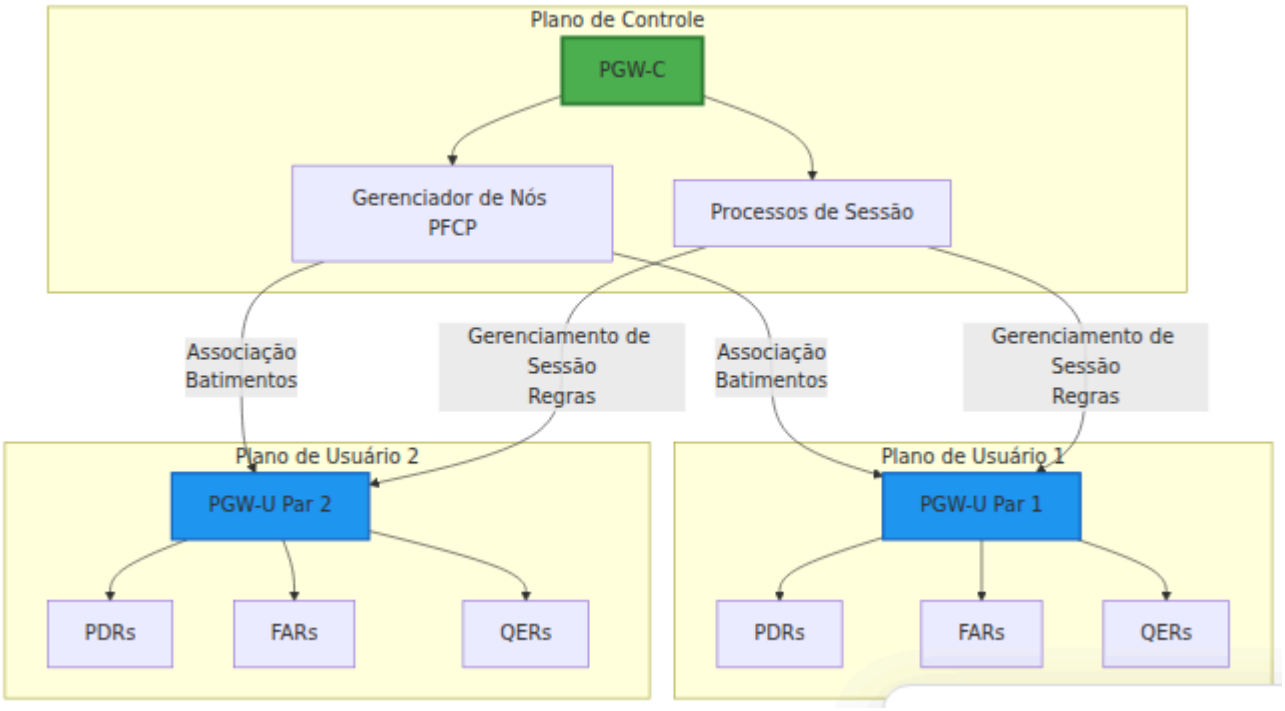
1. Visão Geral
2. Fundamentos do Protocolo
3. Gerenciamento de Associação PFCP
4. Gerenciamento de Sessão PFCP
5. Regras de Processamento de Pacotes
6. Configuração
7. Seleção de UPF Baseada em DNS
8. Fluxos de Mensagens
9. Solução de Problemas
10. Monitoramento PFCP
11. Documentação Relacionada

Visão Geral

A **interface Sxb** utiliza o **PFCP (Protocolo de Controle de Encaminhamento de Pacotes)** para comunicação entre o PGW-C (plano de controle) e o PGW-U (plano de usuário). Essa separação permite:

- **Plano de Controle (PGW-C)** - Gerencia sinalização, gerenciamento de sessão, decisões de política
- **Plano de Usuário (PGW-U)** - Gerencia o encaminhamento real de pacotes em alta velocidade

Arquitetura PFCP



Fundamentos do Protocolo

Versão PFCP

O PGW-C implementa **PFCP Versão 1** (3GPP TS 29.244).

Transporte

- **Protocolo:** UDP
- **Porta Padrão:** 8805
- **Formato da Mensagem:** Codificado em binário usando a especificação PFCP

Tipos de ID de Nó

Os pares PFCP são identificados pelo ID de Nó, que pode ser:

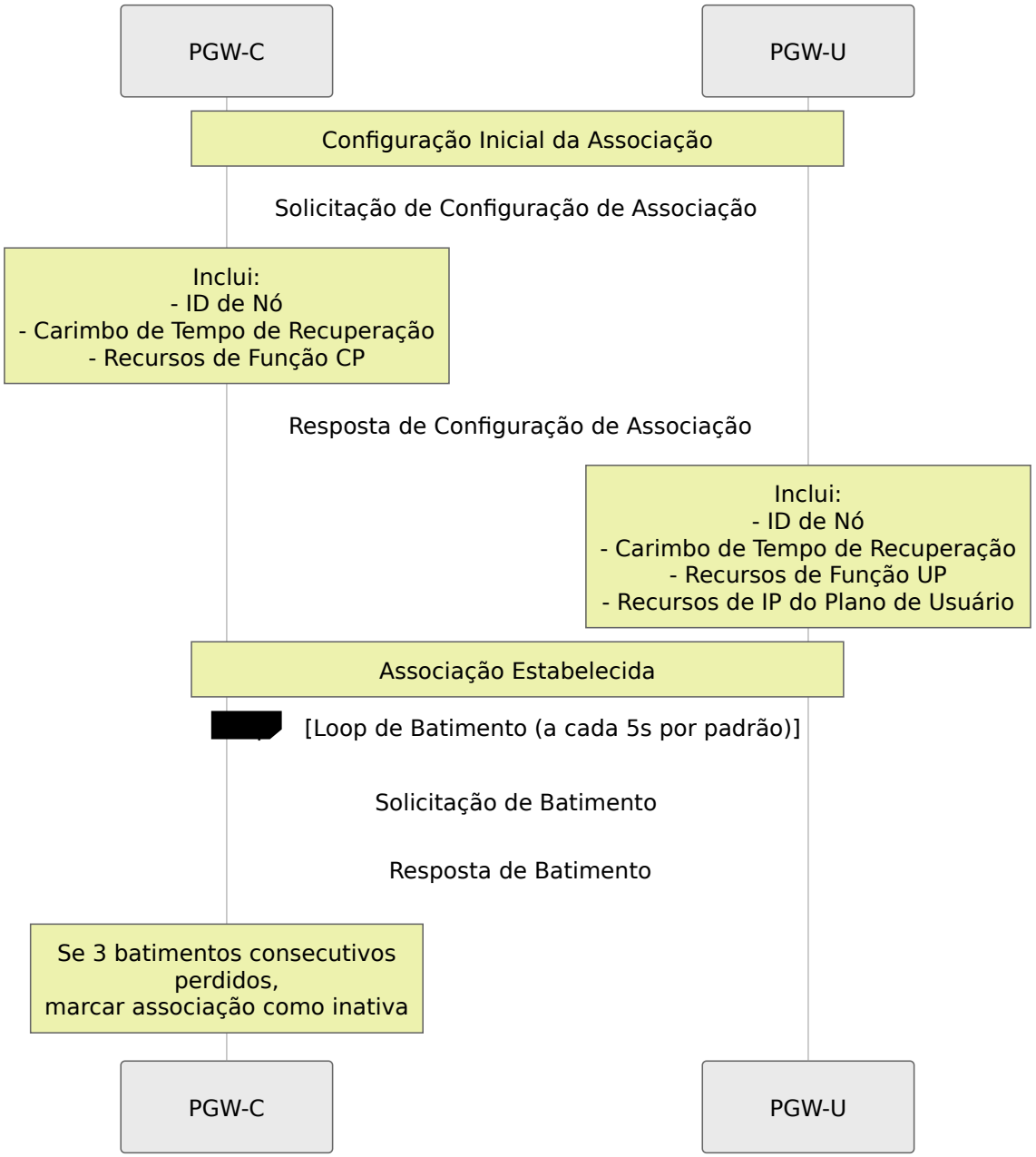
- **Endereço IPv4** - Mais comum

- **Endereço IPv6**
- **FQDN** (Nome de Domínio Totalmente Qualificado)

Gerenciamento de Associação PFCP

Antes que o gerenciamento de sessão possa ocorrer, uma **associação** PFCP deve ser estabelecida entre PGW-C e PGW-U.

Fluxo de Configuração de Associação



Gerenciamento de Estado do Par

Cada par PFCP mantém estado:

Campo	Descrição
`is_associated`	Booleano indicando o status da associação
`remote_node_id`	ID de Nó do par (IP ou FQDN)
`remote_ip_address`	Endereço IP para comunicação
`remote_port`	Porta UDP (padrão 8805)
`heartbeat_period_ms`	Intervalo de batimento em milissegundos
`missed_heartbeats_consecutive`	Contagem de batimentos perdidos
`up_function_features`	Recursos de plano de usuário suportados
`up_recovery_time_stamp`	Carimbo de tempo de recuperação do par

Mecanismo de Batimento

Propósito: Detectar falhas de pares e manter a vivacidade da associação

Configuração:

```
# Em config/runtime.exs
sxb: %{
  local_ip_address: "10.0.0.20"
},
upf_selection: %{
  fallback_pool: [
    %{remote_ip_address: "10.0.0.21", remote_port: 8805, weight:
100}
  ]
}
# Todos os UPFs são registrados automaticamente com batimentos de
5 segundos
```

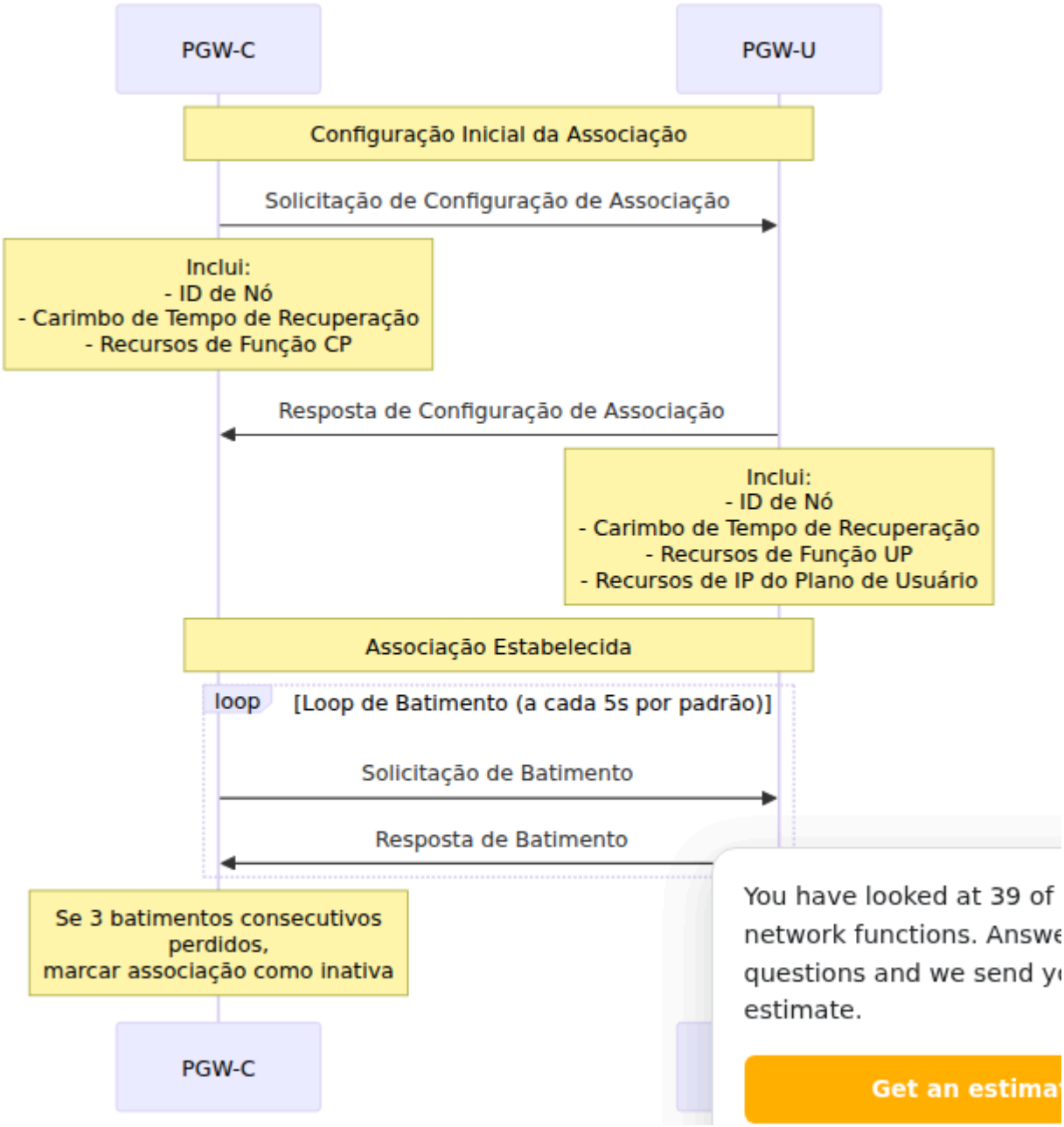
Detecção de Falhas:

- Cada batimento perdido incrementa `missed_heartbeats_consecutive`
- Geralmente configurado para falhar após 3 perdas consecutivas
- Associação falhada impede novas sessões para aquele par

Gerenciamento de Sessão PFCP

As sessões PFCP são criadas para cada conexão PDN de UE para programar regras de encaminhamento no plano de usuário.

Ciclo de Vida da Sessão



Estabelecimento de Sessão

Quando: UE se conecta e cria uma conexão PDN

PGW-C envia para PGW-U:

Solicitação de Estabelecimento de Sessão contendo:

- **SEID** (ID de Ponto de Extremidade da Sessão) - Identificador único da sessão

- **ID de Nó** - ID de Nó do PGW-C
- **F-SEID** - SEID Totalmente Qualificado (inclui IP + SEID)
- **PDRs** - Regras de Detecção de Pacotes (tipicamente 2: uplink + downlink)
- **FARs** - Regras de Ação de Encaminhamento (tipicamente 2: uplink + downlink)
- **QERs** - Regras de Aplicação de QoS (limites de taxa de bits)
- **BAR** - Regra de Ação de Bufferização (para bufferização de downlink)

PGW-U responde:

Resposta de Estabelecimento de Sessão contendo:

- **Causa** - Razão de sucesso ou falha
- **F-SEID** - Ponto de extremidade da sessão do PGW-U
- **PDRs Criados** - Reconhecimento das regras criadas
- **F-TEID** - TEID Totalmente Qualificado para a interface S5/S8

Modificação de Sessão

Quando: Mudanças de QoS, atualizações de política ou modificações de bearer ocorrem

A modificação pode incluir:

- Adicionar novos PDRs, FARs, QERs
- Remover regras existentes
- Atualizar parâmetros de regra

Exclusão de Sessão

Quando: UE se desconecta ou a conexão PDN é encerrada

Processo:

1. PGW-C envia Solicitação de Exclusão de Sessão com SEID
2. PGW-U remove todas as regras e libera recursos
3. PGW-U responde com Resposta de Exclusão de Sessão

Alocação de F-TEID

F-TEID (Identificador de Ponto de Extremidade de Túnel Totalmente Qualificado) identifica pontos de extremidade de túnel GTP-U para tráfego do plano de usuário. Ao estabelecer uma sessão PFCP, alguém deve alocar o F-TEID que identifica onde o UPF deve enviar tráfego de uplink. Existem duas abordagens:

Compreendendo a Alocação de F-TEID

O que está sendo alocado: O F-TEID consiste em:

- **TEID** (Identificador de Ponto de Extremidade de Túnel) - Número de 32 bits identificando o túnel
- **Endereço IP** - Para onde enviar pacotes GTP-U (o endereço IP do UPF)

A Pergunta: Quem aloca o valor TEID?

Opção 1: UPF Aloca (Padrão Recomendado)

- PGW-C diz "por favor, aloque um TEID para mim" (flag CHOOSE)
- UPF escolhe um TEID de sua piscina local e responde com o valor

Opção 2: PGW-C Aloca (Modo de Compatibilidade)

- PGW-C escolhe um TEID e diz ao UPF "use este TEID específico"
- UPF usa o TEID fornecido sem alocação

Alocação UPF (Padrão - Recomendado)

Configuração:

```
sxb: %{
  allocate_uplink_f_teid: false  # Padrão
}
```

Como Funciona:

1. PGW-C constrói Solicitação de Estabelecimento de Sessão PFCP com flag CHOOSE para F-TEID
2. UPF recebe a solicitação, aloca TEID de sua piscina interna
3. UPF responde com F-TEID alocado (TEID + endereço IP)
4. PGW-C armazena o F-TEID alocado durante a vida útil da sessão

Por que isso é melhor (geralmente):

□ Separação de Preocupações

- UPF possui o plano de usuário = UPF gerencia identificadores do plano de usuário
- Sem necessidade do PGW-C rastrear quais TEIDs o UPF tem disponíveis
- Cada componente gerencia sua própria piscina de recursos

□ Escalabilidade Multi-PGW-C

- Várias instâncias de PGW-C podem se comunicar com o mesmo UPF sem coordenação
- Sem risco de colisões de TEID entre diferentes instâncias de PGW-C
- UPF garante unicidade entre todos os pares do plano de controle

□ Comportamento Padrão 3GPP

- A flag CHOOSE é definida na 3GPP TS 29.244 para esse propósito
- Implementações modernas de UPF a suportam
- Segue o princípio "deixe o proprietário alocar"

□ Failover Mais Simples

- Se o PGW-C reiniciar, o UPF ainda possui o namespace de TEID
- Sem necessidade de sincronizar o estado de alocação de TEID
- UPF pode continuar usando TEIDs existentes

Quando Usar:

- □ Implantações de produção com UPFs modernos (padrão)
- □ Implantações Multi-PGW-C compartilhando piscinas de UPF

- □ Arquiteturas nativas da nuvem com planos de controle sem estado
- □ Você deseja o comportamento padrão PFCP 3GPP

Problemas Potenciais:

- ⚠ Algumas implementações de UPF legadas ou proprietárias não suportam a flag CHOOSE
- ⚠ Se o estabelecimento da sessão falhar com "IE obrigatório ausente" ou similar, o UPF pode não suportar CHOOSE

Alocação PGW-C (Compatibilidade Legada)

Configuração:

```
sxb: %{
  allocate_uplink_f_teid: true
}
```

Como Funciona:

1. PGW-C aloca TEID de sua piscina local durante a criação da sessão
2. PGW-C constrói Solicitação de Estabelecimento de Sessão PFCP com valor TEID explícito
3. UPF recebe a solicitação, usa o TEID fornecido sem alocação
4. Tanto PGW-C quanto UPF rastreiam o mesmo valor TEID

Por que você pode precisar disso:

□ UPF Não Suporta CHOOSE

- Algumas implementações de UPF (especialmente legadas/proprietárias) não suportam alocação dinâmica
- UPF espera TEID explícito na Solicitação de Estabelecimento de Sessão PFCP
- Única solução para compatibilidade

□ Gerenciamento Centralizado de TEID

- Se você precisa que o PGW-C tenha visibilidade total sobre todos os TEIDs alocados
- Útil para depuração de problemas do plano de usuário (PGW-C conhece os valores exatos de TEID)
- Pode correlacionar TEID em capturas de pacotes com o estado da sessão

□ Alocação Determinística

- Se você precisa de padrões de alocação de TEID previsíveis
- Alguns ambientes de teste podem exigir faixas específicas de TEID

Compensações:

⚠ Coordenação Necessária para Multi-PGW-C

- Múltiplas instâncias de PGW-C compartilhando um UPF devem evitar colisões de TEID
- Requer:
  - Faixas de TEID particionadas por PGW-C (configuração complexa)
  - Serviço de alocação de TEID compartilhado (infraestrutura adicional)
  - Aceitar risco de colisão com alocação aleatória (baixa probabilidade)

⚠ Sincronização de Estado

- PGW-C deve rastrear TEIDs alocados para evitar reutilização
- Estado da piscina de TEID perdido na reinicialização do PGW-C (deve ser reconstruído a partir das sessões)
- Cenários de failover mais complexos

⚠ Comportamento Não Padrão

- Não é o padrão de design pretendido do PFCP
- Pode não funcionar com todas as implementações de UPF que esperam CHOOSE

Quando Usar:

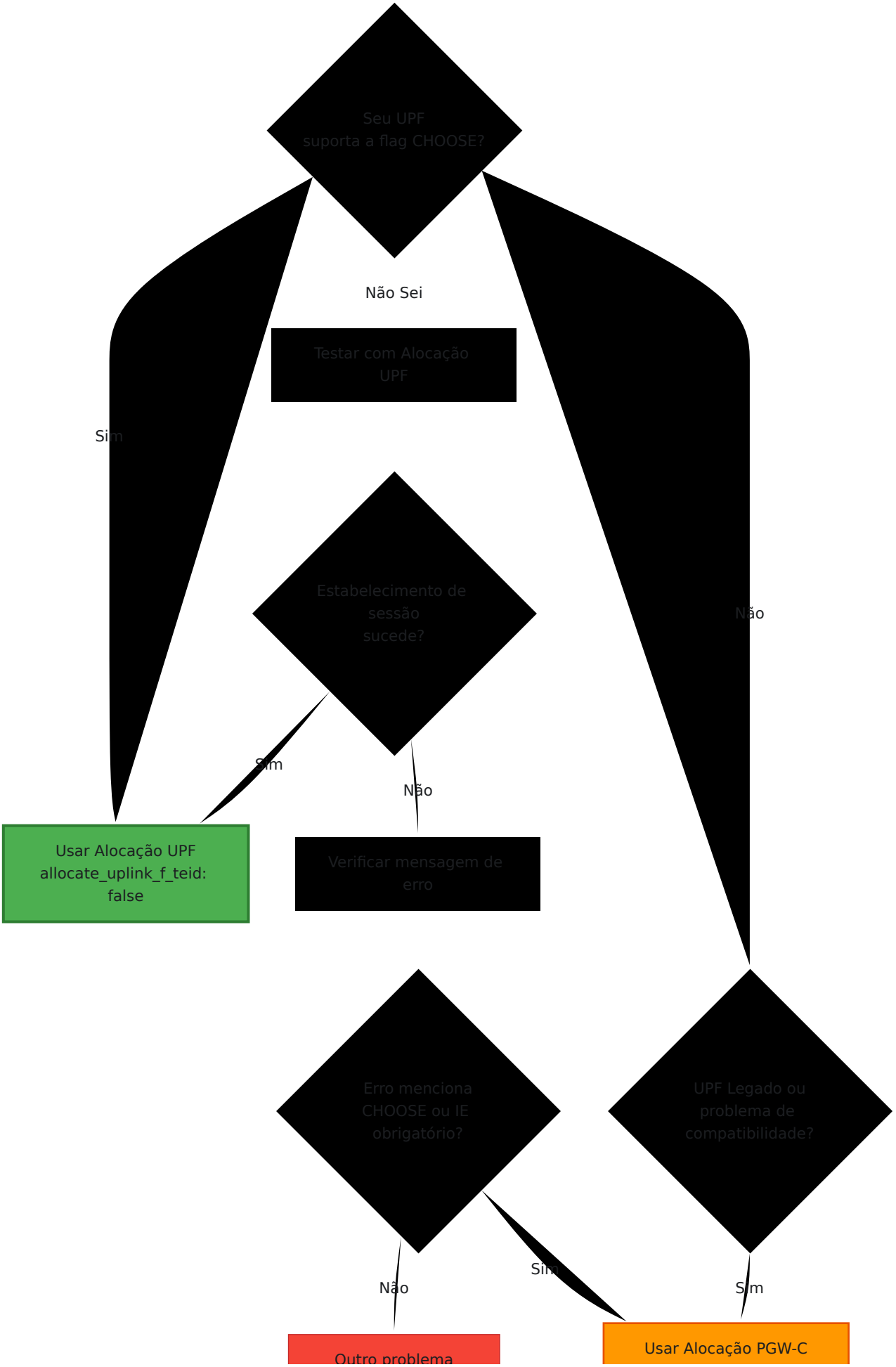
- **⚠ Apenas quando o UPF não suportar a flag CHOOSE**

- ⚠ Implementações legadas de UPF (por exemplo, algum hardware proprietário)
- ⚠ Requisitos específicos de compatibilidade
- ⚠ Cenários de depuração que requerem visibilidade de TEID do PGW-C

Tratamento de Colisões de TEID: PGW-C usa alocação aleatória com detecção de colisões:

- Faixa de TEID: 1 a 0xFFFFFFFF (4,2 bilhões de valores)
- Probabilidade de colisão: ~0,023% em 1 milhão de sessões
- Tentativa automática em caso de colisão (transparente para o chamador)
- TEIDs liberados automaticamente quando a sessão termina

Como Escolher


Seu UPF suporta a flag CHOOSE?
Não Sei
Testar com Alocação UPF
Sim
Não
Estabelecimento de sessão sucede?
Sim
Não
Usar Alocação UPF allocate_uplink_f_teid: false
Verificar mensagem de erro
Erro menciona CHOOSE ou IE obrigatório?
UPF Legado ou problema de compatibilidade?
Não
Sim
Sim
Outro problema
Usar Alocação PGW-C



Solução de Problemas

Sintoma: Estabelecimento de sessão falha imediatamente

Verifique os logs PFCP:

```
# Procure por erros relacionados a CHOOSE
grep -i "choose\|mandatory.*missing" /var/log/pgw_c.log

# Verifique os códigos de causa da Resposta de Estabelecimento de
Sessão PFCP
grep "Session Establishment Response" /var/log/pgw_c.log
```

Se o UPF rejeitar a flag CHOOSE:

- O erro pode dizer "IE obrigatório ausente" ou "IE inválido"
- O UPF espera um F-TEID explícito, mas recebeu CHOOSE
- **Solução:** Defina `allocate_uplink_f_teid: true`

Se a alocação do PGW-C causar problemas:

- Muito raro - o espaço de TEID é enorme (4 bilhões de valores)
- Verifique a exaustão de TEID (improvável abaixo de milhões de sessões):

```
# Verifique a contagem do registro
grep "registered_teid_count" /var/log/pgw_c.log
```

Alternando Entre Modos:

```
# Edite config/runtime.exs
sxb: %{
  local_ip_address: "10.0.0.20",
  allocate_uplink_f_teid: false  # Mude para true se o UPF não
suportar CHOOSE
}
```

Em seguida, reinicie o PGW-C:

```
systemctl restart pgw_c
```

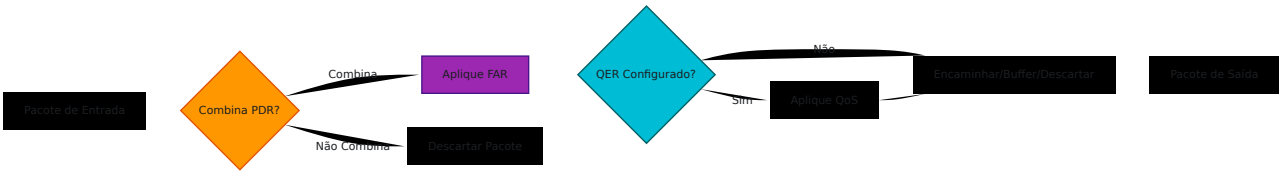
Verificando Qual Modo Está Ativo: Verifique as capturas de pacotes PFCP:

```
# Capture o tráfego PFCP
tcpdump -i any -n port 8805 -w pfcp.pcap

# Abra no Wireshark e veja a Solicitação de Estabelecimento de
Sessão
# Se o F-TEID mostrar flags "CHOOSE": modo de alocação UPF
# Se o F-TEID mostrar valor TEID explícito: modo de alocação PGW-C
```

Regras de Processamento de Pacotes

O PFCP usa um conjunto de regras para definir como o plano de usuário processa pacotes.

Arquitetura de Regras



PDR (Regra de Detecção de Pacotes)

Propósito: Identificar quais pacotes esta regra se aplica

Configuração Típica do PGW-C:

PDR #1 - Downlink:

```
ID do PDR: 1
Precedência: 100
PDI (Informações de Detecção de Pacotes):
  - Interface de Origem: CORE (lado da Internet)
  - Endereço IP do UE: 100.64.1.42/32
ID do FAR: 1 (regra de encaminhamento associada)
```

PDR #2 - Uplink:

```
ID do PDR: 2
Precedência: 100
PDI (Informações de Detecção de Pacotes):
  - Interface de Origem: ACCESS (lado do SGW)
  - F-TEID: <ponto de extremidade do túnel S5/S8>
ID do FAR: 2 (regra de encaminhamento associada)
ID do QER: 1 (aplicação de QoS)
```

Campos Chave do PDR:

- **ID do PDR** - Identificador único da regra (por sessão)
- **Precedência** - Prioridade de correspondência da regra (mais alto = mais específico)
- **PDI** - Critérios de correspondência (interface, IP, TEID, etc.)
- **Remoção do Cabeçalho Externo** - Remover cabeçalho GTP-U na entrada
- **ID do FAR** - Ação de encaminhamento associada
- **ID do QER** - Aplicação de QoS associada (opcional)

FAR (Regra de Ação de Encaminhamento)

Propósito: Definir o que fazer com pacotes combinados

FAR #1 - Downlink (Internet → UE):

```
ID do FAR: 1
Ação Aplicada: ENCAMINHAR
Parâmetros de Encaminhamento:
  - Interface de Destino: ACCESS (para SGW)
  - Criação de Cabeçalho Externo: GTP-U/UDP/IPv4
  - F-TEID Remoto: <ponto de extremidade do túnel S5/S8 do SGW>
```

FAR #2 - Uplink (UE → Internet):

```
ID do FAR: 2
Ação Aplicada: ENCAMINHAR
Parâmetros de Encaminhamento:
  - Interface de Destino: CORE (para Internet)
  - (Sem cabeçalho externo - encaminhamento IP simples)
```

Campos Chave do FAR:

- **ID do FAR** - Identificador único da regra
- **Ação Aplicada** - ENCAMINHAR, DESCARTAR, BUFFER, NOTIFICAR
- **Parâmetros de Encaminhamento:**
  - Interface de destino (ACCESS/CORE)
  - Criação de Cabeçalho Externo (adicionar túnel GTP-U)
  - Instância de Rede (VRF/tabela de roteamento)

QER (Regra de Aplicação de QoS)

Propósito: Aplicar limites de taxa de bits e parâmetros de QoS. QERs também podem rastrear o uso para gerenciamento de cotas de cobrança online (veja Interface Diameter Gy para controle de crédito).

Exemplo de QER:

```
ID do QER: 1
Status do Portão: ABERTO
Taxa Máxima:
  - Uplink: 100 Mbps
  - Downlink: 50 Mbps
Taxa Garantida: (opcional, para bearers GBR)
  - Uplink: 10 Mbps
  - Downlink: 10 Mbps
```

Campos Chave do QER:

- **ID do QER** - Identificador único da regra
- **Status do Portão** - ABERTO (permitir) ou FECHADO (bloquear)
- **MBR** - Taxa Máxima (uplink/downlink)
- **GBR** - Taxa Garantida (para bearers dedicados)
- **QCI** - Identificador de Classe de QoS (afeta o agendamento)

BAR (Regra de Ação de Bufferização)

Propósito: Controlar o buffer de pacotes de downlink quando o UE está ocioso

Exemplo de BAR:

```
ID do BAR: 1
Atraso na Notificação de Dados de Downlink: 100ms
Contagem Sugerida de Pacotes para Bufferização: 10
```

Usado para: Otimização de DRX (Recepção Descontínua) em modo ocioso

---

Configuração

Configuração Básica do Sxb

Edite `config/runtime.exs`:

```
config :pgw_c,
  sxb: %{
    # Endereço IP local para comunicação PFCP
    local_ip_address: "10.0.0.20",

    # Opcional: Substituir porta padrão (8805)
    local_port: 8805,

    # Tempo limite de solicitação PFCP em milissegundos (padrão:
500ms)
    # Tempo a esperar pela resposta do UPF antes de retransmitir
    # Deve ser >= tempo de processamento esperado do UPF para
evitar sessões duplicadas
    request_timeout_ms: 500,

    # Número de tentativas de retransmissão para solicitações PFCP
(padrão: 3)
    # Tempo máximo total de espera = request_timeout_ms *
request_attempts
    request_attempts: 3,

    # Opcional: Controlar alocação de F-TEID para o plano de
usuário
    # Quando falso (padrão): UPF aloca F-TEID (flag CHOOSE)
    # Quando verdadeiro: PGW-C pré-aloca F-TEID e fornece valor
explícito
    # Nota: Alguns UPFs podem não suportar a flag CHOOSE e exigir
alocação explícita
    allocate_uplink_f_teid: false
  },

  # Seleção de UPF - Todos os UPFs definidos aqui são registrados
automaticamente
  upf_selection: %{
    fallback_pool: [
      %{
        # Endereço IP do PGW-U
        remote_ip_address: "10.0.0.21",

        # Porta PFCP (padrão: 8805)
        remote_port: 8805,

        # Peso para balanceamento de carga (100 = normal, 0 =
```

```
reserva)
        weight: 100
      }
    ]
  }
```

Configuração de Tempo Limite de Solicitação

A interface PFCP usa tempos limites configuráveis para Solicitações de Estabelecimento, Modificação e Exclusão de Sessões para o UPF.

Parâmetros:

Parâmetro	Tipo	Padrão	Descrição
`request_timeout_ms`	Inteiro	500	Tempo em milissegundos para esperar pela resposta do UPF antes de retransmitir
`request_attempts`	Inteiro	3	Número máximo de tentativas de transmissão antes de falhar a solicitação

Tempo Total de Espera: `request_timeout_ms × request_attempts`

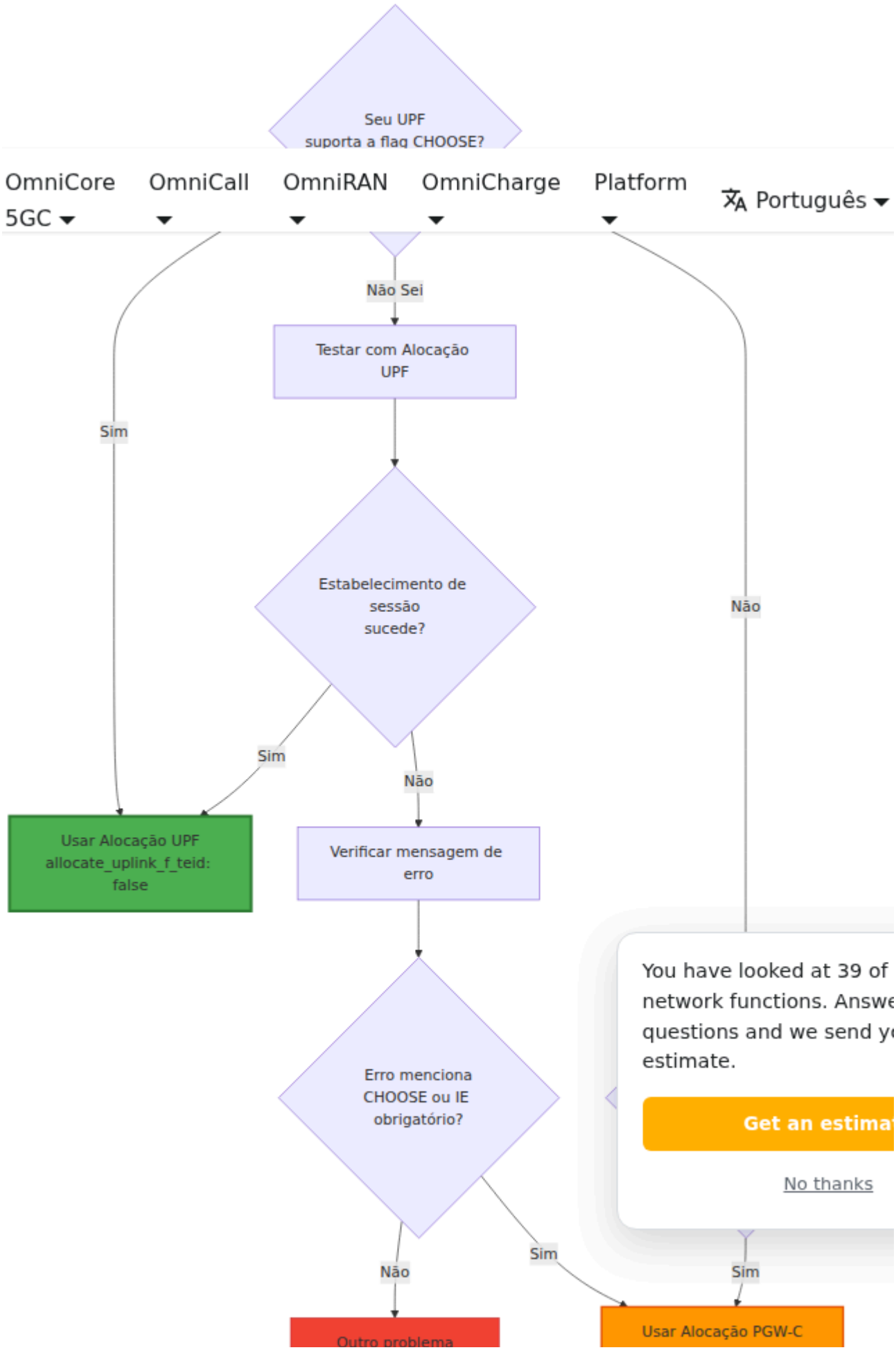
Comportamento padrão: 500ms × 3 tentativas = **1,5 segundos de espera máxima total**

Por que a Configuração de Tempo Limite é Importante

Se `request_timeout_ms` estiver definido muito baixo em relação ao tempo de processamento do UPF:

1. PGW-C envia Solicitação de Estabelecimento de Sessão
2. O tempo limite expira antes que o UPF responda
3. PGW-C retransmite com o mesmo número de sequência
4. O UPF processa ambas as solicitações e cria **sessões PFCP duplicadas**
5. PGW-C recebe a primeira resposta e armazena um ID de sessão

6. A segunda sessão se torna **órfã** no UPF


Seu UPF suporta a flag CHOOSE?
OmniCore 5GC
OmniCall
OmniRAN
OmniCharge
Platform
Português
Não Sei
Testar com Alocação UPF
Sim
Estabelecimento de sessão sucede?
Não
Sim
Não
Usar Alocação UPF allocate_uplink_f_teid: false
Verificar mensagem de erro
You have looked at 39 of
network functions. Answe
questions and we send y
estimate.
Get an estima
No thanks
Erro menciona CHOOSE ou IE obrigatório?
Não
Sim
Sim
Outro problema
Usar Alocação PGW-C



Diretrizes de Ajuste

Tempo de Resposta do UPF	`request_timeout_ms` Recomendado	Tempo Total de Espera
Rápido (<100ms)	200-300ms	600-900ms (3 tentativas)
Normal (100-300ms)	500ms (padrão)	1,5s (3 tentativas)
Lento (300-500ms)	750-1000ms	2,25-3s (3 tentativas)
Muito lento (>500ms)	1500-2000ms	4,5-6s (3 tentativas)

Recomendação: Defina `request_timeout_ms` para pelo menos **2× o tempo de resposta esperado do UPF** para evitar sessões órfãs induzidas por retransmissão.

Exemplo - UPF Lento

```
sxb: %{
  local_ip_address: "10.0.0.20",
  request_timeout_ms: 1000,  # 1 segundo por tentativa
  request_attempts: 3         # Total: 3 segundos no máximo
}
```

Diagnóstico de Problemas de Tempo Limite

Sintomas de tempo limite muito baixo:

- O UPF relata mais sessões PFCP do que o esperado
- Sessões órfãs se acumulam no UPF ao longo do tempo
- Mensagens de log: `"Solicitação de Estabelecimento de Sessão expirou"` seguida por sessão bem-sucedida

Como diagnosticar:

1. Verifique a contagem de sessões do UPF via API ou interface de gerenciamento do UPF
2. Compare com a contagem de sessões ativas do PGW-C
3. Se o UPF tiver mais sessões, existem órfãs de retransmissões

Resolução:

1. Aumente `request_timeout_ms` para exceder o tempo de resposta do UPF
2. Reinicie o PGW-C para aplicar a nova configuração
3. Limpe sessões órfãs do UPF (limpeza manual ou reinicialização do UPF)

Múltiplos Pares PGW-U

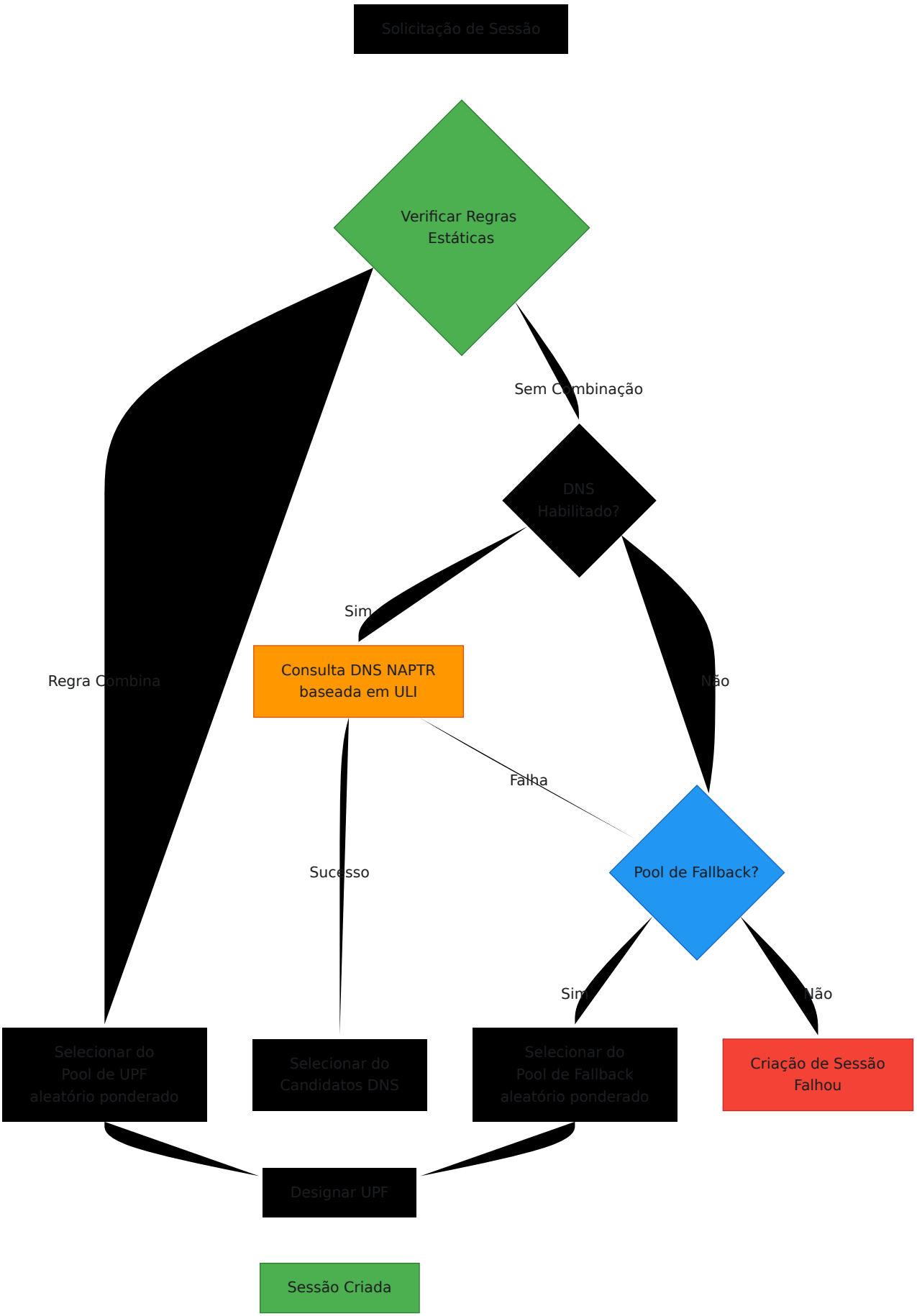
Para balanceamento de carga ou redundância:

```
config :pgw_c,
  sxb: %{
    local_ip_address: "10.0.0.20"
  },
  upf_selection: %{
    fallback_pool: [
      %{remote_ip_address: "10.0.1.21", remote_port: 8805, weight:
50},  # 50% do tráfego
      %{remote_ip_address: "10.0.2.21", remote_port: 8805, weight:
50}   # 50% do tráfego
    ]
  }
# Ambos os UPFs registrados automaticamente com batimentos de 5
segundos
```

Configuração de Seleção de UPF

O PGW-C usa um **sistema de seleção de UPF de três níveis** com regras baseadas em prioridade:

1. **Regras Estáticas** (Maior Prioridade) - Correspondem com base em atributos de sessão
2. **Seleção Baseada em DNS** (Prioridade Média) - Roteamento ciente da localização via consultas DNS NAPTR
3. **Pool de Fallback** (Menor Prioridade) - Pool de UPF padrão quando nenhuma regra corresponde



Exemplo Completo de Seleção de UPF

```
config :pgw_c,
  # Interface PFCP
  sxb: %{
    local_ip_address: "10.0.0.20"
  },

  # Seleção de UPF: Todos os UPFs definidos aqui são registrados
automaticamente
  upf_selection: %{
    #
=============================================================================
    # Seleção Baseada em DNS (Roteamento Ciente da Localização)
    #
=============================================================================
    # Consulta DNS usando Informações de Localização do Usuário (ULI)
    # Fornece seleção dinâmica de UPF com base na localização da
célula
    dns_enabled: false,
    dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],
    dns_suffix: "epc.3gppnetwork.org",
    dns_timeout_ms: 5000,

    #
=============================================================================
    # Regras de Seleção Estáticas (Avaliada por Prioridade)
    #
=============================================================================
    # As regras são verificadas da maior para a menor prioridade
    # A primeira regra correspondente determina o pool de UPF
    rules: [
      # Regra 1: Tráfego IMS - Maior Prioridade
      %{
        name: "Tráfego IMS",
        priority: 20,
        match_field: :apn,
        match_regex: "^ims",
        upf_pool: [
          %{remote_ip_address: "10.100.2.21", remote_port: 8805,
weight: 80},
          %{remote_ip_address: "10.100.2.22", remote_port: 8805,
weight: 20}
        ],
        # Opcional: Sobrescritas de PCO para esta regra
```

```
        pco: %{
          p_cscf_ipv4_address_list: ["203.0.113.100",
"203.0.113.101"]
        }
      },

      # Regra 2: APN Empresarial - Alta Prioridade
      %{
        name: "Tráfego Empresarial",
        priority: 15,
        match_field: :apn,
        match_regex: "^(enterprise|corporate)\.apn",
        upf_pool: [
          %{remote_ip_address: "10.100.3.21", remote_port: 8805,
weight: 100}
        ],
        pco: %{
          primary_dns_server_address: "192.168.1.10",
          secondary_dns_server_address: "192.168.1.11",
          ipv4_link_mtu_size: 1500
        }
      },

      # Regra 3: Assinantes em Roaming - Prioridade Média
      %{
        name: "Assinantes em Roaming",
        priority: 10,
        match_field: :serving_network_plmn_id,
        match_regex: "^(310|311|312|313)",  # Redes dos EUA
        upf_pool: [
          %{remote_ip_address: "10.100.4.21", remote_port: 8805,
weight: 100}
        ]
      },

      # Regra 4: Tráfego de Internet - Menor Prioridade
      %{
        name: "Tráfego de Internet",
        priority: 5,
        match_field: :apn,
        match_regex: "^internet",
        upf_pool: [
          %{remote_ip_address: "10.100.1.21", remote_port: 8805,
weight: 33},
```

```
          %{remote_ip_address: "10.100.1.22", remote_port: 8805,
weight: 33},
          %{remote_ip_address: "10.100.1.23", remote_port: 8805,
weight: 34}
        ]
      }
    ],

    #
=============================================================================
    # Pool de Fallback (Último Recurso)
    #
=============================================================================
    # Usado quando nenhuma regra corresponde e a seleção DNS falha ou
está desabilitada
    fallback_pool: [
      %{remote_ip_address: "127.0.0.21", remote_port: 8805, weight:
100}
    ]
  }
```

Campos de Correspondência Suportados

Campo de Correspondência	Descrição	Valor de Exemplo
`:imsi`	Identidade do Assinante Móvel Internacional	`"310260123456789"`
`:apn`	Nome do Ponto de Acesso	`"internet"`, `"ims"`
`:serving_network_plmn_id`	PLMN da rede servidora (MCC+MNC)	`"310260"` (rede de teste)
`:sgw_ip_address`	Endereço IP do SGW (formato string)	`"10.0.1.50"`
`:uli_tai_plmn_id`	ID PLMN da Área de Rastreamento	`"310260"`
`:uli_ecgi_plmn_id`	ID PLMN da Célula E-UTRAN	`"310260"`

Pool de UPF e Balanceamento de Carga

Cada regra pode especificar um **pool de UPF** com seleção aleatória ponderada:

```
upf_pool: [
  %{remote_ip_address: "10.100.1.21", remote_port: 8805, weight:
50},
  %{remote_ip_address: "10.100.1.22", remote_port: 8805, weight:
30},
  %{remote_ip_address: "10.100.1.23", remote_port: 8805, weight:
20}
]
```

Como Funciona a Seleção Ponderada:

1. Calcule o peso total: 50 + 30 + 20 = 100
2. Gere um número aleatório: 0.0 a 100.0
3. Selecione o UPF com base nas faixas de peso cumulativo:
    - 0-50: UPF-1 (50% de chance)
    - 50-80: UPF-2 (30% de chance)
    - 80-100: UPF-3 (20% de chance)

Casos de Uso:

- **Distribuição igual:** Todos os pesos iguais (33, 33, 34)
- **Primário/reserva:** Primário de alto peso (80), reserva de baixo peso (20)
- **Baseado em capacidade:** Peso proporcional à capacidade do UPF

Sobrescritas de PCO

As regras podem sobrescrever valores de PCO (Opções de Configuração de Protocolo):

```
%{
  name: "Tráfego IMS",
  match_field: :apn,
  match_regex: "^ims",
  upf_pool: [...],
  pco: %{
    # Sobrescrever apenas campos específicos
    p_cscf_ipv4_address_list: ["203.0.113.100", "203.0.113.101"],
    # Outros campos usam os padrões da configuração principal de
pco
  }
}
```

Campos de Sobrescrita de PCO Disponíveis:

- `primary_dns_server_address`
- `secondary_dns_server_address`
- `primary_nbns_server_address`

- `secondary_nbns_server_address`
- `p_cscf_ipv4_address_list`
- `ipv4_link_mtu_size`

Seleção Baseada em DNS

Quando habilitada, o PGW-C realiza consultas DNS NAPTR com base nas Informações de Localização do Usuário:

```
upf_selection: %{
  dns_enabled: true,
  dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],
  dns_suffix: "epc.3gppnetwork.org",
  dns_timeout_ms: 5000
}
```

Prioridade da Consulta:

1. **ECGI** (Identificador Global da Célula E-UTRAN) - Mais específico
2. **TAI** (Identidade da Área de Rastreamento) - Área da célula
3. **RAI** (Identidade da Área de Roteamento) - Área 3G/2G
4. **SAI** (Identidade da Área de Serviço) - Área de serviço 3G
5. **CGI** (Identidade Global da Célula) - Célula 2G

Exemplo de Consulta DNS:

```
# Para consulta ECGI:
eci-1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org

# Para consulta TAI:
tac-lb64.tac-hb00.tac.epc.mnc999.mcc999.epc.3gppnetwork.org
```

Processo de Seleção DNS:

1. Tente consultas na ordem de prioridade (ECGI primeiro, depois TAI, etc.)
2. Se o DNS retornar candidatos, use o primeiro resultado (registrado dinamicamente, se necessário)

3. Selecione o UPF retornado
4. Se nenhuma correspondência DNS ou DNS desabilitado, passe para o pool de fallback

Veja Seleção de UPF Baseada em DNS para informações detalhadas.

Seleção de UPF Baseada em DNS

Visão Geral

A seleção de UPF baseada em DNS fornece **roteamento ciente da localização** realizando consultas DNS NAPTR usando Informações de Localização do Usuário (ULI) da célula atual do UE.

Referência 3GPP: TS 23.003 - Procedimentos DNS para descoberta de UPF

Benefícios:

- Seleção automática de UPF com base na localização geográfica
- Sem configuração manual de regras por célula
- Adaptação dinâmica a mudanças na topologia da rede
- Reduz o backhaul roteando para o UPF mais próximo

Como Funciona



Configuração

```
config :pgw_c,
  upf_selection: %{
    # Habilitar seleção baseada em DNS
    dns_enabled: true,

    # Prioridade da consulta: tentar ECGI primeiro, depois TAI,
depois RAI, etc.
    dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],

    # Sufixo DNS para consultas
    dns_suffix: "epc.3gppnetwork.org",

    # Tempo limite da consulta DNS
    dns_timeout_ms: 5000,

    # Regras estáticas ainda têm precedência sobre DNS
    rules: [...],

    # Fallback se DNS falhar
    fallback_pool: [...]
  }
```

Formatos de Consulta DNS

As consultas DNS são construídas usando Informações de Localização do Usuário (ULI) da mensagem GTP-C:

1. ECGI (Identificador Global da Célula E-UTRAN)

Mais específico - Roteamento em nível de célula LTE

Formato:

```
eci-<HEX-ECI>.ecgi.epc.mnc<MNC>.mcc<MCC>.<dns_suffix>
```

Exemplo:

```
# ID da Célula: 0x1A2B3C (1.715.004 decimal)
# PLMN: MCC=999, MNC=999
eci-1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org
```

Quando Usado: Redes LTE (4G)

2. TAI (Identidade da Área de Rastreamento)

Área da célula - M��ltiplas células na mesma área de rastreamento

Formato:

```
tac-lb<LB>.tac-hb<HB>.tac.epc.mnc<MNC>.mcc<MCC>.<dns_suffix>
```

Exemplo:

```
# TAC: 0x0064 (100 decimal)
# Byte baixo: 0x64, Byte alto: 0x00
tac-lb64.tac-hb00.tac.epc.mnc999.mcc999.epc.3gppnetwork.org
```

Quando Usado: Áreas de rastreamento LTE (4G)

3. RAI (Identidade da Área de Roteamento)

Área de roteamento 3G/2G

Formato:

```
rac<RAC>.lac-lb<LB>.lac-hb<HB>.lac.rai.mnc<MNC>.mcc<MCC>.
<dns_suffix>
```

Exemplo:

```
# RAC: 0x0A (10 decimal)
# LAC: 0x1234 (4660 decimal)
rac0a.lac-lb34.lac-hb12.lac.rai.mnc999.mcc999.epc.3gppnetwork.org
```

Quando Usado: Redes 3G/2G UMTS/GPRS

4. SAI (Identidade da Área de Serviço)

Área de serviço 3G

Formato:

```
sac<SAC>.lac-lb<LB>.lac-hb<HB>.lac.sai.mnc<MNC>.mcc<MCC>.
<dns_suffix>
```

Exemplo:

```
# SAC: 0x0001
# LAC: 0x1234
sac0001.lac-lb34.lac-
hb12.lac.sai.mnc999.mcc999.epc.3gppnetwork.org
```

Quando Usado: Áreas de serviço 3G UMTS

5. CGI (Identidade Global da Célula)

Nível de célula 2G

Formato:

```
ci<CI>.lac-lb<LB>.lac-hb<HB>.lac.cgi.mnc<MNC>.mcc<MCC>.
<dns_suffix>
```

Exemplo:

```
# CI: 0x5678
# LAC: 0x1234
ci5678.lac-lb34.lac-hb12.lac.cgi.mnc999.mcc999.epc.3gppnetwork.org
```

Quando Usado: Células 2G GSM

Processamento da Resposta DNS

Formato do Registro NAPTR:

O DNS retorna registros NAPTR apontando para endereços IP de UPF:

```
eci-1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org.
  IN NAPTR 10 50 "a" "x-3gpp-upf:x-s5-gtp:x-s8-gtp" ""
upf1.epc.mnc999.mcc999.3gppnetwork.org.

upf1.epc.mnc999.mcc999.3gppnetwork.org.
  IN A 10.100.1.21
```

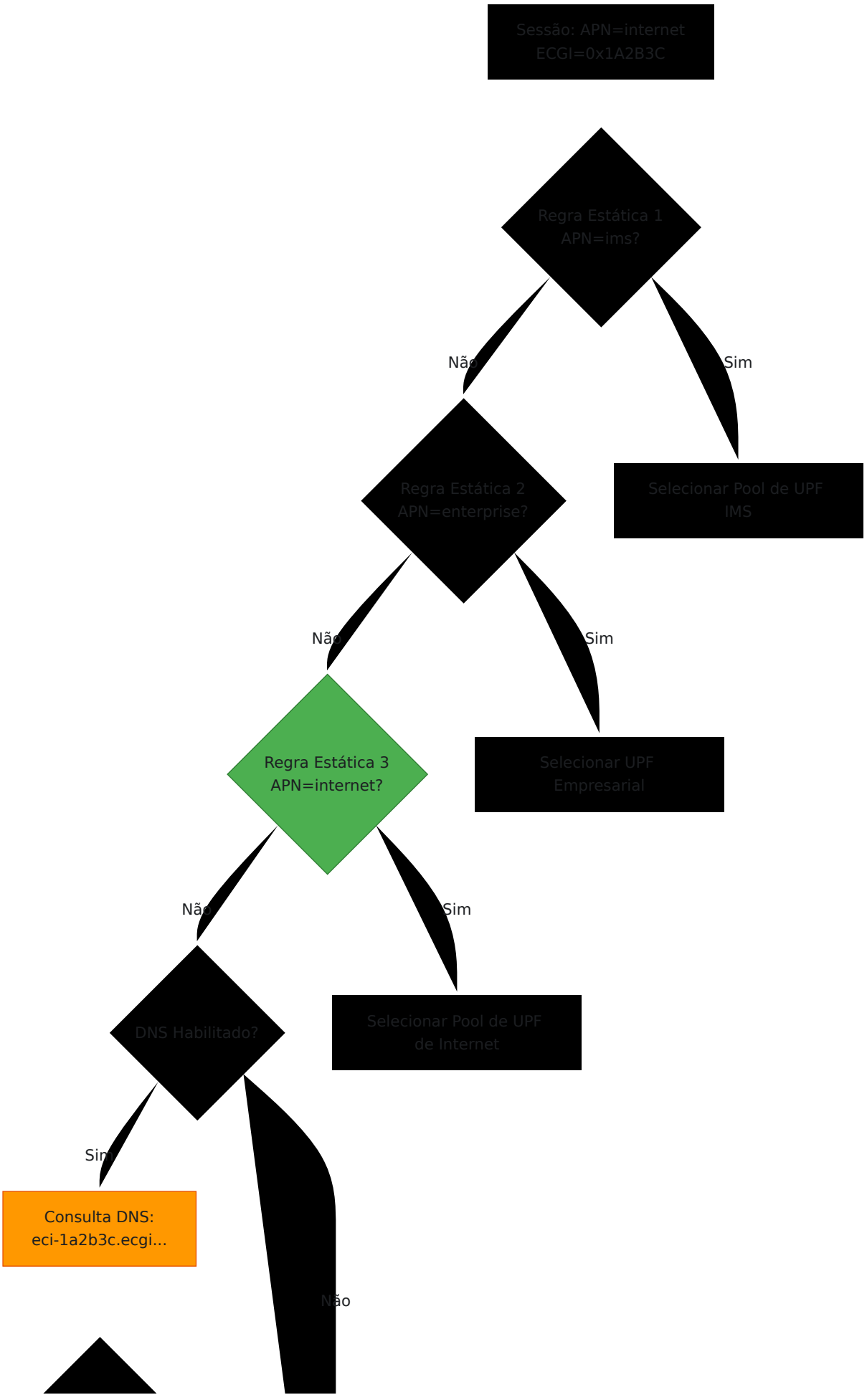
Processamento do PGW-C:

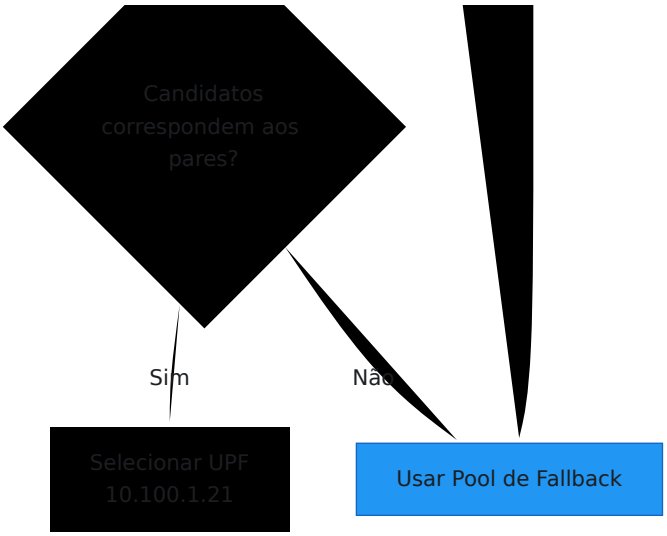
1. Analisa os registros NAPTR para extrair endereços IP de UPF
2. Seleciona o primeiro candidato da resposta DNS
3. Registra dinamicamente se ainda não estiver configurado (ou implementa seleção baseada em carga)

Exemplo:

```
DNS retorna: [10.100.1.21, 10.100.5.99, 10.200.3.50]

Selecionado:    10.100.1.21 (primeiro candidato)
Ação:          Registrar dinamicamente se não estiver na seleção
de upf
```

Exemplo de Prioridade de Seleção


Sessão: APN=internet
ECGI=0x1A2B3C
Regra Estática 1
APN=ims?
Não
Sim
Selecionar Pool de UPF
IMS
Regra Estática 2
APN=enterprise?
Não
Sim
Selecionar UPF
Empresarial
Regra Estática 3
APN=internet?
Não
Sim
Selecionar Pool de UPF
de Internet
DNS Habilitado?
Sim
Não
Consulta DNS:
eci-1a2b3c.ecgi...



Casos de Uso

1. Balanceamento de Carga Geográfica

Cenário: O operador tem UPFs em várias cidades

Configuração DNS:

```
# Célula de Chicago
eci-aaa.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org → UPF-Chicago
(10.1.1.21)

# Célula de Nova York
eci-bbb.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org → UPF-NewYork
(10.2.1.21)

# Célula de Los Angeles
eci-ccc.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org → UPF-
LosAngeles (10.3.1.21)
```

Benefício: Usuários roteados automaticamente para o UPF mais próximo, reduzindo latência e backhaul

2. Computação de Borda

Cenário: UPFs de MEC (Computação de Borda Multi-Acesso) implantados em locais de células

Configuração DNS:

```
# Cada célula aponta para o UPF de borda local
eci-*.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org → UPF de Borda
Local
```

Benefício: Latência ultra-baixa para aplicações de borda

3. Topologia de Rede Dinâmica

Cenário: Endereços de UPF mudam devido a atualizações ou manutenção

Benefício: Atualizar registros DNS sem mudar a configuração do PGW-C

Solução de Problemas de Seleção DNS

Falhas de Consulta DNS

Sintomas:

- Log: "Falha na seleção de UPF DNS: :nxdomain"
- Sessões caem para o pool de fallback

Causas Possíveis:

1. Servidor DNS não configurado corretamente
2. Zona DNS não populada para IDs de célula
3. ULI não presente na mensagem GTP-C

Resolução:

```
# Testar consulta DNS manualmente
dig eci-1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org NAPTR

# Verifique os logs do PGW-C para consultas DNS
grep "DNS UPF selection: querying" /var/log/pgw_c.log

# Verifique se ULI está presente na sessão
# Verifique o campo "uli" no estado da sessão
```

DNS Retorna UPF Desconhecido

Comportamento:

- O DNS retorna um UPF candidato que não está na `upf_selection`
- O sistema tenta registrar dinamicamente
- Se a associação PFCP for bem-sucedida, o UPF é usado para a sessão
- Se a associação PFCP falhar, cai para o pool de fallback

Exemplo:

```
DNS retorna: [10.99.1.50]
upf_selection: [10.100.1.21, 10.100.1.22]

Ação: Registrar dinamicamente 10.99.1.50
  - Enviar Configuração de Associação PFCP
  - Se sucesso: Usar para sessão
  - Se tempo limite: Cair para o pool de fallback
```

Opções de Resolução:

1. Pré-configurar na `upf_selection` para monitoramento imediato:

```
upf_selection: %{
  fallback_pool: [
    %{remote_ip_address: "10.99.1.50", remote_port: 8805, weight:
100}
  ]
}
```

2. Atualizar DNS para retornar IPs de UPF pré-configurados

3. Permitir registro dinâmico (recomendado para cenários de MEC/borda)

Tempo Limite de Consulta

Sintomas:

- Log: "Seleção de UPF DNS: tempo limite de consulta"
- Sessões demoram mais para serem estabelecidas

Resolução:

```
upf_selection: %{
  dns_timeout_ms: 10000  # Aumentar o tempo limite para 10
segundos
}
```

Monitoramento da Seleção DNS

Métricas:

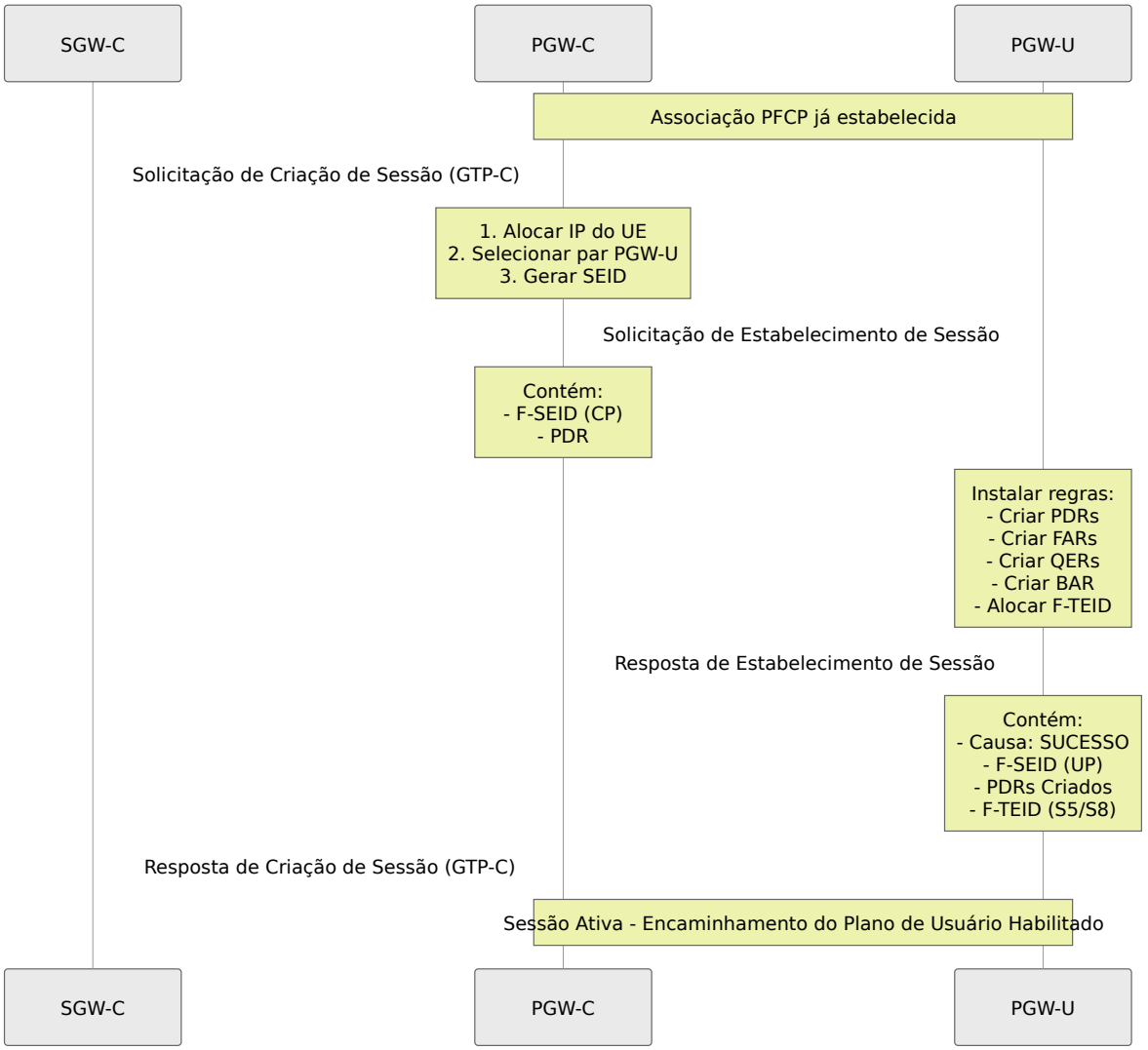
```
# Taxa de sucesso da consulta DNS
rate(upf_selection_dns_success_total[5m]) /
rate(upf_selection_dns_attempts_total[5m])

# Latência da consulta DNS
histogram_quantile(0.95,
rate(upf_selection_dns_duration_seconds_bucket[5m]))

# Uso de fallback (indica problemas de DNS)
rate(upf_selection_fallback_used_total[5m])
```

Exemplos de Logs:

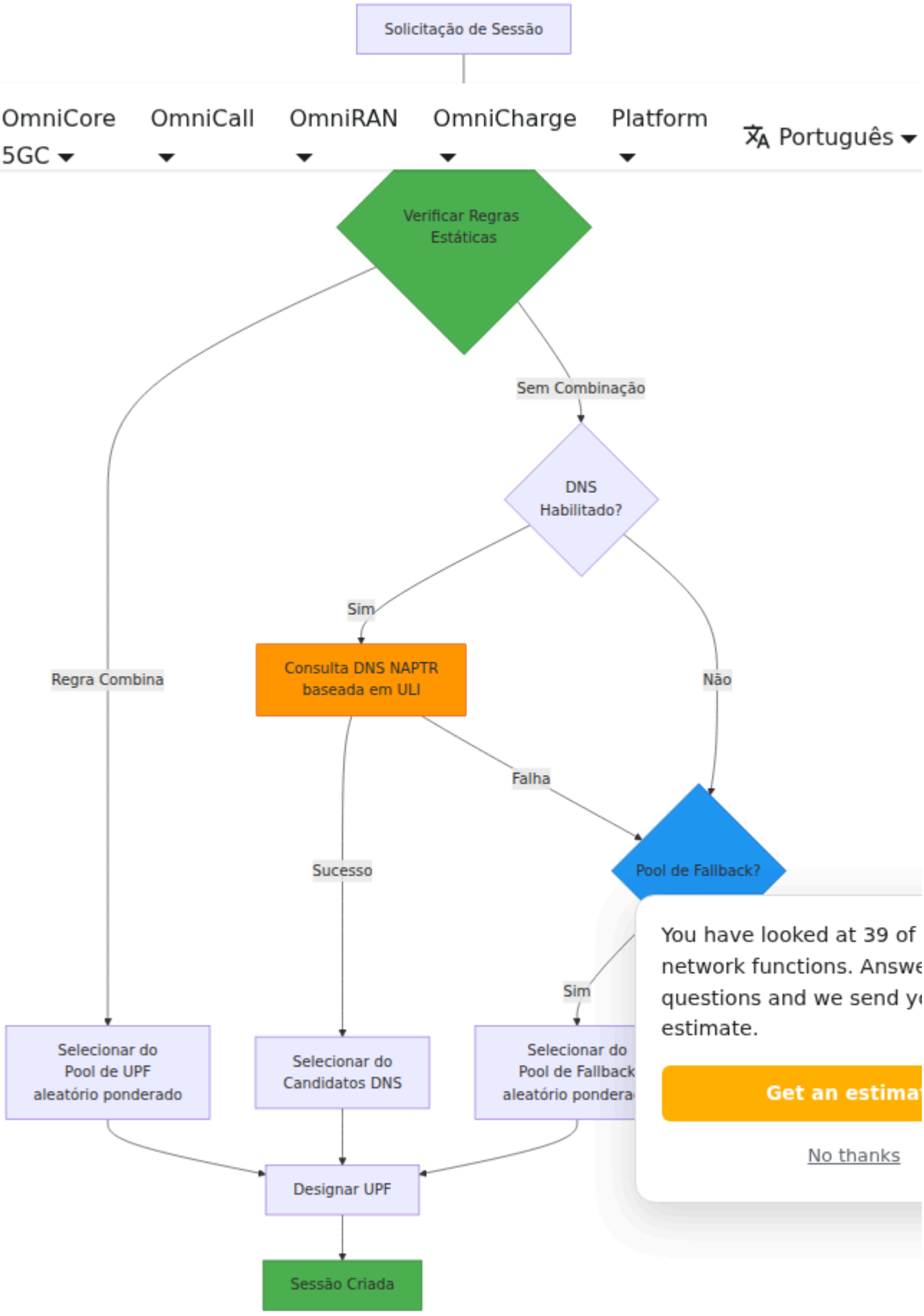
```
[debug] Seleção de UPF DNS: consultando eci-
1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org
[debug] Seleção de UPF DNS: obteve 2 candidatos do DNS
[info]  Seleção de UPF DNS: selecionado 10.100.1.21
```

Fluxos de Mensagens

Fluxo Completo de Estabelecimento de Sessão

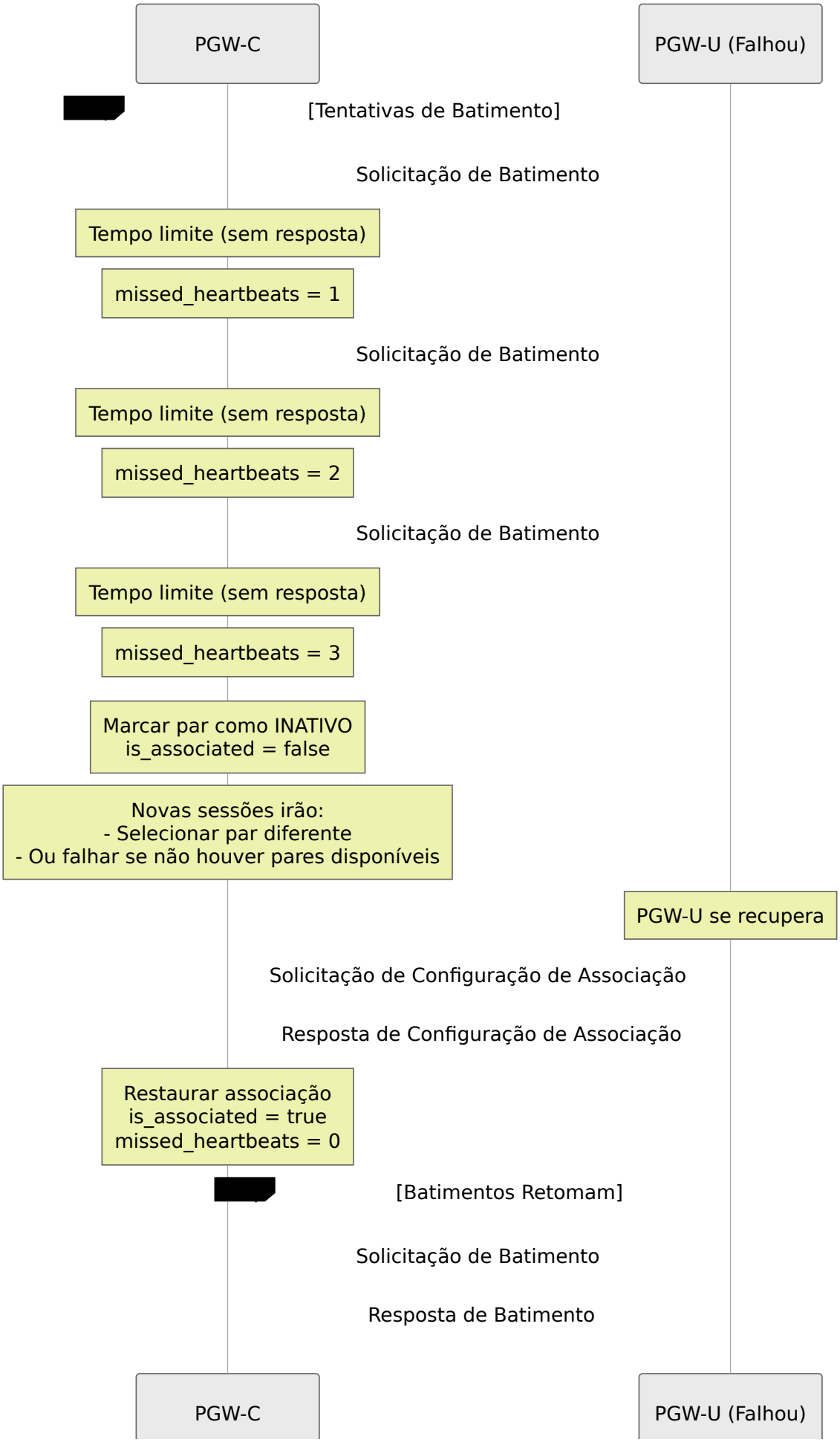


Fluxo de Modificação de Sessão



You have looked at 39 of
network functions. Answe
questions and we send y
estimate.
Get an estima
No thanks

Recuperação de Falha de Batimento


PGW-C
PGW-U (Falhou)
[Tentativas de Batimento]
Solicitação de Batimento
Tempo limite (sem resposta)
missed_heartbeats = 1
Solicitação de Batimento
Tempo limite (sem resposta)
missed_heartbeats = 2
Solicitação de Batimento
Tempo limite (sem resposta)
missed_heartbeats = 3
Marcar par como INATIVO
is_associated = false
Novas sessões irão:
- Selecionar par diferente
- Ou falhar se não houver pares disponíveis
PGW-U se recupera
Solicitação de Configuração de Associação
Resposta de Configuração de Associação
Restaurar associação
is_associated = true
missed_heartbeats = 0
[Batimentos Retomam]
Solicitação de Batimento
Resposta de Batimento
PGW-C
PGW-U (Falhou)

Solução de Problemas

Problemas Comuns

1. Configuração de Associação Falha

Sintomas:

- Mensagem de log: "Falha na Configuração da Associação PFCP"
- Sem resposta à Solicitação de Configuração de Associação

Causas Possíveis:

- PGW-U não acessível (problema de rede)
- PGW-U não está em execução
- Firewall bloqueando a porta UDP 8805
- Endereço `remote_ip_address` incorreto na configuração

Resolução:

```
# Testar conectividade
ping <pgw_u_ip_address>

# Testar porta UDP
nc -u -v <pgw_u_ip_address> 8805

# Verificar firewall
iptables -L -n | grep 8805
```

2. Batimentos Falhando

Sintomas:

- Log: "Falhas consecutivas de batimento: 3"
- Associação marcada como inativa

Causas Possíveis:

- Latência de rede ou perda de pacotes
- PGW-U sobrecarregado
- Intervalo de batimento muito agressivo

Resolução:

O período de batimento é fixo em 5 segundos com um limite de falha de 3 batimentos consecutivos perdidos.

3. Estabelecimento de Sessão Falha

Sintomas:

- Resposta de Criação de Sessão com código de erro
- Log: "Falha no Estabelecimento de Sessão PFCP"

Causas Possíveis:

- Nenhum par PGW-U disponível
- Exaustão de recursos do PGW-U
- Configuração de regra inválida

Verifique:

1. Verifique se pelo menos um par tem `is_associated = true`
2. Verifique os logs do PGW-U para erros
3. Verifique a unicidade do SEID

4. Erros de SEID Duplicados

Sintomas:

- Resposta de Estabelecimento de Sessão: Causa "Contexto de sessão não encontrado"

Causa:

- Colisão de SEID (muito raro)

- Reinicialização do PGW-U sem conhecimento do PGW-C

Resolução:

- Reinicie a associação PFCP (dispara um novo carimbo de tempo de recuperação)
- O PGW-C detectará a reinicialização do PGW-U e limpará as sessões antigas

Monitoramento da Saúde do PFCP

Métricas a Monitorar:

```
# Status da associação do par PFCP
pfcp_peer_associated{peer="PGW-U Primário"} 1

# Sessões PFCP ativas
seid_registry_count 150

# Taxas de mensagens PFCP
rate(sxb_inbound_messages_total[5m])

# Erros PFCP
rate(sxb_inbound_errors_total[5m])

# Falhas de batimento
pfcp_consecutive_heartbeat_failures{peer="PGW-U Primário"} 0
```

Exemplos de Alertas:

```yaml
# Alerta sobre associação inativa
- alert: PFCPAssociationDown
  expr: pfcp_peer_associated == 0
  for: 1m
  annotations:
    summary: "Par PFCP {{ $labels.peer }} está inativo"

# Alerta sobre altas falhas de estabelecimento de sessão
- alert: PFCPSessionEstablishmentFailureHigh
  expr:
rate(sxb_inbound_errors_total{message_type="session_establishment_res
[5m]) > 0.1
  for: 5m
  annotations:
    summary: "Alta taxa de falhas de estabelecimento de sessão PFCP"
```

Monitoramento PFCP

A associação PFCP/Sxb e o status do par UPF são monitorados através da API REST OAM. A API é servida sobre HTTPS/TLS na porta `8443` com todas as rotas sob o prefixo `/api` documentação interativa (Swagger UI) está disponível em `https://<omnipgw-ip>:8443/api/docs`.

Status do UPF / Par PFCP

```
GET /api/upf            # todos os pares UPF / PFCP + status
GET /api/upf/<ip>       # um UPF por IP
```

Propósito: Monitorar o status da associação PFCP com todos os pares PGW-U configurados.

Exemplo:

```
curl -k https://<omnipgw-ip>:8443/api/upf
curl -k https://<omnipgw-ip>:8443/api/upf/10.0.0.1
```

Informações por Par retornadas para cada par PGW-U configurado:

Campo	Descrição
Nome do Par	Nome amigável da configuração
Endereço IP	IP remoto do PGW-U
Status da Associação	Associado ou Não Associado
ID do Nó	Identificador do Nó PFCP
Carimbo de Tempo de Recuperação	Último horário de reinicialização do par
Período de Batimento	Intervalo de batimento configurado
Batimentos Perdidos Consecutivos	Contagem atual de falhas
Recursos de Função UP	Capacidades anunciadas pelo PGW-U

`GET /api/upf/<ip>` retorna o estado completo de um único par, incluindo o bitmap de recursos da função UP e os carimbos de tempo de associação.

Inspeção de Sessão PFCP

As sessões PFCP ativas são visíveis através do detalhe da sessão por assinante:

```
GET /api/sessions/<imsi>
```

O detalhe da sessão inclui o par PGW-U associado e as regras de encaminhamento programadas (PDRs, FARs, QERs, BARs) e F-SEIDs para a sessão.

Casos de Uso Operacionais

Monitorar a Saúde da Associação PFCP:

```
1. GET /api/upf
2. Verifique se todos os pares mostram "Associado"
3. Verifique se a contagem de batimentos perdidos = 0
4. Se um par mostrar "Não Associado":
   - Verifique a acessibilidade do IP do par
   - Verifique se o par está em execução
   - Verifique o firewall (UDP 8805)
```

Solução de Problemas de Falhas de Estabelecimento de Sessão:

```
1. A sessão do usuário falha ao estabelecer
2. GET /api/sessions/<imsi> - a sessão existe?
3. Inspecione o estado PFCP da sessão - sessão PFCP criada?
4. Se não houver sessão PFCP:
   - GET /api/upf - algum par está associado?
   - Verifique os logs para erros PFCP
5. Se a sessão PFCP existir:
   - Inspecione PDRs/FARs para verificar regras programadas
   - O problema é provavelmente a jusante (PGW-U ou rede)
```

Verificar Distribuição de Carga do Par:

```
1. Com múltiplos pares PGW-U configurados
2. GET /api/upf para revisar o estado de cada par
3. Verifique se as sessões estão distribuídas entre os pares
4. Identifique se um par tem carga desproporcional
```

Detectar Falhas de Par:

- `GET /api/upf` relata o estado "Não Associado" por par
- O contador de batimentos perdidos mostra degradação antes da falha total
- Configure alertas de monitoramento com base nos dados da API OAM

Vantagens:

- **Monitoramento Scriptável** - Consulte `GET /api/upf` a partir de ferramentas ou alertas
- **Status Explícito** - Associado / não associado por par
- **Tendências de Saúde do Par** - Contagem de batimentos perdidos mostra aviso antecipado
- **Inspeção em Nível de Sessão** - Veja exatamente PDRs/FARs/QERs programados

---

Documentação Relacionada

Configuração

- **Guia de Configuração** - Seleção de UPF, monitoramento de saúde, configuração PFCP
- **Gerenciamento de Sessão** - Ciclo de vida da sessão PDN, estabelecimento de bearer

Cobrança e Monitoramento

- **Interface Diameter Gx** - Regras PCC que impulsionam a aplicação de QoS PFCP
- **Interface Diameter Gy** - Gerenciamento de cotas de cobrança online via URRs
- **Formato de CDR de Dados** - Geração de CDR a partir de relatórios de uso PFCP
- **Guia de Monitoramento** - Métricas PFCP, rastreamento de sessões, alertas de saúde do UPF

Interfaces de Rede

- **Interface S5/S8** - Gerenciamento de bearer do plano de controle
- **Alocação de IP do UE** - Atribuição de endereço do UE via PFCP

Voltar ao Guia de Operações

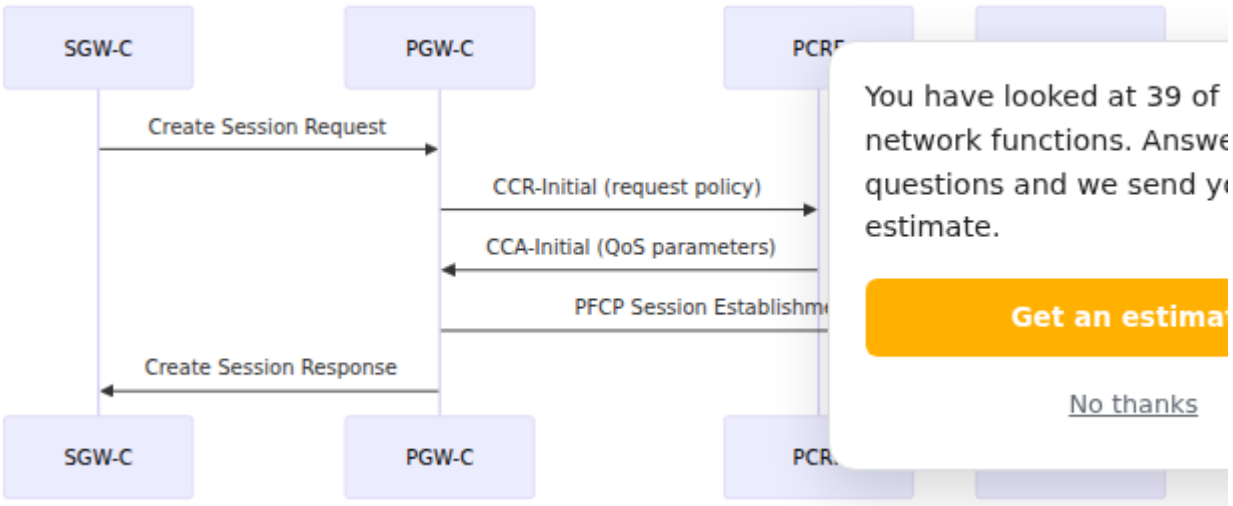
QoS & Gerenciamento de Bearer

Visão Geral

O PGW-C implementa um sistema de gerenciamento de bearer e QoS orientado por políticas que coordena três interfaces principais:

- **Gx (Diameter)** - Recebe decisões de políticas e parâmetros de QoS do PCRF
- **S5/S8 (GTP-C)** - Gerencia contextos de bearer com SGW-C
- **Sxb (PFCP)** - Programa regras de aplicação de QoS no PGW-U

Fluxo da Arquitetura



Conceitos Chave

- **Sessão**: Contém informações do UE, mapa de bearer, mapas de PDR/FAR/QER/BAR e AMBR
- **Contexto de Bearer**: Liga EBI (EPS Bearer ID) a PDRs, FARs e QERs específicos

- **QER (Regra de Aplicação de QoS)**: Impõe limites de MBR/GBR e status de portão no plano do usuário
- **Bearer Padrão**: Sempre criado com a sessão PDN, fornece conectividade básica
- **Bearer Dedicado**: Criado dinamicamente com base na política do PCRF, fornece garantias específicas de QoS

Configuração

Importante: Política de QoS Dinâmica

Todos os parâmetros de QoS são recebidos dinamicamente do PCRF via interface Diameter Gx e definidos no PCRF (Veja OmniHSS para mais informações).

Os operadores configuram a **conexão PCRF** em `config/runtime.exs`:

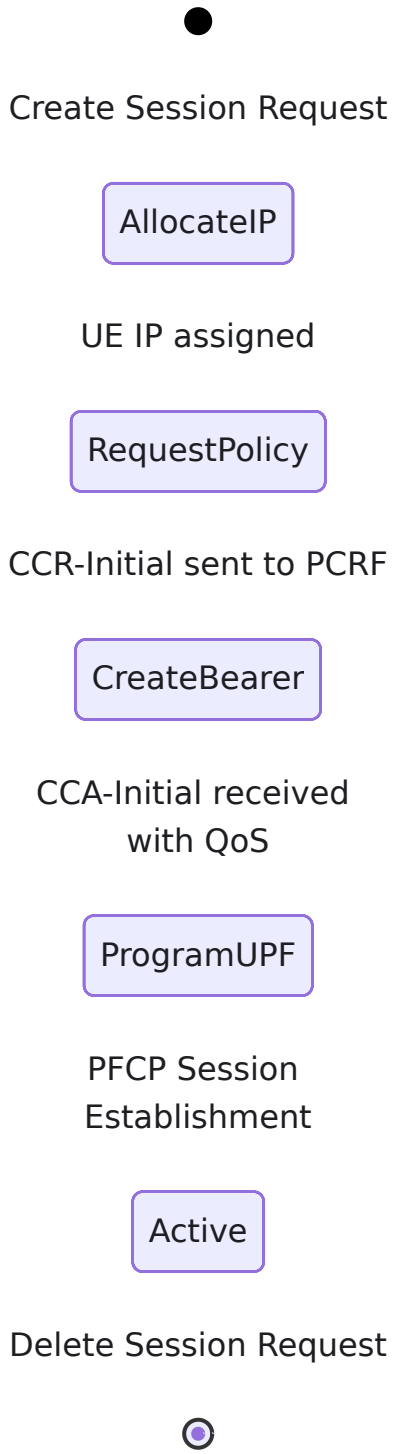
```
config :pgw_c,
  diameter: %{
    listen_ip: "0.0.0.0",
    host: "omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org",
    realm: "epc.mnc999.mcc999.3gppnetwork.org",
    peer_list: [
      %{
        host: "pcrf.epc.mnc999.mcc999.3gppnetwork.org",
        realm: "epc.mnc999.mcc999.3gppnetwork.org",
        ip: "192.168.1.100",
        initiate_connection: true
      }
    ]
  }
```

Políticas de QoS, regras de cobrança e limites de largura de banda são configurados no PCRF, não nos arquivos de configuração do PGW-C.

Ciclo de Vida do Bearer

Criação do Bearer Padrão

O bearer padrão é criado durante o estabelecimento da sessão PDN:



Fluxo de Trabalho:

1. SGW-C envia Create Session Request
2. PGW-C aloca o endereço IP do UE a partir do pool configurado
3. PGW-C envia CCR-Initial ao PCRF com IMSI, APN, endereço IP
4. PCRF responde com CCA-Initial contendo parâmetros de QoS:
    - Default-EPS-Bearer-QoS (QCI, ARP)
    - QoS-Information (ajustes de AMBR)
5. PGW-C cria contexto de bearer com:
    - IDs fixos: Downlink PDR=1, Uplink PDR=2, Downlink FAR=1, Uplink FAR=2, QER=1, BAR=1
    - QER programado com MBR do QoS do bearer
6. PGW-C envia PFCP Session Establishment Request ao PGW-U
7. PGW-C envia Create Session Response ao SGW-C

Características do bearer padrão:

- Sempre existe durante a vida útil da sessão PDN
- Normalmente usa QCI 5 ou QCI 9 (não-GBR)
- EBI rastreado no estado da sessão
- Não pode ser excluído independentemente (excluí-lo termina a sessão)

Criação do Bearer Dedicado

Bearers dedicados são criados dinamicamente com base na política do PCRF:

Gatilho: Re-Auth Request (RAR) do PCRF com Charging-Rule-Install

Fluxo de Trabalho:

1. PCRF envia RAR com Charging-Rule-Definition contendo:
    - Charging-Rule-Name (identificador da regra de política)
    - Flow-Information (filtros de pacotes)
    - QoS-Information (QCI, MBR, GBR, ARP)
    - Precedence (prioridade de correspondência da regra)
2. PGW-C traduz a regra dinâmica para entidades PFCP:
    - Cada entrada de Flow-Information → novo PDR com SDF Filter
    - QoS-Information → novo QER com aplicação de MBR/GBR

    - Flow-Description → regras de correspondência de IP 5-tuple
3. PGW-C envia PFCP Session Modification Request para adicionar PDRs/FARs/QERs
4. PGW-C inicia Create Bearer Request ao SGW-C
5. SGW-C responde com Create Bearer Response confirmando o estabelecimento

Exemplo de Charging-Rule-Definition:

```
Charging-Rule-Name: "video_streaming"
Flow-Information:
  - Flow-Description: "permit in ip from any to 10.0.0.1 5000-
6000"
    Flow-Direction: 1 (downlink)
QoS-Information:
  QoS-Class-Identifier: 7
  Max-Requested-Bandwidth-UL: 5000000 (5 Mbps)
  Max-Requested-Bandwidth-DL: 10000000 (10 Mbps)
  Guaranteed-Bitrate-UL: 1000000 (1 Mbps)
  Guaranteed-Bitrate-DL: 2000000 (2 Mbps)
Precedence: 100
Flow-Status: 2 (ENABLED)
```

Regras Ligadas ao Bearer Padrão

Uma regra dinâmica cuja QCI e ARP correspondem ao bearer padrão se liga ao bearer padrão (TS 29.212 §4.5.2). Nenhum bearer dedicado é criado.

Quando a regra corresponde a todo o tráfego (cada Flow-Description é `permit out ip from any to any`), seu MBR limita o APN-AMBR no próprio QER do bearer padrão:

- Na instalação (CCA-Initial ou RAR), o MBR do QER padrão se torna o menor entre o APN-AMBR e o MBR da regra, por direção. Uma regra do CCA-Initial já está no QER que o Estabelecimento da Sessão carrega.
- Na remoção da Charging-Rule, o QER volta ao APN-AMBR.
- Uma alteração posterior do APN-AMBR pelo PCRF mantém o limite enquanto a regra estiver instalada.

É assim que um throttle de uso justo funciona com Sy: o PCRF instala uma regra como `throttle` (QCI 9, MBR 1 Mbit/s) quando a cota é utilizada, e a remove após uma recarga.

Uma regra de bearer padrão que corresponde apenas a parte do tráfego, ou que controla uma direção com Flow-Status, precisaria de PDRs próprios e não é suportada. Um RAR que carrega uma é respondido com DIAMETER_UNABLE_TO_COMPLY (5012). No CCA-Initial, é registrado e não aplicado.

Modificação do Bearer

A QoS do bearer pode ser modificada via:

- **Gx RAR** com Charging-Rule-Definition atualizada
- **PFCP Session Modification** para atualizar QERs existentes (alterar bitrates), FARs (alterar encaminhamento) ou PDRs (alterar filtros de pacotes)

Exclusão do Bearer

Gatilhos:

- **Delete Session Request** (iniciado pelo SGW) - Exclui o bearer padrão e termina a sessão
- **Re-Auth Request com Charging-Rule-Remove** (iniciado pelo PCRF) - Exclui o bearer dedicado

Fluxo de Trabalho:

1. Remover bearer do estado da sessão
2. Remover PDRs/FARs/QERs associados
3. Enviar Delete Bearer Request ao SGW-C (se iniciado pelo PCRF)
4. Enviar PFCP Session Modification (remover regras) ou Session Deletion (se bearer padrão)

Parâmetros de QoS

QCI (Identificador de Classe de QoS)

Fonte: PCRF via Gx `QoS-Class-Identifier` AVP

Valores Padrão:

- **QCI 1**: Voz Conversacional (GBR, 100ms de orçamento de atraso)
- **QCI 2**: Vídeo Conversacional (GBR, 150ms de orçamento de atraso)
- **QCI 3**: Jogos em Tempo Real (GBR, 50ms de orçamento de atraso)
- **QCI 4**: Vídeo Não Conversacional (GBR, 300ms de orçamento de atraso)
- **QCI 5**: Sinalização IMS (não-GBR, 100ms de orçamento de atraso) - **Padrão para bearer padrão**
- **QCI 6**: Vídeo (baseado em TCP), Streaming Ao Vivo (não-GBR, 300ms de orçamento de atraso)
- **QCI 7**: Voz, Jogos Interativos (não-GBR, 100ms de orçamento de atraso)
- **QCI 8**: Vídeo (baseado em TCP), por exemplo, YouTube (não-GBR, 300ms de orçamento de atraso)
- **QCI 9**: Internet Padrão (não-GBR, 300ms de orçamento de atraso)

Nota do Operador:

- O QCI é recebido do PCRF e sinalizado ao SGW-C no Bearer-Level-QoS IE
- O PGW-C não aplica diretamente o comportamento do QCI - a aplicação real é via MBR/GBR nos QERs
- Valores de QCI mais baixos geralmente indicam maior prioridade
- O QCI determina o tratamento de encaminhamento de pacotes e a prioridade de agendamento

ARP (Prioridade de Alocação e Retenção)

Fonte: PCRF via AVP agrupado `Allocation-Retention-Priority`

Componentes:

- **Priority-Level**: 1 (maior prioridade) a 15 (menor prioridade)
- **Pre-emption-Capability**: Este bearer pode preemptar bearers de menor prioridade?
  - 0 = HABILITADO (pode preemptar outros)
  - 1 = DESABILITADO (não pode preemptar)
- **Pre-emption-Vulnerability**: Este bearer pode ser preemptado por bearers de maior prioridade?
  - 0 = HABILITADO (pode ser preemptado)
  - 1 = DESABILITADO (não pode ser preemptado)

Valores Padrão:

- Priority-Level: 1
- Pre-emption-Capability: HABILITADO (0)
- Pre-emption-Vulnerability: DESABILITADO (1)

Nota do Operador:

- O ARP é sinalizado ao SGW-C e, em última instância, ao eNodeB
- **Não é aplicado pelo PGW-C** - a aplicação é tipicamente no eNodeB durante o controle de admissão de rádio
- Usado durante a congestão da rede para determinar quais bearers admitir ou descartar
- Crítico para serviços de emergência (nível de prioridade 1) e serviços de alto valor

MBR (Taxa Máxima de Bits)

Fonte: PCRF via AVPs `Max-Requested-Bandwidth-UL` e `Max-Requested-Bandwidth-DL`

Formato: Bytes por segundo (convertido para kbps internamente: `bytes / 1000`)

Aplicado a: Todos os bearers (padrão e dedicado)

Como funciona:

- O PGW-C cria QER com `mbr: %Bitrate{ul: kbps_ul, dl: kbps_dl}`
- QER enviado ao PGW-U via PFCP
- **PGW-U aplica limitação de taxa** (policiamento de tráfego)
- Tráfego excedente acima do MBR é descartado

Exemplo:

```
Max-Requested-Bandwidth-UL: 5000000 (5 Mbps)
Max-Requested-Bandwidth-DL: 10000000 (10 Mbps)

→ QER criado com mbr: {ul: 5000, dl: 10000} kbps
→ PGW-U descarta pacotes de uplink que excedem 5 Mbps
→ PGW-U descarta pacotes de downlink que excedem 10 Mbps
```

GBR (Taxa Garantida de Bits)

Fonte: PCRF via AVPs `Guaranteed-Bitrate-UL` e `Guaranteed-Bitrate-DL`

Formato: Bytes por segundo (convertido para kbps)

Aplicado a: Apenas bearers dedicados (bearers GBR)

Como funciona:

- Se GBR for especificado na Charging-Rule-Definition, o bearer é **do tipo GBR**
- O PGW-U aplica a garantia de taxa mínima via QER
- Requer agendamento adequado no eNodeB para reservar recursos de rádio
- Bearers GBR têm controle de admissão - podem ser rejeitados se recursos não estiverem disponíveis

Exemplo:

```
Guaranteed-Bitrate-UL: 1000000 (1 Mbps)
Guaranteed-Bitrate-DL: 2000000 (2 Mbps)

→ QER criado com gbr: {ul: 1000, dl: 2000} kbps
→ A rede garante pelo menos 1 Mbps de uplink e 2 Mbps de downlink
→ Usado para VoIP, chamadas de vídeo, streaming ao vivo
```

Nota do Operador:

- GBR requer planejamento de capacidade de rede suficiente
- Superlotar recursos GBR leva a falhas de admissão
- Monitore o uso de GBR via contagens de sessão e métricas de bearer

AMBR (Taxa Máxima Agregada de Bits)

Fonte: PCRF via AVPs `APN-Aggregate-Max-Bitrate-UL` e `APN-Aggregate-Max-Bitrate-DL`

Escopo: Aplica-se a **todos os bearers não-GBR** para o APN (não por bearer)

Como funciona:

- AMBR é um limite agregado em todos os bearers não-GBR em uma sessão
- Enviado ao SGW-C na Create Session Response
- A aplicação é tipicamente no eNodeB/SGW
- O PGW-C armazena AMBR no estado da sessão e sinaliza ao SGW-C

Exemplo:

```
APN-Aggregate-Max-Bitrate-UL: 50000000 (50 Mbps)
APN-Aggregate-Max-Bitrate-DL: 100000000 (100 Mbps)

→ Todos os bearers não-GBR combinados não podem exceder 50 Mbps de
uplink / 100 Mbps de downlink
→ Bearers individuais limitados pelo seu próprio MBR
→ AMBR fornece um limite adicional geral por UE/APN
```

Nota do Operador:

- Definido via perfil de assinante no HSS/PCRF
- Usado para aplicar níveis de assinatura (por exemplo, plano de 10 Mbps vs plano de 100 Mbps)
- Não afeta bearers GBR

Status do Fluxo e Controle

Mapeamento de Status do Fluxo (Gx) para Status do Portão (PFCP)

O PCRF controla se o tráfego é permitido via o AVP `Flow-Status` na Charging-Rule-Definition:

Flow-Status (Gx)	Gate-Status (PFCP QER)	Significado
0 = ENABLED-UPLINK	ul: OPEN, dl: CLOSED	Apenas tráfego de uplink permitido
1 = ENABLED-DOWNLINK	ul: CLOSED, dl: OPEN	Apenas tráfego de downlink permitido
2 = ENABLED	ul: OPEN, dl: OPEN	Ambas as direções permitidas
3 = DISABLED	ul: CLOSED, dl: CLOSED	Nenhum tráfego permitido
4 = REMOVED	ul: CLOSED, dl: CLOSED	Bearer sendo excluído

Casos de uso:

- **DISABLED**: Usado para serviços estacionados ou exaustão de crédito (pacotes descartados, mas bearer mantido)

- **ENABLED-UPLINK**: Incomum, mas pode ser usado para serviços apenas de upload
- **ENABLED-DOWNLINK**: Serviços apenas de download ou cenários limitados por crédito
- **ENABLED**: Operação normal

Monitoramento & Observabilidade

Métricas do Prometheus

Métricas de nível de sessão:

```
session_registry_count        # Bearers ativos (pares IMSI, EBI)
address_registry_count        # IPs do UE alocados
charging_id_registry_count    # Sessões de cobrança ativas
```

Métricas da interface Gx:

```
gx_inbound_messages_total{message_type="gx_RAR"}     #
Atualizações de política do PCRF
gx_outbound_messages_total{message_type="gx_CCR"}    #
Solicitações de política ao PCRF
gx_outbound_transaction_duration_bucket              # Latência ao
PCRF
```

Métricas da interface PFCP:

```
sxb_outbound_messages_total{message_type="pfcp_session_establishment_
sxb_outbound_messages_total{message_type="pfcp_session_modification_r
sxb_outbound_transaction_duration_bucket
```

Métricas de criação de bearer:

```
s5s8_inbound_messages_total{message_type="create_session_request"}
# Bearers padrão
s5s8_outbound_messages_total{message_type="create_bearer_request"}
# Bearers dedicados
```

Monitoramento via API OAM

O estado do bearer e da QoS é inspecionado através da API REST OAM (HTTPS/TLS na porta `8443`, rotas sob o prefixo `/api`). A documentação interativa (Swagger UI) está disponível em `https://<omnipgw-ip>:8443/api/docs`.

As informações de bearer e QoS são expostas através dos detalhes da sessão por assinante:

```
GET /api/sessions/<imsi>       # detalhes da sessão incl. bearers,
QCI/5QI, regras de cobrança
```

Exemplo:

```
curl -k https://<omnipgw-ip>:8443/api/sessions/999000123456789
```

Os detalhes da sessão incluem os bearers ativos para a sessão, seus parâmetros de QoS (QCI/5QI, MBR, GBR, AMBR), as regras de cobrança instaladas e o mapa QER programado (status do portão e taxas de bits). Não há um endpoint de bearers dedicados; o estado do bearer é sempre lido da sessão à qual pertence.

Mensagens de Log Chave

```
[debug] Sending Credit Control Request: ...          # CCR para
PCRF
[debug] Handling Credit Control Answer: ...          # CCA do PCRF
(contém QoS)
[debug] Handling Re-Auth Request                     # RAR do PCRF
(mudança de política)
[debug] Sending Session Establishment Request        # PFCP para
PGW-U (programar QERs)
[debug] Sending Session Modification Request         # PFCP para
PGW-U (atualizar QERs)
```

Tarefas Operacionais

Verificar QoS Aplicada à Sessão

1. `GET /api/sessions/<imsi>` (por exemplo, `curl -k https://<omnipgw-ip>:8443/api/sessions/999000123456789`)
2. Localize a sessão para o assinante na resposta
3. Inspecione os detalhes da sessão
4. Verifique a seção **qer_map**:

   ```
   qer_id: 1
   gate_status: {ul: OPEN, dl: OPEN}
   mbr: {ul: 50000, dl: 100000}  # kbps
   gbr: {ul: 10000, dl: 20000}   # kbps (ou nil para não-GBR)
   ```

5. Verifique se os valores correspondem à política esperada do PCRF

Solucionar Problemas de QoS Ausente

Sintoma: Sessão criada, mas QoS não aplicada

Etapas:

1. **Verifique a conectividade do PCRF:**

- Verifique o status do peer PCRF = "conectado" (veja as métricas Diameter/Gx e o estado do peer)
- Se desconectado, verifique a conectividade da rede e a configuração do Diameter

2. **Verifique a troca CCR/CCA:**

   - Pesquise os logs do OmniPGW por "Credit Control Answer"
   - Verifique se o AVP `QoS-Information` está presente no log CCA
   - Verifique se há erros no CCA (Result-Code deve ser 2001 = SUCESSO)

3. **Verifique a programação PFCP:**

   - Pesquise logs por "PFCP Session Establishment Request"
   - Verifique se o QER está incluído na mensagem
   - Verifique os logs do PGW-U para erros de processamento PFCP

4. **Verifique a configuração da política do PCRF:**

   - Verifique o perfil do assinante no PCRF
   - Confirme se as regras de política específicas do APN existem
   - Verifique os logs do PCRF para erros de avaliação de política

Monitorar Taxa de Criação de Bearer

Consultas Prometheus:

```
# Taxa de criação de bearer padrão (sessões/segundo)
rate(s5s8_inbound_messages_total{message_type="create_session_request
[5m])

# Taxa de criação de bearer dedicado
rate(s5s8_outbound_messages_total{message_type="create_bearer_request
[5m])

# Taxa de atualização de política do PCRF
rate(gx_inbound_messages_total{message_type="gx_RAR"}[5m])
```

Planejamento de Capacidade

Métricas chave a serem monitoradas:

```
# Utilização de endereços IP do UE (porcentagem)
(address_registry_count / <configured_pool_size>) * 100

# Contagem de bearers ativos
session_registry_count

# Latência da consulta do PCRF (P95)
histogram_quantile(0.95, gx_outbound_transaction_duration_bucket)
```

Limites de capacidade:

- Tamanho do pool de endereços: configurado em `config/runtime.exs` sob `ue.subnet_map`
- Espaço TEID: 32 bits (4 bilhões de identificadores únicos, gerenciados automaticamente)
- Sessões concorrentes: tipicamente limitadas pelo tamanho do pool de endereços

Diretrizes de planejamento:

- Monitore a utilização de endereços IP - amplie o pool antes de exceder 80%
- Monitore a latência do PCRF - alta latência impacta o tempo de configuração da sessão
- Monitore a taxa de criação de bearers dedicados - indica a complexidade da política

Documentação Relacionada

- Gerenciamento de Sessão - Ciclo de vida da sessão PDN
- Interface Diameter Gx - Detalhes do protocolo de política do PCRF
- Interface PFCP - Programação do plano do usuário
- Guia de Configuração - Configuração do sistema

- Guia de Monitoramento - Métricas e observabilidade

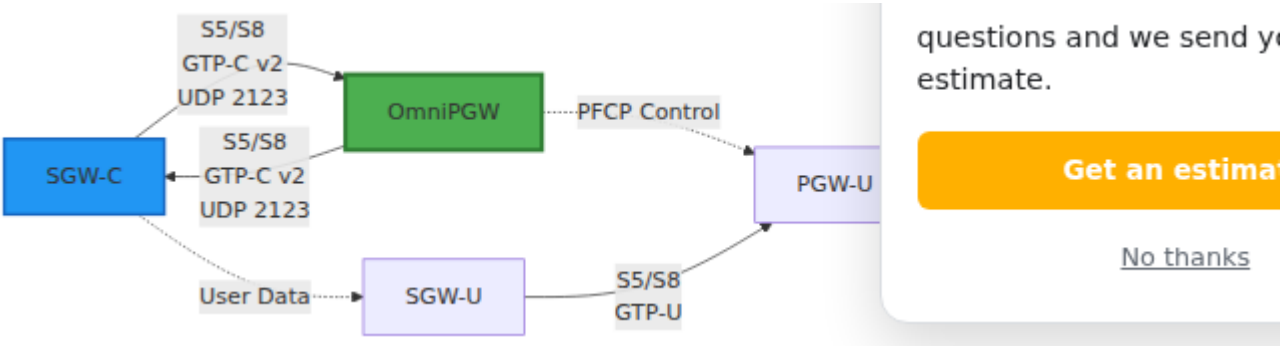
Documentação da Interface S5/S8

Comunicação GTP-C com SGW-C

OmniPGW da Omnitouch Network Services

Visão Geral

A **interface S5/S8** conecta o OmniPGW ao SGW-C (Serving Gateway Control plane) usando o protocolo **GTP-C v2** (GPRS Tunnelling Protocol - Control plane). Esta interface gerencia o sinalizador de gerenciamento de sessão entre os gateways.



Detalhes do Protocolo

GTP-C Versão 2

- **Protocolo:** GTP-C v2 (3GPP TS 29.274)
- **Transporte:** UDP
- **Porta:** 2123 (padrão)
- **Tipo de Interface:** Plano de Controle

TEID (Identificador de Ponto de Extremidade do Túnel)

Cada sessão possui um **TEID** único para roteamento de mensagens:

- **TEID Local** - Alocado pelo OmniPGW para mensagens de entrada
- **TEID Remoto** - Alocado pelo SGW-C para mensagens de saída

```
Fluxo da Mensagem:
  SGW-C → OmniPGW: TEID de Destino = TEID Local do OmniPGW
  OmniPGW → SGW-C: TEID de Destino = TEID Remoto do SGW-C
```

Configuração

Configuração Básica

```
# config/runtime.exs
config :pgw_c,
  s5s8: %{
    # Endereço IPv4 local para a interface S5/S8
    local_ipv4_address: "10.0.0.20",

    # Opcional: Endereço IPv6 local
    local_ipv6_address: nil,

    # Opcional: Substituir porta padrão
    local_port: 2123,

    # Tempo limite de solicitação GTP-C em milissegundos (padrão:
500ms)
    # Tempo limite por tentativa ao aguardar respostas GTP-C
(Criar Bearer, Deletar Bearer, etc.)
    request_timeout_ms: 500,

    # Número de tentativas de reenvio para solicitações GTP-C
(padrão: 3)
    # Tempo total máximo de espera = request_timeout_ms *
request_attempts
    # Exemplo: 500ms * 3 tentativas = 1500ms (1,5 segundos) no
total
    request_attempts: 3
  }
```

Configuração de Tempo Limite

A interface S5/S8 usa tempos limites configuráveis para transações de solicitação/resposta GTP-C.

Parâmetros:

- **`request_timeout_ms`** - Tempo limite em milissegundos por tentativa de reenvio (padrão: 500ms)

- `request_attempts` - Número de tentativas de reenvio antes de desistir (padrão: 3)

Tempo Total de Espera: `request_timeout_ms × request_attempts`

Comportamento padrão: 500ms × 3 tentativas = **1,5 segundos de espera máxima total**

Diretrizes de Ajuste:

Latência da Rede	`request_timeout_ms` Recomendado	Tempo Total de Espera
Baixa latência (<50ms)	200-300ms	600-900ms (3 tentativas)
Normal (50-150ms)	500ms (padrão)	1,5s (3 tentativas)
Alta latência (>150ms)	1000-2000ms	3-6s (3 tentativas)
Instável/satélite	2000-3000ms	6-9s (3 tentativas)

Exemplo - Rede de Alta Latência:

```
s5s8: %{
  local_ipv4_address: "10.0.0.20",
  request_timeout_ms: 1500,  # 1,5 segundos por tentativa
  request_attempts: 3         # Total: 4,5 segundos no máximo
}
```

Quando ocorre o tempo limite:

- OmniPGW registra erro: `"Tempo limite da solicitação de criação de bearer"`
- Retorna erro ao PCRF (Código de Resultado Diameter: 5012 UNABLE_TO_COMPLY)

- Bearer permanece em armazenamento inicial para limpeza via Charging-Rule-Remove

Requisitos de Rede

Regras de Firewall:

```
# Permitir GTP-C da rede SGW-C
iptables -A INPUT -p udp --dport 2123 -s <sgw_network>/24 -j
ACCEPT

# Permitir GTP-C de saída para SGW-C
iptables -A OUTPUT -p udp --dport 2123 -d <sgw_network>/24 -j
ACCEPT
```

Roteamento:

```
# Garantir rota para a rede SGW-C
ip route add <sgw_network>/24 via <gateway_ip> dev eth0
```

Tipos de Mensagens

A interface S5/S8 gerencia o sinalizador GTP-C para gerenciamento de sessão PDN. Para um ciclo de vida de sessão detalhado e gerenciamento de estado, consulte o Guia de Gerenciamento de Sessão.

Gerenciamento de Sessão

Solicitação de Criação de Sessão

Direção: SGW-C → OmniPGW

Propósito: Estabelecer uma nova conexão PDN

Principais IEs (Elementos de Informação):

Nome IE	Tipo	Descrição
IMSI	Identidade	Identidade Internacional do Assinante Móvel
MSISDN	Identidade	Número de telefone móvel
APN	String	Nome do Ponto de Acesso (por exemplo, "internet")
Tipo de RAT	Enum	Tecnologia de Acesso por Rádio (EUTRAN)
Contexto de Bearer	Agrupado	Informações do bearer padrão
Fuso Horário do UE	Timestamp	Fuso horário do UE
ULI	Agrupado	Informações de Localização do Usuário (TAI, ECGI)
Rede Servidora	PLMN	MCC/MNC da rede servidora

Exemplo:

```
Solicitação de Criação de Sessão
├── IMSI: 310260123456789
├── MSISDN: 14155551234
├── APN: internet
├── Tipo de RAT: EUTRAN (6)
├── Contexto de Bearer
│   ├── EBI: 5
│   ├── QoS do Bearer (QCI 9, ARP, taxas de bits)
│   └── S5/S8 F-TEID (ponto de extremidade do túnel SGW-U)
└── ULI
    ├── TAI: MCC 310, MNC 260, TAC 12345
    └── ECGI: MCC 310, MNC 260, ECI 67890
```

Resposta de Criação de Sessão

Direção: OmniPGW → SGW-C

Propósito: Reconhecer a criação da sessão

Principais IEs:

Nome IE	Tipo	Descrição
Causa	Resultado	Código de sucesso ou erro
Contexto de Bearer	Agrupado	Informações do bearer
Alocação de Endereço PDN	IP	Endereço IP alocado para o UE (veja Alocação de IP do UE)
Restrição de APN	Enum	Restrições de uso do APN
PCO	Opções	Opções de Configuração de Protocolo (veja Configuração de PCO)

Resposta de Sucesso:

```
Resposta de Criação de Sessão
├── Causa: Solicitação aceita (16)
├── Alocação de Endereço PDN
│   └── IPv4: 100.64.1.42
├── Contexto de Bearer
│   ├── EBI: 5
│   ├── Causa: Solicitação aceita
│   └── S5/S8 F-TEID (ponto de extremidade do túnel PGW-U do PFCP)
├── Restrição de APN: Público-1 (1)
└── PCO
    ├── Servidor DNS: 8.8.8.8
    ├── Servidor DNS: 8.8.4.4
    └── MTU do Link: 1400
```

> O layout da Alocação de Endereço PDN segue o tipo de PDN negociado (TS 29.274 §8.14): um endereço IPv4 de 4 bytes; um octeto de comprimento de prefixo mais um prefixo IPv6 de 16 bytes; ou para IPv4v6 um octeto de comprimento de prefixo mais um prefixo IPv6 de 16 bytes mais um IPv4 de 4 bytes. Se um PDN IPv4v6 for solicitado para um APN de emergência (SOS) que não possui pool IPv6, o PGW-C degrada para um PAA apenas IPv4 (APNs regulares são rejeitadas) — veja solicitação IPv4v6 quando o APN não tem pool IPv6.

Solicitação de Exclusão de Sessão

Direção: SGW-C → OmniPGW

Propósito: Encerrar a conexão PDN

Principais IEs:

Nome IE	Descrição
EBI	ID do Bearer EPS a ser excluído
EBI Vinculado	Bearer relacionado (opcional)

Resposta de Exclusão de Sessão

Direção: OmniPGW → SGW-C

Propósito: Reconhecer a exclusão da sessão

Principais IEs:

Nome IE	Descrição
Causa	Código de sucesso ou erro

Gerenciamento de Bearer

Solicitação de Criação de Bearer

Direção: OmniPGW → SGW-C

Propósito: Criar bearer dedicado (iniciado pela política do PCRF)

Acionado por:

- PCRF envia nova regra PCC exigindo bearer dedicado
- OmniPGW solicita ao SGW-C para estabelecer o bearer

Solicitação de Exclusão de Bearer

Direção: OmniPGW → SGW-C ou SGW-C → OmniPGW

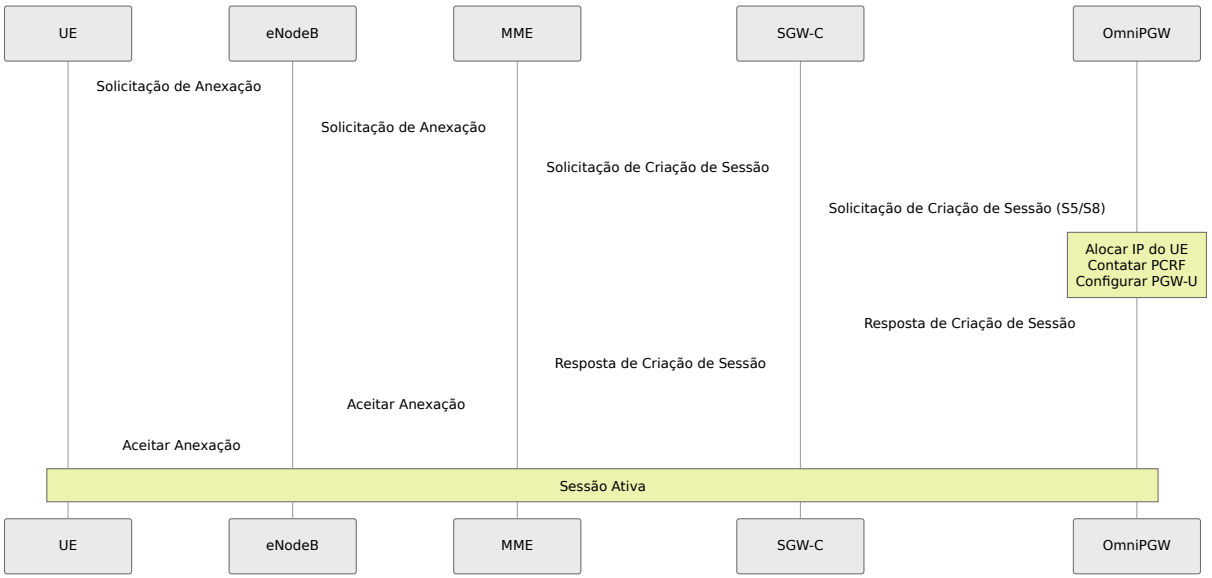
Propósito: Excluir bearer dedicado

Cenários:

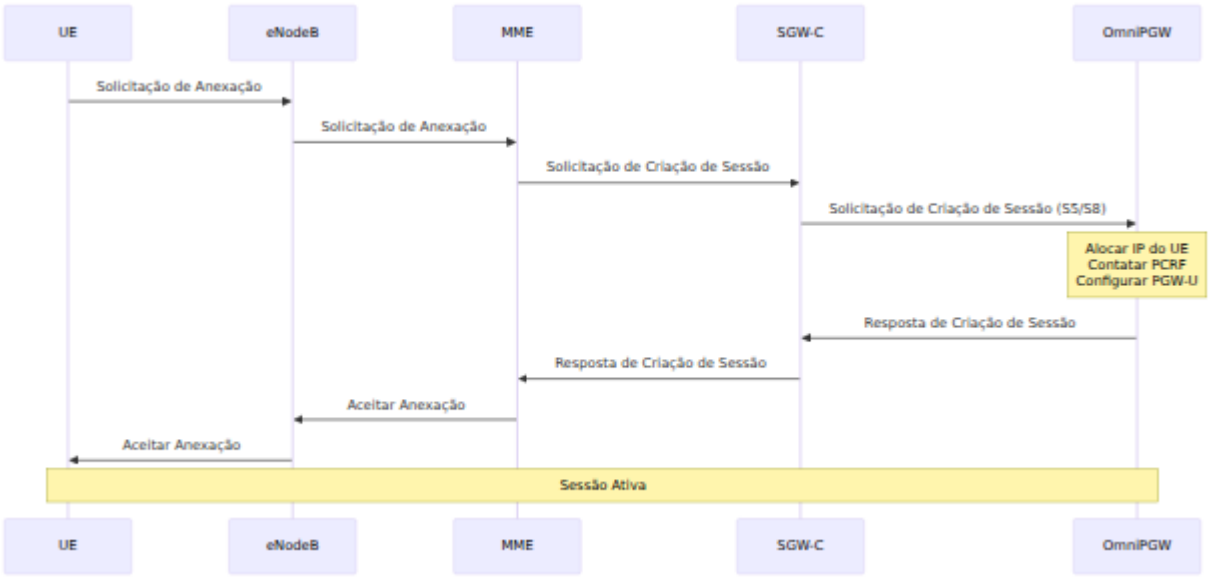
- **Iniciado pelo PGW:** Mudança de política do PCRF remove bearer dedicado
- **Iniciado pelo SGW:** Liberação de recurso de rádio

Fluxos de Mensagens

Estabelecimento de Sessão



Término de Sessão



Códigos de Causa

Sucesso

Código	Nome	Descrição
16	Solicitação aceita	Operação bem-sucedida

Erros (Falhas Permanentes)

Código	Nome	Quando Usado
65	Usuário Desconhecido	PCRF rejeitou (IMSI não encontrado)
66	Nenhum recurso disponível	Pool de IP esgotado
93	Serviço não suportado	APN inválido
94	Erro semântico no TFT	Modelo de fluxo de tráfego inválido

Erros (Falhas Transitórias)

Código	Nome	Quando Usado
72	Par remoto n��o respondendo	Tempo limite do PCRF/PGW-U
73	Colisão com solicitação iniciada pela rede	Operações simultâneas

---

Monitoramento

Métricas S5/S8

```
# Contadores de mensagens
s5s8_inbound_messages_total{message_type="create_session_request"}
s5s8_inbound_messages_total{message_type="delete_session_request"}

# Contadores de erro
s5s8_inbound_errors_total

# Latência de manipulação de mensagens
s5s8_inbound_handling_duration_bucket

# TEIDs ativos
teid_registry_count
```

Consultas Úteis

Taxa de Criação de Sessão:

```
rate(s5s8_inbound_messages_total{message_type="create_session_request
[5m])
```

Taxa de Erro:

```
rate(s5s8_inbound_errors_total[5m])
```

Latência (p95):

```
histogram_quantile(0.95,

rate(s5s8_inbound_handling_duration_bucket{request_message_type="crea
[5m])
)
```

Solução de Problemas

Problema: Sem Resposta do OmniPGW

Sintomas:

- SGW-C envia Solicitação de Criação de Sessão
- Nenhuma resposta recebida
- Tempo limite no SGW-C

Causas:

1. Problema de conectividade de rede
2. OmniPGW não ouvindo no IP configurado
3. Firewall bloqueando UDP 2123
4. TEID incorreto na solicitação

Depuração:

```
# Verificar se o OmniPGW está ouvindo
netstat -ulnp | grep 2123

# Verificar pacotes de entrada
tcpdump -i any -n port 2123

# Verificar configuração
grep "local_ipv4_address" config/runtime.exs

# Verificar firewall
iptables -L -n | grep 2123
```

Problema: Falha na Criação de Sessão

Sintomas:

- Resposta de Criação de Sessão com causa de erro
- Sessão não estabelecida

Causas Comuns:

```
Causa 65 (Usuário Desconhecido):
  → PCRF rejeitou assinante
  → Verifique IMSI no HSS/SPR

Causa 66 (Sem recursos):
  → Pool de IP esgotado
  → Verifique: curl http://pgw:9090/metrics | grep
address_registry_count
  → Expanda o pool de IP

Causa 72 (Par remoto não respondendo):
  → Tempo limite do PCRF ou PGW-U fora do ar
  → Verifique a conectividade Gx
  → Verifique a associação PFCP
```

Problema: Colisão de TEID

Sintomas:

- Mensagem roteada para a sessão errada
- Comportamento inesperado

Causa:

- TEID reutilizado antes da limpeza
- Bug na alocação de TEID

Resolução:

- Garantir alocação única de TEID
- Verificar registro de TEID para vazamentos

Melhores Práticas

Design de Rede

1. **Interface de Rede Dedicada**
    - Use VLAN separada para S5/S8
    - Isolar do tráfego de gerenciamento
2. **Otimização de MTU**
    - Garantir que o MTU suporte cabeçalhos GTP
    - MTU mínima: 1500 bytes (1464 carga útil + 36 GTP)
3. **Redundância**
    - Múltiplas instâncias do OmniPGW
    - Balanceamento de carga baseado em DNS do SGW-C

Desempenho

1. **Tamanhos de Buffer UDP**
    - Aumentar buffers de soquete para alta carga
    - Típico: 4-8 MB por soquete
2. **Limites de Conexão**
    - Planejar para o número esperado de sessões
    - Monitorar contagem do registro de TEID

Segurança

1. **Filtragem de IP**
    - Permitir apenas GTP-C de IPs SGW-C conhecidos
    - Usar iptables ou ACLs de rede

2. **Validação de Mensagens**
    - OmniPGW valida todas as mensagens de entrada
    - Rejeita pacotes GTP-C malformados

Documentação Relacionada

Funções Principais

- **Guia de Configuração** - Configuração da interface S5/S8, configuração de IP local
- **Gerenciamento de Sessão** - Ciclo de vida da sessão PDN, estabelecimento de bearer
- **Alocação de IP do UE** - Entrega de endereço IP via Resposta de Criação de Sessão
- **Configuração de PCO** - Parâmetros de PCO em mensagens GTP-C

Interfaces Relacionadas

- **Interface Gn/Gp** - GTP-C v1 para funcionalidade GGSN 2G/3G
- **Interface PFCP** - Coordenação do plano de usuário com o plano de controle S5/S8
- **Interface Diameter Gx** - Integração de políticas com configuração de bearer
- **Interface Diameter Gy** - Integração de cobrança com gerenciamento de bearer

Operações

- **Guia de Monitoramento** - Métricas GTP-C S5/S8, rastreamento de mensagens
- **Formato de CDR de Dados** - Geração de CDR a partir de sessões GTP-C

Voltar ao Guia de Operações

Interface S5/S8 do OmniPGW - *por Omnitouch Network Services*

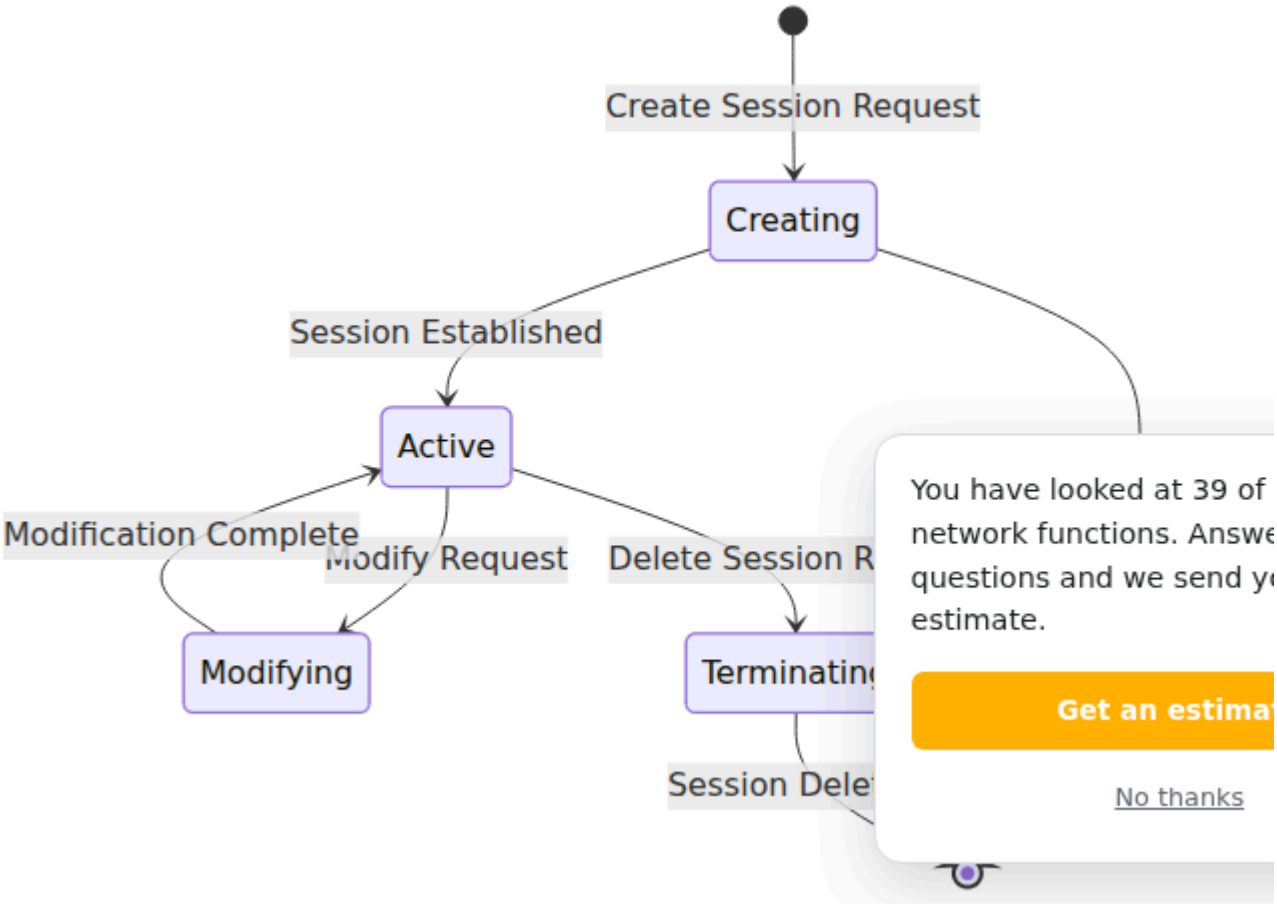
Guia de Gerenciamento de Sessões

Ciclo de Vida e Operações da Conexão PDN

OmniPGW da Omnitouch Network Services

Visão Geral

Uma **Sessão PDN (Packet Data Network)** representa a conexão de dados de um UE através do OmniPGW. Cada sessão coordena múltiplas interfaces e recursos para habilitar a conectividade de dados.



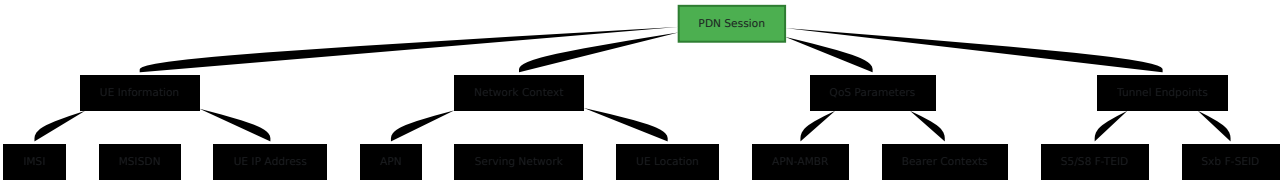
Componentes da Sessão

Identificadores da Sessão

Cada sessão possui múltiplos identificadores para diferentes interfaces:

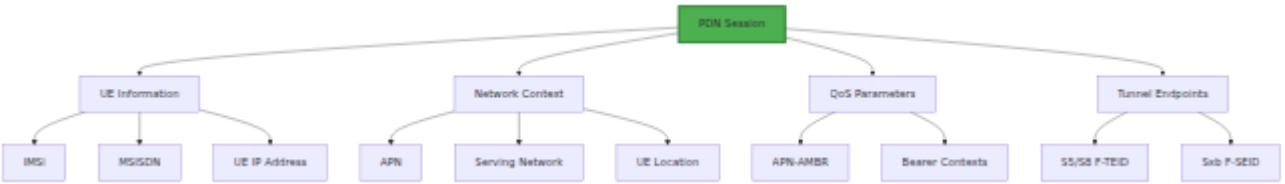
Identificador	Interface	Propósito
TEID	S5/S8 (GTP-C)	Tunnel Endpoint ID para comunicação SGW-C
SEID	Sxb (PFCP)	Session Endpoint ID para comunicação PGW-U
Session-ID	Gx (Diameter)	Sessão Diameter para comunicação PCRF
Charging-ID	Contabilidade	ID único para cobrança

Dados da Sessão



Criação da Sessão

Fluxo de Chamadas



Etapas

1. Receber Create Session Request (S5/S8)

A criação da sessão é iniciada via sinalização GTP-C na interface S5/S8. Veja Interface S5/S8 para detalhes completos do protocolo GTP-C e formatos de mensagem.

Entrada:

- IMSI, MSISDN, IMEI
- APN (por exemplo, "internet")
- Tipo de RAT (EUTRAN)
- Localização do UE (TAI, ECGI)
- Contexto do Bearer (QoS, F-TEID)

2. Alocação de Recursos

```
- Alocar IP do UE do pool de APN
- Gerar Charging ID
- Gerar Gx Session-ID
- Alocar S5/S8 TEID
- Selecionar par PGW-U
```

3. Solicitação de Política (Gx)

Solicitar política do PCRF:

- Enviar CCR-Initial
- Receber CCA-Initial com regras de QoS e PCC

4. Configuração do Plano do Usuário (PFCP)

Programar o PGW-U com regras de encaminhamento:

- Enviar Session Establishment Request
- Incluir PDRs, FARs, QERs, BAR
- Receber F-TEID para o túnel S5/S8

5. Resposta ao SGW-C

Enviar Create Session Response:

- Endereço IP do UE
- S5/S8 F-TEID (do PGW-U)
- PCO (DNS, P-CSCF, MTU)
- Contexto do Bearer

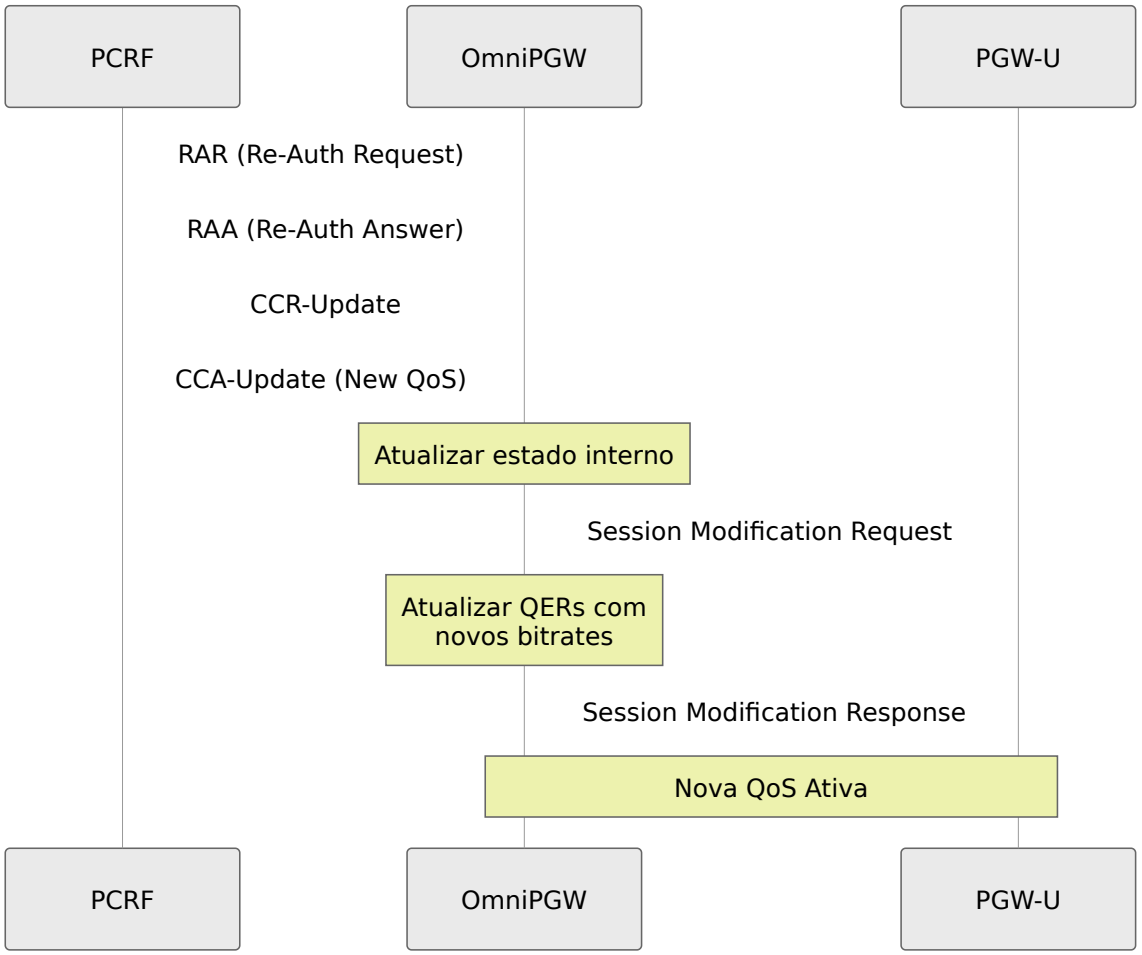
Modificação da Sessão

Gatilhos

As sessões podem ser modificadas devido a:

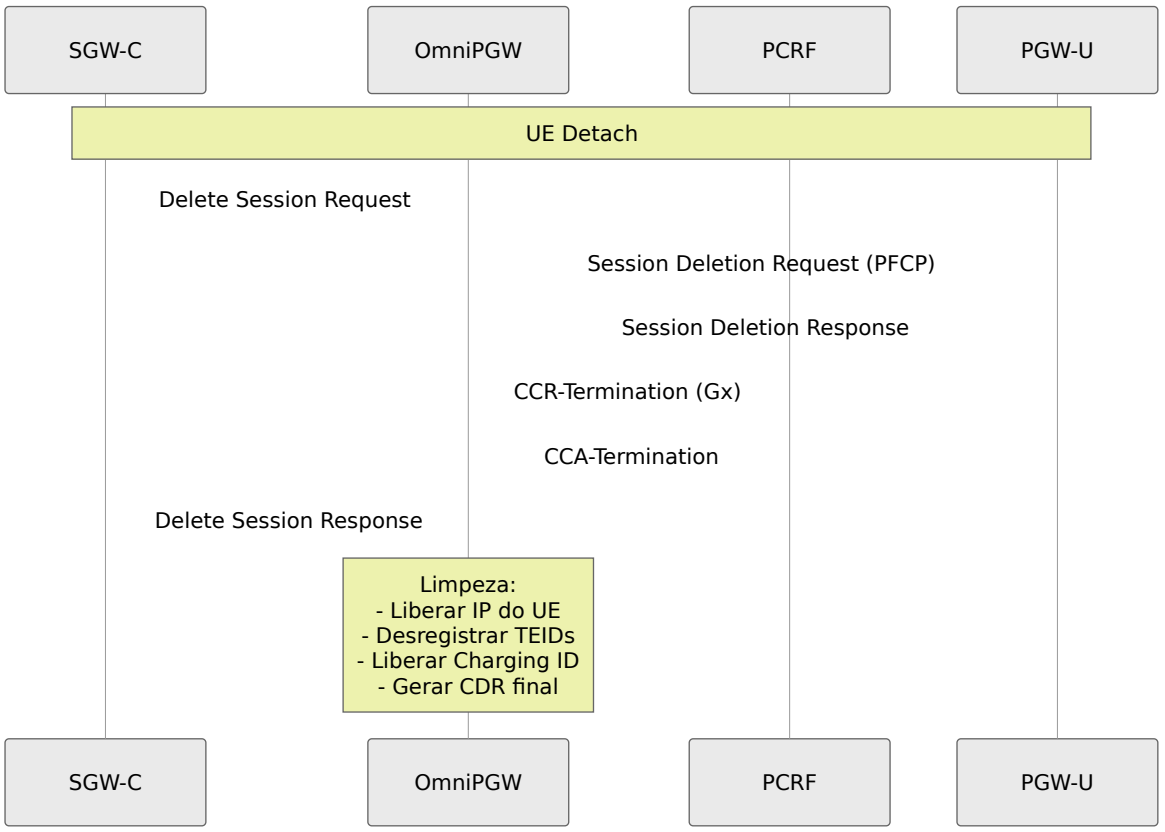
- **Mudanças de QoS** - Atualizações de bitrates pelo PCRF
- **Operações de Bearer** - Adicionar/remover bearers dedicados
- **Handover** - Mudança de SGW
- **Atualizações de Política** - Novas regras de PCC do PCRF

Fluxo de Modificação de QoS



Exclusão da Sessão

Fluxo de Chamadas



Processo de Limpeza

Recursos Liberados:

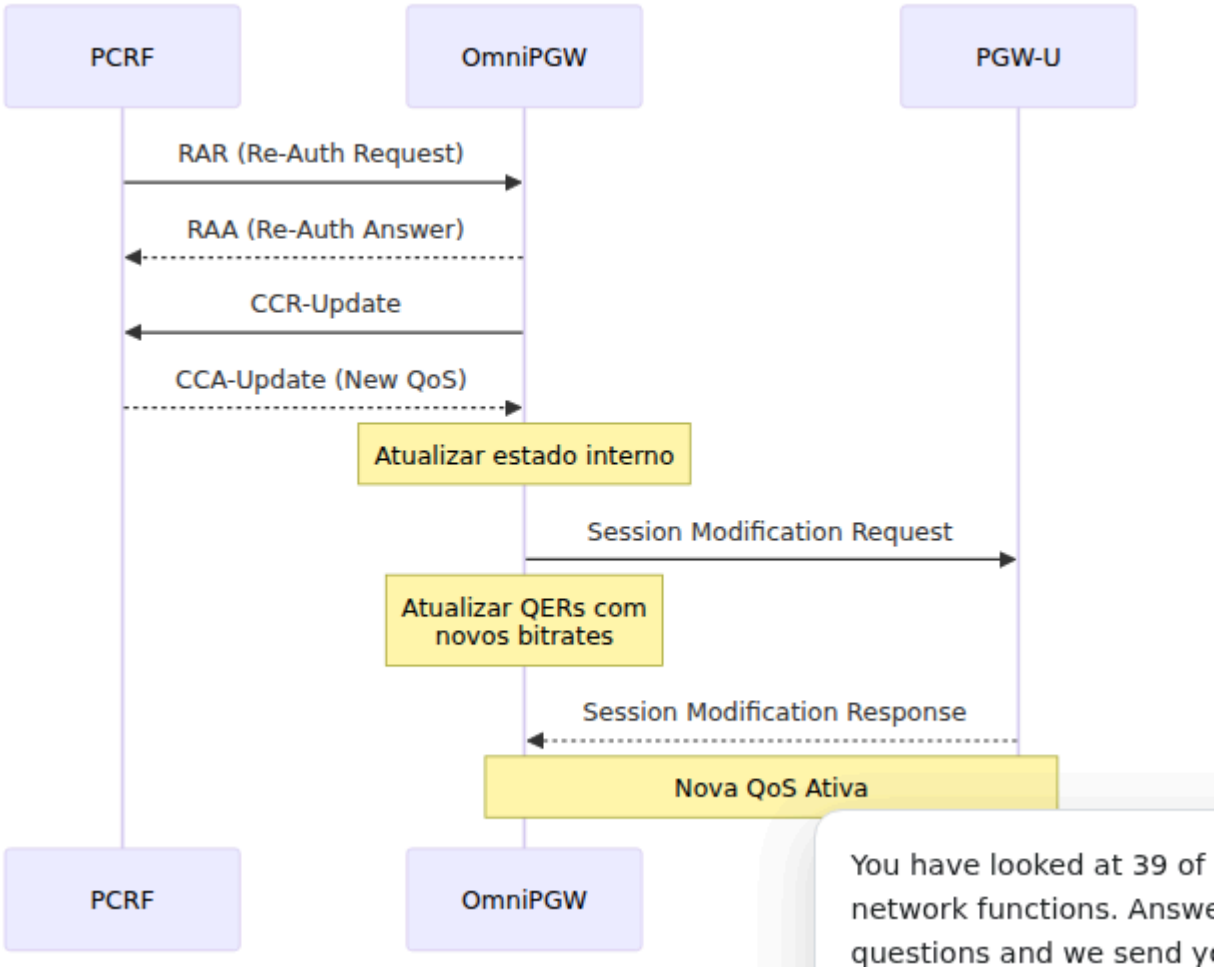
1. Endereço IP do UE → de volta ao pool
2. TEID → removido do registro
3. SEID → removido do registro
4. Session-ID → removido do registro
5. Charging-ID → liberado
6. Processo da sessão encerrado

Registros de Cobrança Gerados:

- CDR final (Charging Data Record) escrito para cobrança offline - Veja Formato de CDR de Dados

Estado da Sessão

Máquina de Estados



Rastreamento da Sessão

Consultas ao Registro:

```
Por TEID (S5/S8):
  TEID 0x12345678 → Session PID

Por SEID (Sxb):
  SEID 0xABCDEF → Session PID

Por Session-ID (Gx):
  "pgw.example.com;123;456" → Session PID

Por IP do UE:
  100.64.1.42 → Session PID

Por IMSI + EBI:
  "310260123456789" + EBI 5 → Session PID
```

Monitoramento de Sessões

Contagem de Sessões Ativas

```
# Total de sessões ativas
teid_registry_count

# Sessões PFCP
seid_registry_count

# Sessões Gx
session_id_registry_count
```

Métricas da Sessão

```
# Taxa de criação de sessões
rate(s5s8_inbound_messages_total{message_type="create_session_request

# Taxa de exclusão de sessões
rate(s5s8_inbound_messages_total{message_type="delete_session_request

# Latência de criação de sessões (p95)
histogram_quantile(0.95,

rate(s5s8_inbound_handling_duration_bucket{request_message_type="crea
[5m])
)
```

Problemas Comuns

Falhas na Criação da Sessão

Causas:

1. **Pool de IPs Esgotado** - Nenhum IP disponível
2. **PCRF Inacessível** - Timeout Gx
3. **PGW-U Fora do Ar** - Nenhum par PFCP disponível
4. **Rejeição do PCRF** - Usuário desconhecido, não autorizado

Depuração:

```
# Verificar pool de IP
curl http://pgw:9090/metrics | grep address_registry_count

# Verificar conectividade do PCRF
# Verificar erros de Gx nos logs

# Verificar associação do PGW-U
# Verificar status do par PFCP
```

Sessão Presa/Obsoleta

Sintomas:

- Sessão não excluída corretamente
- Recursos não liberados
- Registros mostram contagem maior do que o esperado

Causas:

1. Delete Session Request não recebida
2. Crash do processo da sessão sem limpeza
3. Vazamento de registro

Resolução:

```
# Reiniciar OmniPGW (libera todas as sessões)
# Implementar mecanismo de timeout de sessão
```

UE Não Consegue Estabelecer Sessão

Sintomas:

- Falha no attach do UE
- Create Session Response com causa de erro

Causas Comuns e Respostas:

Valor da Causa	Significado	Ação
Usuário Desconhecido	PCRF rejeitou (IMSI não está no banco de dados)	Provisionar assinante
Recursos Não Disponíveis	Pool de IP esgotado	Expandir pool de IP
Par Remoto Não Respondendo	Timeout do PCRF/PGW-U	Verificar conectividade
Serviço Não Suportado	APN inválido	Configurar pool de APN

Melhores Práticas

Limites de Sessão

Configurar capacidade apropriada:

```
Usuários concorrentes esperados: 10.000
Sobrecarga de sessão por usuário: ~10KB RAM
Total de RAM para sessões: ~100MB

Configurações da VM Erlang:
  - Máx. processos: 262.144 (padrão)
  - Tamanho do heap do processo: Ajustar com base na carga
```

Limpeza de Sessão

Garantir limpeza adequada:

1. Sempre responder a Delete Session Requests
2. Implementar timeout de sessão para sessões obsoletas

3. Monitorar contagens de registro para vazamentos

Alta Disponibilidade

Redundância de Sessão:

- Usar design sem estado (sessões vinculadas à instância)
- Implementar banco de dados de sessão para HA (futuro)
- DNS/balancer de carga para failover

Elementos de Dados da Sessão

Que Informações Uma Sessão Armazena?

Cada sessão PDN ativa mantém as seguintes informações:

Identificação do UE:

- IMSI: "310260123456789" (identidade do assinante)
- MSISDN: "14155551234" (número de telefone)
- MEI/IMEI: Identificador do dispositivo

Detalhes da Conexão PDN:

- APN: "internet" (nome da rede)
- Endereço IP do UE: 100.64.1.42 (IP alocado)
- Tipo de PDN: IPv4, IPv6 ou IPv4v6

Identificadores da Sessão:

- Charging ID: Identificador único de cobrança
- EBI do Bearer Padrão: EPS Bearer Identifier (normalmente 5)

Parâmetros de QoS:

- APN-AMBR: Taxa Máxima de Bits Agregada

    - Uplink: 100 Mbps
    - Downlink: 50 Mbps

Regras de Encaminhamento:

- PDRs (Regras de Detecção de Pacotes): Combinar pacotes
- FARs (Regras de Ação de Encaminhamento): Ações de encaminhar/descartar
- QERs (Regras de Aplicação de QoS): Limitação de taxa
- BAR (Regra de Ação de Bufferização): Bufferização de downlink

Contexto da Interface:

- Estado S5/S8: TEIDs locais/remotos, endereço SGW-C
- Estado Sxb: SEIDs locais/remotos, endereço PGW-U
- Estado Gx: Diameter Session-ID, contador de solicitações

Monitoramento de Sessões Ao Vivo

OmniPGW expõe uma **API REST OAM** em tempo real para inspecionar e gerenciar sessões ativas sem precisar consultar métricas ou logs. A API é servida sobre HTTPS/TLS na porta `8443` com todas as rotas sob o prefixo `/api`:

```
GET    /api/sessions                 # todas as sessões ativas
GET    /api/sessions?search=<imsi>   # filtrar por IMSI / MSISDN /
IP
GET    /api/sessions/<imsi>          # detalhes completos de uma
sessão
PUT    /api/sessions/<imsi>          # atualizar estado da sessão
ao vivo (ação do operador)
DELETE /api/sessions/<imsi>          # encerrar uma sessão (ação
do operador)
```

Documentação interativa da API (Swagger UI) está disponível em `https://<omnipgw-ip>:8443/api/docs`.

```
# Listar todas as sessões ativas
curl -k https://<omnipgw-ip>:8443/api/sessions

# Filtrar por IMSI, MSISDN ou endereço IP
curl -k "https://<omnipgw-ip>:8443/api/sessions?
search=310170123456789"
```

Pesquisa de UE & Análise Detalhada

Endpoints:

```
GET /api/sessions?search=<imsi|msisdn|ip>   # encontrar sessões de
um assinante
GET /api/sessions/<imsi>                    # detalhes completos
de uma sessão
```

O parâmetro `search` aceita qualquer um de:

- **IMSI** (por exemplo, "310170123456789")
- **MSISDN** (número de telefone)
- **Endereço IP** (por exemplo, "100.64.1.42")

```
# Análise detalhada da sessão de um único assinante
curl -k https://<omnipgw-ip>:8443/api/sessions/310170123456789
```

Os detalhes da sessão retornam informações abrangentes:

a) Sessões Ativas

- Todas as sessões ativas para este assinante
- IMSI, MSISDN, Endereço IP do UE
- APN, Tipo de RAT
- PGW TEID, SGW TEID

b) Localização Atual Dados de localização em tempo real da sessão:

- **TAC** (Código da Área de Rastreamento) - Área de rastreamento onde o UE está localizado
- **ID da Célula (ECI)** - Identificador da Célula E-UTRAN
- **ECGI** - Identificador Global da Célula E-UTRAN (PLMN + ECI)
- **MCC/MNC** - Código do País Móvel / Código da Rede Móvel

c) Informações do Bearer Listagem detalhada dos bearers com parâmetros de QoS:

Bearer Padrão:

- EBI (EPS Bearer Identifier)
- QCI (QoS Class Identifier)
- Nome da Regra de Cobrança
- APN-AMBR (uplink/downlink)

Bearers Dedicados (se ativos):

- EBI, QCI, Nome da Regra de Cobrança
- MBR UL/DL (Taxa Máxima de Bits)
- GBR UL/DL (Taxa Garantida de Bits)

d) Informações de Cobrança (Interface Gy)

- Gy Session ID
- Quota Concedida, Quota Usada
- Características de Cobrança

e) Informações de Política (Interface Gx)

- Gx Session ID
- Host de Origem/Destino do PCRF
- Número da Solicitação CC
- Regras de Cobrança Instaladas (regras PCC dos bearers)

f) Eventos Recentes

- Histórico de eventos para este assinante (veja Histórico de Sessão & Log de Auditoria)
- Eventos de criação/atualização/exclusão de sessão

Casos de Uso:

- Solucionar problemas específicos de assinantes
- Verificar o estabelecimento da sessão
- Checar o endereço IP atribuído
- Inspecionar parâmetros da sessão

Todas as Sessões Ativas

Endpoints:

```
GET    /api/sessions                 # listar todas as sessões PDN
ativas
GET    /api/sessions?search=<term>   # filtrar a lista
PUT    /api/sessions/<imsi>          # atualizar o estado de uma
sessão ao vivo
DELETE /api/sessions/<imsi>          # encerrar uma sessão
```

Propósito: Enumerar todas as sessões PDN ativas. Consulte `GET /api/sessions` no seu intervalo preferido para uma visão ao vivo da tabela de sessões.

```
# Instantâneo de todas as sessões ativas
curl -k https://<omnipgw-ip>:8443/api/sessions

# Filtrar por prefixo de IMSI, IP do UE, MSISDN ou APN
curl -k "https://<omnipgw-ip>:8443/api/sessions?search=100.64"

# Ações do operador em uma sessão específica
curl -k -X PUT    https://<omnipgw-
ip>:8443/api/sessions/310170123456789   # atualizar estado
curl -k -X DELETE https://<omnipgw-
ip>:8443/api/sessions/310170123456789   # encerrar
```

Informações Rápidas da Sessão (retornadas por sessão):

- **IMSI** - Identidade do assinante
- **UE IP** - Endereço IP alocado
- **SGW TEID** - ID do túnel S5/S8 do SGW
- **PGW TEID** - ID do túnel S5/S8 do OmniPGW
- **APN** - Nome do Ponto de Acesso

Filtragem (via `?search=`):

- IMSI (por exemplo, "310260")
- Endereço IP do UE (por exemplo, "100.64")
- MSISDN / número de telefone
- Nome da APN

Detalhes Completos (`GET /api/sessions/<imsi>`):

- Informações completas do assinante (IMSI, MSISDN, IMEI)
- Contexto da rede (tipo de RAT, rede servidora MCC/MNC)
- Parâmetros de QoS (AMBR uplink/downlink)
- Identificadores de túnel (ambos TEIDs em formato hexadecimal)
- ID do processo para depuração
- Estado completo da sessão (estrutura de dados bruta)

Endpoints de Saúde Referenciados Cruzadamente

A API OAM também expõe a saúde de pares/topologia da qual a visualização da sessão depende:

```
GET /api/upf         # status e associações de pares PFCP / UPF
GET /api/diameter    # conectividade de pares Diameter (Gx/Gy)
```

```
# Verificar conectividade de pares do plano do usuário e Diameter
curl -k https://<omnipgw-ip>:8443/api/upf
curl -k https://<omnipgw-ip>:8443/api/diameter
```

Casos de Uso:

- Verificar conexões de pares (associações UPF, conectividade PCRF/OCS)
- Correlacionar falhas de estabelecimento de sessão com saúde de pares
- Verificação rápida da saúde da rede

Histórico de Sessão & Log de Auditoria

Endpoint:

```
GET /api/session_history[?type=&search=&page=&page_size=]
```

Propósito: Rastrear eventos históricos de sessão e trilha de auditoria.

Parâmetros:

- `type` - filtrar por tipo de evento (por exemplo, sessão criada, sessão excluída)
- `search` - filtrar por IMSI, MSISDN, endereço IP ou TEID
- `page`, `page_size` - paginar pelos resultados

```
# Eventos de exclusão recentes para um assinante, primeira página
curl -k "https://<omnipgw-ip>:8443/api/session_history?
type=session_deleted&search=310170123456789&page=1&page_size=50"
```

Tipos de Eventos Rastreados:

- Eventos de criação de sessão
- Eventos de exclusão de sessão
- Eventos de modificação
- Eventos de erro

Casos de Uso:

- Trilhas de auditoria para conformidade (paginando e arquivando resultados)
- Análise histórica de sessões
- Solucionar problemas de eventos passados
- Gerar relatórios de uso
- Rastrear padrões de sessão ao longo do tempo

Casos de Uso Operacionais

Verificação da Sessão:

```
1. Usuário relata problema de conectividade
2. GET /api/sessions?search=<imsi|msisdn> para localizar o
assinante
3. GET /api/sessions/<imsi> para verificar se a sessão existe e se
o UE tem um endereço IP
4. Verificar se os valores de QoS correspondem ao plano do
assinante
5. Verificar se os endpoints de túnel estão estabelecidos
```

Monitoramento de Capacidade:

- Consulte `GET /api/sessions` (ou a métrica `teid_registry_count`) para a contagem de sessões ativas
- Compare com a capacidade licenciada
- Identifique padrões de uso por APN (via `?search=<apn>`)

Solução de Problemas:

- Encontrar uma sessão específica por qualquer identificador via `?search=`
- Inspecionar o estado completo da sessão (`GET /api/sessions/<imsi>`) sem SSH/IEx
- Verificar se os TEIDs do SGW e PGW correspondem entre os sistemas
- Verificar os valores de AMBR aplicados do PCRF
- Encerrar uma sessão presa com `DELETE /api/sessions/<imsi>`

Vantagens Sobre Métricas:

- Ver detalhes individuais da sessão (métricas mostram agregados)
- Capacidades de busca e filtragem via `?search=`
- Inspeção de estado em tempo real
- Scriptável/automatizável (JSON sobre HTTPS) para integração com ferramentas

---

Documentação Relacionada

Funções Centrais da Sessão

- **Interface PFCP** - Estabelecimento de sessão do plano do usuário, PDRs, FARs, QERs, URRs
- **Alocação de IP do UE** - Atribuição de endereço IP, gerenciamento de pool de APN
- **Configuração de PCO** - Parâmetros de DNS, P-CSCF, MTU entregues ao UE
- **Guia de Configuração** - Seleção de UPF, fluxos de estabelecimento de sessão

Política e Cobrança

- **Interface Diameter Gx** - Controle de política do PCRF, regras de PCC, gerenciamento de QoS
- **Interface Diameter Gy** - Cobrança online OCS, rastreamento de quota
- **Formato de CDR de Dados** - Geração de registros de cobrança offline

Interfaces de Rede

- **Interface S5/S8** - Protocolo GTP-C, comunicação SGW-C
- **Gerenciamento de QoS & Bearer** - Aplicação de QoS do Bearer

Operações

- **Guia de Monitoramento** - Métricas de sessão, rastreamento de sessões ativas, alertas
- **Monitoramento P-CSCF** - Monitoramento de sessão IMS

Voltar ao Guia de Operações

Guia de Solução de Problemas do OmniPGW

Procedimentos de Solução de Problemas e Problemas Comuns

por Omnitouch Network Services

Índice

1. Visão Geral
2. Ferramentas de Solução de Problemas
3. Problemas de Estabelecimento de Sessão
4. Problemas de PFCP / Plano de Usuário
5. Problemas de Diâmetro (Gx/Gy)
6. Problemas de Alocação de IP
7. Referência Rápida

Visão Geral

Este guia fornece procedimentos de solução de problemas passo a passo para problemas operacionais comuns do OmniPGW. Cada problema inclui:

- **Sintoma**: O que você observará
- **Causas Prováveis**: Causas raiz comuns
- **Diagnóstico**: Como confirmar a causa
- **Resolução**: Correção passo a passo
- **Prevenção**: Como evitar recorrências

Documentação Relacionada

- **Guia de Monitoramento** - Métricas do Prometheus, alertas, monitoramento de desempenho
- **Guia de Configuração** - Referência de configuração do sistema

---

Ferramentas de Solução de Problemas

API OAM

URL Base: `https://<omnipgw_ip>:8443` (HTTPS/TLS; todas as rotas sob o prefixo `/api`)

Documentação da API Interativa (Swagger UI):
`https://<omnipgw_ip>:8443/api/docs`

Endpoints Principais:

- `GET /api/sessions[?search=<imsi|msisdn|ip>]`, `GET /api/sessions/<imsi>` - Visualizador de sessões ativas (pesquisar por IMSI, IP, MSISDN)
- `GET /api/diameter[?status=connected|disconnected]`, `GET /api/diameter/<origin-host>` - Status do par de Diâmetro (Gx PCRF, Gy OCS)
- `GET /api/charging[?search=<imsi>]`, `GET /api/charging/<imsi>` - Status de cobrança online (Gy)
- `GET /api/upf`, `GET /api/upf/<ip>` - Status do par UPF / PFCP (conectividade PGW-U)
- `GET /api/ip_pools`, `GET /api/ip_pools/<name>` - Utilização do pool de IP
- `GET /api/session_history[?type=&search=&page=&page_size=]` - Histórico de sessões
- `GET /api/status` - Verificação de saúde (`{"result":"ok"}`)

Exemplo:

```
curl -k https://<omnipgw_ip>:8443/api/sessions
```

O streaming de logs não é mais fornecido pela plataforma; use logs do host (`journalctl`/arquivos de log) com os filtros abaixo.

Métricas do Prometheus

Acesso: `http://<omnipgw_ip>:9090/metrics`

Métricas Principais:

- `teid_registry_count` - Sessões ativas
- `address_registry_count` - IPs de UE alocados
- `sxb_inbound_errors_total` - Erros de PFCP
- `gx_inbound_errors_total` - Erros de Diâmetro Gx
- `gy_inbound_errors_total` - Erros de Diâmetro Gy

Veja o Guia de Monitoramento para referência completa de métricas.

Análise de Logs

Logs: Inspecione os logs do host (`journalctl -u omnipgw_c` ou o arquivo de log configurado) e grep para as strings abaixo. Para uma visão estruturada e consultável dos eventos do ciclo de vida da sessão, use `GET /api/session_history[?type=&search=&page=&page_size=]`.

Filtros Comuns de Log:

- "create_session_request" - Estabelecimento de sessão
- "Credit Control" - Interações Gx/Gy
- "PFCP Session" - Programação do plano de usuário
- "error" ou "ERROR" - Mensagens de erro
- "timeout" - Problemas de timeout

Problemas de Estabelecimento de Sessão

Problema: Solicitação de Criação de Sessão Rejeitada com "Sem Recursos Disponíveis"

Sintoma:

- SGW-C recebe Resposta de Criação de Sessão com causa "Sem recursos disponíveis" (73)
- Todas as tentativas de nova sessão falham
- Sessões existentes continuam funcionando
- Logs: `[PGW-C] Solicitação de Criação de Sessão bloqueada - licença inválida`

Captura do Wireshark mostrando Resposta de Criação de Sessão com causa "Sem recursos disponíveis"

Causa Provável:

- Licença OmniPGW inválida ou expirada
- Servidor de licença inacessível

Diagnóstico:

1. **Verifique a métrica da licença:**

   ```
   license_status
   ```

   - Valor de 0 indica licença inválida

2. **Verifique logs para avisos de licença:**

   - Pesquise por "license" ou "License"
   - Procure mensagens "Unable to contact license server"

3. **Verifique a conectividade do servidor de licença:**

   - Verifique a URL configurada em `config/runtime.exs` sob `:license_client`
   - Padrão: `https://localhost:10443/api`

Resolução:

1. **Verifique se o servidor de licença é acessível:**

   ```
   curl -k https://<license_server_ip>:10443/api/status
   ```

2. **Verifique a configuração da licença em `config/runtime.exs`:**

   ```
   config :license_client,
     license_server_api_urls:
   ["https://<license_server_ip>:10443/api"],
     licensee: "Nome da Sua Empresa"
   ```

3. **Verifique se o produto está licenciado:**

   - Nome do produto: `omnipgwc`
   - Entre em contato com a Omnitouch para verificar o status da licença

4. **Reinicie o OmniPGW após alterações na configuração**

Prevenção:

- Monitore a métrica `license_status` com alertas críticos
- Garanta alta disponibilidade do servidor de licença
- Configure alertas de expiração de licença antes da expiração

---

Problema: Solicitação de Criação de Sessão Rejeitada (Outras Causas)

Sintoma:

- SGW-C recebe Resposta de Criação de Sessão com causa de erro
- Usuários não conseguem estabelecer conexões PDN
- Métrica: `s5s8_inbound_errors_total` aumentando

Causas Prováveis:

1. Pool de IP esgotado
2. PCRF (Gx) inacessível ou rejeitando política
3. PGW-U (PFCP) indisponível
4. Configuração de APN inválida

Diagnóstico:

1. **Verifique a utilização do pool de IP:**

   ```
   address_registry_count
   ```

   - Se igual ao tamanho do pool configurado, o pool está esgotado

2. **Verifique a conectividade do PCRF:**

   - Consulte `GET /api/diameter` (ou `GET /api/diameter?status=disconnected`)
   - Procure status do par PCRF = "disconnected"
   - Logs: Pesquise "Credit Control Answer" para erros

3. **Verifique o status do par PFCP:**
    - Consulte `GET /api/upf` (ou `GET /api/upf/<ip>` para um PGW-U espec��fico)
    - Procure por "Association: DOWN"
    - Métrica: `pfcp_peer_associated` = 0
4. **Verifique a configuração de APN:**
    - Revise `config/runtime.exs` sob `ue.apn_map`
    - Verifique se a APN solicitada existe na configuração

Resolução:

Para Esgotamento do Pool de IP:

1. Identifique sessões antigas: consulte `GET /api/sessions` (opcionalmente `GET /api/sessions?search=<imsi|msisdn|ip>`), procure por sessões antigas
2. Expanda o pool de IP em `config/runtime.exs`:

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => "10.0.0.0/23"  # Alterado de /24 para /23
(dobra a capacidade)
    }
  }
```

3. Reinicie o OmniPGW
4. Verifique: `curl http://<ip>:9090/metrics | grep address_registry_count`

Para Problemas de Conectividade do PCRF:

1. Verifique a conectividade da rede: `ping <pcrf_ip>`
2. Verifique o serviço de Diâmetro do PCRF: `telnet <pcrf_ip> 3868`
3. Verifique a configuração do par de Diâmetro em `config/runtime.exs`
4. Reinicie o OmniPGW se a configuração foi alterada

5. Verifique via `GET /api/diameter?status=connected` (o par deve mostrar "connected")

Para Problemas de PFCP:

- Veja a seção Problemas de PFCP / Plano de Usuário

Prevenção:

- Monitore a utilização do pool de IP com alertas em 80%
- Monitore a conectividade do PCRF com alertas de par de Diâmetro
- Implemente limpeza de sessões para sessões ociosas

Problema: Sessões Presas em Estado Intermediário

Sintoma:

- A sessão aparece em `GET /api/sessions` mas está incompleta
- Métricas mostram contagem de sessões crescente, mas sem tráfego de usuário
- Solicitação de Exclusão de Sessão falha ou expira

Causas Prováveis:

1. Estabelecimento de Sessão PFCP falhou, mas sessão S5/S8 foi criada
2. CCR-Initial do PCRF expirou
3. Solicitação de Criação de Bearer (bearer dedicado) falhou
4. Interrupção da rede durante a configuração da sessão

Diagnóstico:

1. **Pesquise a sessão via API OAM:**
   - `GET /api/sessions/<imsi>` (ou `GET /api/sessions?search=<imsi>`)
   - Verifique se `pfcp_seid` está presente (se ausente, PFCP falhou)
   - Verifique se `gx_session_id` está presente (se ausente, Gx falhou)

2. **Verifique logs para o IMSI:**
    - Filtre logs por IMSI
    - Procure por "Solicitação de Estabelecimento de Sessão" (PFCP)
    - Procure por "Solicitação de Controle de Crédito" (Gx)
    - Procure por mensagens de timeout ou erro
3. **Verifique métricas:**

```
# Sessões com TEID, mas sem sessão PFCP
teid_registry_count - seid_registry_count

# Sessões com TEID, mas sem sessão Gx
teid_registry_count - session_id_registry_count
```

Resolução:

1. **Para falhas de estabelecimento PFCP:**
    - Verifique a saúde e logs do PGW-U
    - Verifique a associação PFCP: `GET /api/upf`
    - Envie Solicitação de Exclusão de Sessão do SGW-C para limpeza
2. **Para problemas de timeout Gx:**
    - Verifique a latência do PCRF: `histogram_quantile(0.95, rate(gx_outbound_transaction_duration_bucket[5m]))`
    - Aumente o timeout Gx em `config/runtime.exs` se necessário
    - Envie Solicitação de Exclusão de Sessão para limpeza
3. **Limpeza manual (último recurso):**
    - Atualmente requer reinício do OmniPGW para limpar sessões presas
    - Monitore `teid_registry_count` antes/depois do reinício para confirmar a limpeza

Prevenção:

- Monitore métricas de latência PFCP e Gx

- Implemente timeout/limpeza de sessão para sessões incompletas
- Alerta sobre discrepâncias na contagem de registros

Problemas de PFCP / Plano de Usuário

Problema: Associação PFCP Indisponível

Sintoma:

- `GET /api/upf` mostra "Associação: DOWN"
- Todas as novas estabelecimentos de sessão falham
- Métrica: `pfcp_peer_associated` = 0
- Logs: "timeout de heartbeat PFCP" ou "Falha na Configuração da Associação"

Causas Prováveis:

1. PGW-U inacessível (problema de rede)
2. PGW-U travou ou reiniciou
3. Incompatibilidade de configuração PFCP (IP, porta)
4. Firewall bloqueando UDP 8805

Diagnóstico:

1. **Verifique a conectividade da rede:**

   ```
   ping <pgw_u_ip>
   nc -u -v <pgw_u_ip> 8805
   ```

2. **Verifique a configuração PFCP:**

   - Revise `config/runtime.exs` sob `upf.peer_list`

- Verifique se o endereço IP e o ID do nó correspondem à configuração do PGW-U

3. **Verifique o status do PGW-U:**

   - Acesse os logs do PGW-U
   - Verifique se o PGW-U está em execução: `systemctl status omnipgw_u` (ou equivalente)

4. **Verifique métricas:**

```
# Falhas de heartbeat
pfcp_consecutive_heartbeat_failures

# Taxa de erro PFCP
rate(sxb_inbound_errors_total[5m])
```

Resolução:

1. **Para problemas de rede:**

   - Verifique o roteamento: `traceroute <pgw_u_ip>`
   - Verifique as regras do firewall: Certifique-se de que UDP 8805 está permitido
   - Verifique grupos de segurança (se implantação em nuvem)

2. **Para travamentos do PGW-U:**

   - Reinicie o serviço PGW-U
   - Aguarde 30 segundos para restabelecer a associação
   - Verifique via `GET /api/upf` (deve mostrar "Associação: UP")

3. **Para problemas de configuração:**

   - Corrija a configuração do par PFCP em `config/runtime.exs`
   - Reinicie o OmniPGW
   - Verifique se a associação foi estabelecida

Prevenção:

- Monitore a métrica `pfcp_peer_associated` com alertas críticos
- Monitore `pfcp_consecutive_heartbeat_failures` (alerta em > 2)
- Implemente instâncias redundantes do PGW-U
- Habilite keepalive/heartbeat PFCP (deve ser padrão)

Problema: Falhas de Modificação de Sessão PFCP

Sintoma:

- Criação de bearer dedicado falha
- Atualizações de políticas de QoS (do RAR do PCRF) falham
- Logs: "Falha na Solicitação de Modificação de Sessão"
- Métrica: `sxb_inbound_errors_total{message_type="session_modification_response"}` aumentando

Causas Prováveis:

1. Regras PFCP inválidas (referências PDR/FAR/QER)
2. Esgotamento de recursos do PGW-U
3. Conflitos de ID de regra
4. Bug de software do PGW-U

Diagnóstico:

1. **Verifique logs:**
    - Filtre por "Modificação de Sessão" e SEID
    - Procure códigos de causa de erro na resposta PFCP
    - Causas comuns: "ID de Regra já existe", "Fora de recursos"
2. **Verifique logs do PGW-U:**
    - Procure por erros de processamento PFCP
    - Verifique a utilização de recursos (CPU, memória)

3. **Verifique o estado da sessão via API OAM:**
    - `GET /api/sessions/<imsi>` (ou `GET /api/sessions?search=<imsi>`)
    - Revise `pdr_map`, `far_map`, `qer_map` para conflitos
    - Procure por IDs duplicados

Resolução:

1. **Para conflitos de regra:**
    - Exclua e recrie o bearer dedicado
    - Se persistir, exclua a sessão e faça o UE reconectar
2. **Para problemas de recursos do PGW-U:**
    - Verifique a capacidade do PGW-U (sessões, PDRs, throughput)
    - Escale o PGW-U se necessário
    - Reduza a carga de sessões na instância afetada do PGW-U
3. **Para bugs de software:**
    - Capture o estado completo da sessão (`GET /api/sessions/<imsi>`)
    - Capture logs de mensagens PFCP
    - Relate ao fornecedor com etapas de reprodução

Prevenção:

- Monitore a utilização de recursos do PGW-U
- Teste a criação de bearer dedicado em staging
- Monitore `sxb_inbound_errors_total` com alertas

Problemas de Diâmetro (Gx/Gy)

Problema: Par PCRF Desconectado (Gx)

Sintoma:

- `GET /api/diameter` (ou `GET /api/diameter?status=disconnected`) mostra par PCRF "desconectado"
- Sessões criadas sem políticas de QoS (QCI padrão=5 aplicado)
- Logs: "Falha na conexão do par de Diâmetro" ou "timeout CER/CEA"

Causas Prováveis:

1. PCRF inacessível (problema de rede)
2. Serviço PCRF fora do ar
3. Incompatibilidade de configuração de Diâmetro (Origin-Host, Realm)
4. Firewall bloqueando TCP 3868

Diagnóstico:

1. **Verifique a conectividade da rede:**

```
ping <pcrf_ip>
telnet <pcrf_ip> 3868
```

2. **Verifique a configuração de Diâmetro:**
   - Revise `config/runtime.exs` sob `diameter.peer_list`
   - Verifique se `host`, `realm`, `ip` correspondem à configuração do PCRF
   - Verifique se `origin_host` corresponde ao que o PCRF espera
3. **Verifique logs do PCRF:**
   - Procure por CER (Capabilities-Exchange-Request) do PGW-C
   - Procure razões de rejeição
4. **Verifique métricas:**

```
# Erros de conexão de Diâmetro
diameter_peer_connected{peer="<pcrf_host>"}
```

Resolução:

1. **Para problemas de rede:**

- Verifique o roteamento para o PCRF
- Verifique as regras do firewall: Certifique-se de que TCP 3868 está permitido
- Teste a conectividade: `nc -v <pcrf_ip> 3868`

2. **Para serviço PCRF fora do ar:**
   - Reinicie o serviço PCRF
   - Aguarde a reconexão automática (intervalo de tentativa de 30s)
   - Verifique via `GET /api/diameter?status=connected`
3. **Para incompatibilidade de configuração:**
   - Corrija a configuração de Diâmetro em `config/runtime.exs`:

```
config :pgw_c,
  diameter: %{
    host: "pgw-c.epc.mnc999.mcc999.3gppnetwork.org",  #
Deve corresponder à configuração do PCRF
    realm: "epc.mnc999.mcc999.3gppnetwork.org",
    peer_list: [
      %{
        host: "pcrf.epc.mnc999.mcc999.3gppnetwork.org",
        realm: "epc.mnc999.mcc999.3gppnetwork.org",
        ip: "192.168.1.100",
        initiate_connection: true
      }
    ]
  }
```

   - Reinicie o OmniPGW
   - Verifique se a conexão foi estabelecida

Prevenção:

- Monitore a conectividade do par de Diâmetro com alertas críticos
- Implemente instâncias redundantes do PCRF (se suportado)
- Documente a configuração de Diâmetro no runbook

Problema: Timeouts de CCR/CCA (Solicitações de Política Gx)

Sintoma:

- Estabelecimento de sessão lento (> 5 segundos)
- Logs: "timeout de Solicitação de Controle de Crédito"
- Métrica: `gx_outbound_transaction_duration` muito alta (> 5s)
- Sessões criadas com QoS padrão (comportamento de fallback)

Causas Prováveis:

1. PCRF sobrecarregado
2. Banco de dados do PCRF lento
3. Latência de rede
4. Problema de software do PCRF

Diagnóstico:

1. **Verifique a latência Gx:**

```
# Latência P95
histogram_quantile(0.95,
rate(gx_outbound_transaction_duration_bucket[5m]))

# Latência P99 (outliers)
histogram_quantile(0.99,
rate(gx_outbound_transaction_duration_bucket[5m]))
```

2. **Verifique a saúde do PCRF:**
    - Acesse painéis de monitoramento do PCRF
    - Verifique CPU, memória, conexões de banco de dados
    - Revise logs do PCRF para consultas lentas
3. **Verifique a latência da rede:**

```
ping -c 100 <pcrf_ip> | tail -1  # Verifique a latência média
```

4. **Verifique logs:**
    - Conte trocas CCR/CCA: Filtre "Credit Control"
    - Meça o tempo entre "Enviando CCR" e "Recebido CCA"

Resolução:

1. **Para sobrecarga do PCRF:**
    - Escale o PCRF (adicione instâncias)
    - Reduza o tamanho da mensagem CCR, se possível
    - Ajuste pools de threads/trabalhadores do PCRF
2. **Para latência de rede:**
    - Investigue o caminho da rede (roteadores, switches)
    - Considere co-localizar PGW-C e PCRF
3. **Solução temporária (aumentar timeout):**
    - Edite `config/runtime.exs`:
      ```
      config :pgw_c,
        diameter: %{
          transaction_timeout_ms: 10000  # Aumentar de 5000 para
      10000
        }
      ```
    - Reinicie o OmniPGW
    - **Nota:** Isso apenas mascara o problema; conserte a causa raiz

Prevenção:

- Monitore a latência Gx com alertas (aviso > 1s, crítico > 5s)
- Planeje a capacidade do PCRF para a taxa de sessões esperada
- Teste o desempenho do PCRF sob carga

Problema: Par OCS Desconectado (Gy)

Sintoma:

- `GET /api/diameter` (ou `GET /api/diameter?status=disconnected`) mostra par OCS "desconectado"; `GET /api/charging` reflete o status de cobrança online (Gy)
- Sessões não podem ser cobradas (cobrança online falha)
- Logs: "Falha na conexão do par Gy"

Diagnóstico e Resolução:

Semelhante ao Par PCRF Desconectado, mas para a interface Gy.

Principais diferenças:

- Porta: Normalmente TCP 3868 (mesma que Gx)
- Impacto: Cobrança falha, sessões podem ser rejeitadas ou permitidas sem cobrança (depende da configuração)
- Configuração: Verifique `diameter.peer_list` para entrada OCS

Veja: Interface Diâmetro Gy para solução de problemas específica do Gy

---

Problemas de Alocação de IP

Problema: Pool de IP Esgotado

Sintoma:

- Solicitação de Criação de Sessão rejeitada com causa "Sem recursos disponíveis"
- Métrica: `address_registry_count` igual ao tamanho do pool configurado
- `GET /api/sessions` mostra muitas sessões ativas; `GET /api/ip_pools` mostra o pool na capacidade
- Logs: "Falha na alocação de IP: pool esgotado"

Causas Prováveis:

1. Pool muito pequeno para a base de assinantes
2. Sessões não liberando IPs (falhas na Exclusão de Sessão)
3. Rotatividade rápida de sessões sem limpeza
4. Vazamento de endereço IP

Diagnóstico:

1. **Verifique a utilização do pool:**

```
# Para sub-rede /24 (254 IPs)
(address_registry_count / 254) * 100
```

2. **Verifique o tamanho do pool configurado:**
   - Revise `config/runtime.exs` sob `ue.subnet_map`
   - Exemplo: "10.0.0.0/24" = 254 IPs utilizáveis
3. **Compare a contagem de sessões com a contagem de IPs:**

```
# Deve ser aproximadamente igual
teid_registry_count
address_registry_count
```

4. **Revise sessões ativas:**
   - `GET /api/sessions` (e `GET /api/ip_pools/<name>` para utilização por pool)
   - Classifique por hora de início da sessão
   - Procure por sessões muito antigas (potenciais vazamentos)

Resolução:

Imediata (expansão do pool):

1. Edite `config/runtime.exs` (cada APN mapeia para uma **lista** de strings CIDR):

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => ["10.0.0.0/22"]  # 1022 IPs (era /24 = 254
IPs)
    }
  }
```

2. Reinicie o OmniPGW
3. Verifique: Sessões agora podem ser estabelecidas

A longo prazo (limpeza):

1. Identifique sessões antigas via `GET /api/sessions`
2. Coordene com SGW-C para enviar Solicitações de Exclusão de Sessão
3. Implemente política de timeout de sessão no PCRF/SGW
4. Monitore `address_registry_count` para verificar se o pool foi liberado após a limpeza

Prevenção:

- Monitore a utilização do pool de IP com alertas:
  - Aviso: > 70%
  - Crítico: > 85%
- Análise de tendências para prever esgotamento
- Implemente timeout de sessão ociosa
- Auditorias regulares de sessão

Problema: Endereço IP Duplicado Atribuído

Sintoma:

- UE relata conflito de endereço IP
- Logs: "IP já alocado" aviso
- Duas sessões em `GET /api/sessions` com o mesmo endereço IP

Causas Prováveis:

1. Bug de software (raro)
2. Inconsistência de banco de dados após falha
3. Erro de intervenção manual

Diagnóstico:

1. **Pesquise o IP via API OAM:**
   - `GET /api/sessions?search=<ip>`
   - Verifique se múltiplos IMSIs têm o mesmo IP
2. **Verifique logs:**
   - Pesquise pelo endereço IP
   - Procure por eventos de "alocação de IP"

Resolução:

1. **Identifique sessões afetadas:**
   - Anote ambos os IMSIs com IP duplicado
2. **Exclua uma sessão:**
   - Coordene com SGW-C para enviar Solicitação de Exclusão de Sessão para um IMSI
   - Prefira excluir a sessão mais nova
3. **UE reconecta:**
   - UE deve reconectar automaticamente
   - Receberá um novo IP único
4. **Se persistir:**
   - Reinicie o OmniPGW para reconstruir o registro de IP
   - Todas as sessões serão perdidas (coordene janela de manutenção)

Prevenção:

- Monitore alocações duplicadas (nenhuma métrica interna atualmente)

- Verificações regulares de integridade do banco de dados (se aplicável)

Problema: Sessão IPv4v6 em uma APN Sem Pool IPv6

Sintoma:

- Um UE solicita um PDN **IPv4v6**, mas a APN correspondente tem apenas um CIDR IPv4 configurado em `ue.subnet_map`
- Uma APN **de emergência (SOS)** aparece **somente IPv4** (sem IPv6 no PAA); logs mostram um `aviso` como "Sem pool IPv6 para APN de emergência ...; concedendo somente IPv4"
- Uma APN **regular** tem a sessão **rejeitada**; logs mostram um `erro` como "Sem pool IPv6 para APN ...; rejeitando solicitação IPv4v6"

Causas Prováveis:

1. A APN intencionalmente não tem faixa IPv6 (comum em implantações somente IPv4)
2. Um CIDR IPv6 foi omitido da entrada `ue.subnet_map` da APN por engano

Comportamento:

- O alocador IPv6 relata que não há pool IPv6 (não trava em um pool vazio).
- APNs de emergência (nome contém `"sos"`) degradam para somente IPv4 — endereço IPv4 mantido, IPv6 limpo, tipo de PDN definido como IPv4, e o P-CSCF ainda é retornado no PCO, de modo que o registro IMS de emergência funcione.
- APNs regulares são rejeitadas (sem degradação silenciosa), evidenciando a solicitação não suportada / má configuração.

Resolução:

1. Se a APN deve oferecer suporte a dual-stack, adicione um CIDR IPv6 à sua entrada `ue.subnet_map` e reinicie o OmniPGW.
2. Se o comportamento somente IPv4 for intencional (por exemplo, uma APN SOS em uma rede somente IPv4), nenhuma ação é necessária.

Veja solicitação IPv4v6 quando a APN não tem pool IPv6 e Alocação de IP do UE → Problema 4.

Referência Rápida

Consultas Comuns do Prometheus

```
# Sessões ativas
teid_registry_count

# Taxa de configuração de sessão (por segundo)
rate(s5s8_inbound_messages_total{message_type="create_session_request

# Utilização do pool de IP (para sub-rede /24)
(address_registry_count / 254) * 100

# Latência de configuração de sessão P95
histogram_quantile(0.95,
rate(s5s8_inbound_handling_duration_bucket{request_message_type="crea
[5m]))

# Taxa de erro
rate(s5s8_inbound_errors_total[5m])

# Latência do PCRF
histogram_quantile(0.95, rate(gx_outbound_transaction_duration_bucket

# Status da associação PFCP
pfcp_peer_associated
```

Filtros Comuns de Log

Grep essas strings nos logs do host (`journalctl -u omnipgw_c` ou o arquivo de log configurado). Para eventos do ciclo de vida da sessão, `GET /api/session_history[?type=&search=]` fornece as mesmas informações de forma estruturada.

Filtro	Propósito
IMSI	Encontrar todos os logs para um assinante específico
"create_session"	Fluxo de estabelecimento de sessão
"delete_session"	Fluxo de desmontagem de sessão
"Credit Control"	Interações Gx PCRF
"PFCP Session"	Programação do plano de usuário
"error"	Todas as mensagens de erro
"timeout"	Problemas de timeout
"Association"	Eventos de associação PFCP

Comandos de Verificação de Saúde

```
# Verifique o status do serviço
systemctl status omnipgw_c

# Verifique a saúde da API OAM (HTTPS/TLS)
curl -k https://<omnipgw_ip>:8443/api/status

# Verifique o endpoint de métricas
curl http://<omnipgw_ip>:9090/metrics

# Verifique sessões ativas
curl http://<omnipgw_ip>:9090/metrics | grep teid_registry_count

# Verifique a associação PFCP
curl http://<omnipgw_ip>:9090/metrics | grep pfcp_peer_associated

# Verifique o uso do pool de IP
curl http://<omnipgw_ip>:9090/metrics | grep
address_registry_count
```

Documentação Relacionada

- **Guia de Monitoramento** - Métricas do Prometheus, painéis do Grafana, alertas
- **Guia de Configuração** - Referência de configuração do sistema
- **Gerenciamento de Sessões** - Detalhes do ciclo de vida da sessão
- **Interface PFCP** - Detalhes de solução de problemas do PFCP
- **Interface Diâmetro Gx** - Solução de problemas de política Gx
- **Interface Diâmetro Gy** - Solução de problemas de cobrança Gy
- **QoS & Gerenciamento de Bearer** - Problemas relacionados a QoS

Voltar ao Guia de Operações

Documentação de Alocação de Pool de IP para UE

Gerenciamento de Endereços IP para Dispositivos Móveis

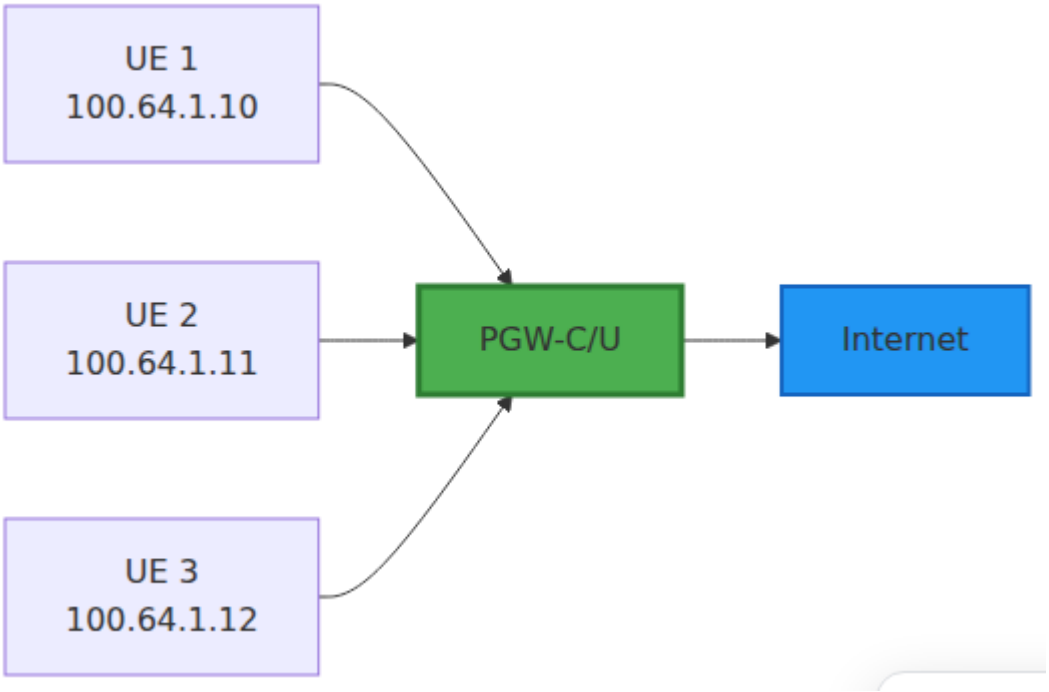
Índice

1. Visão Geral
2. Conceitos de Alocação de IP
3. Configuração
4. Processo de Alocação
5. Tópicos Avançados
6. Monitoramento
7. Solução de Problemas

Visão Geral

O PGW-C aloca endereços IP para dispositivos UE (Equipamento do Usuário) quando eles estabelecem conexões PDN (Rede de Dados por Pacote). Esta é uma função crítica que permite que dispositivos móveis se comuniquem com redes externas.

Por que a Alocação de IP é Importante



Cada UE recebe um **endereço IP único** do PGW-C que:

- Identifica o dispositivo na rede
- Roteia o tráfego para/de o dispositivo
- Permite cobrança e aplicação de políticas
- Persiste durante a duração da conexão PDN

Versões de IP Suportadas

Versão de IP	Suporte	Descrição
IPv4	□ Completo	Endereços IPv4 padrão
IPv6	□ Completo	Endereços e prefixos IPv6
IPv4v6	□ Completo	Dual-stack (IPv4 e IPv6)

Conceitos de Alocação de IP

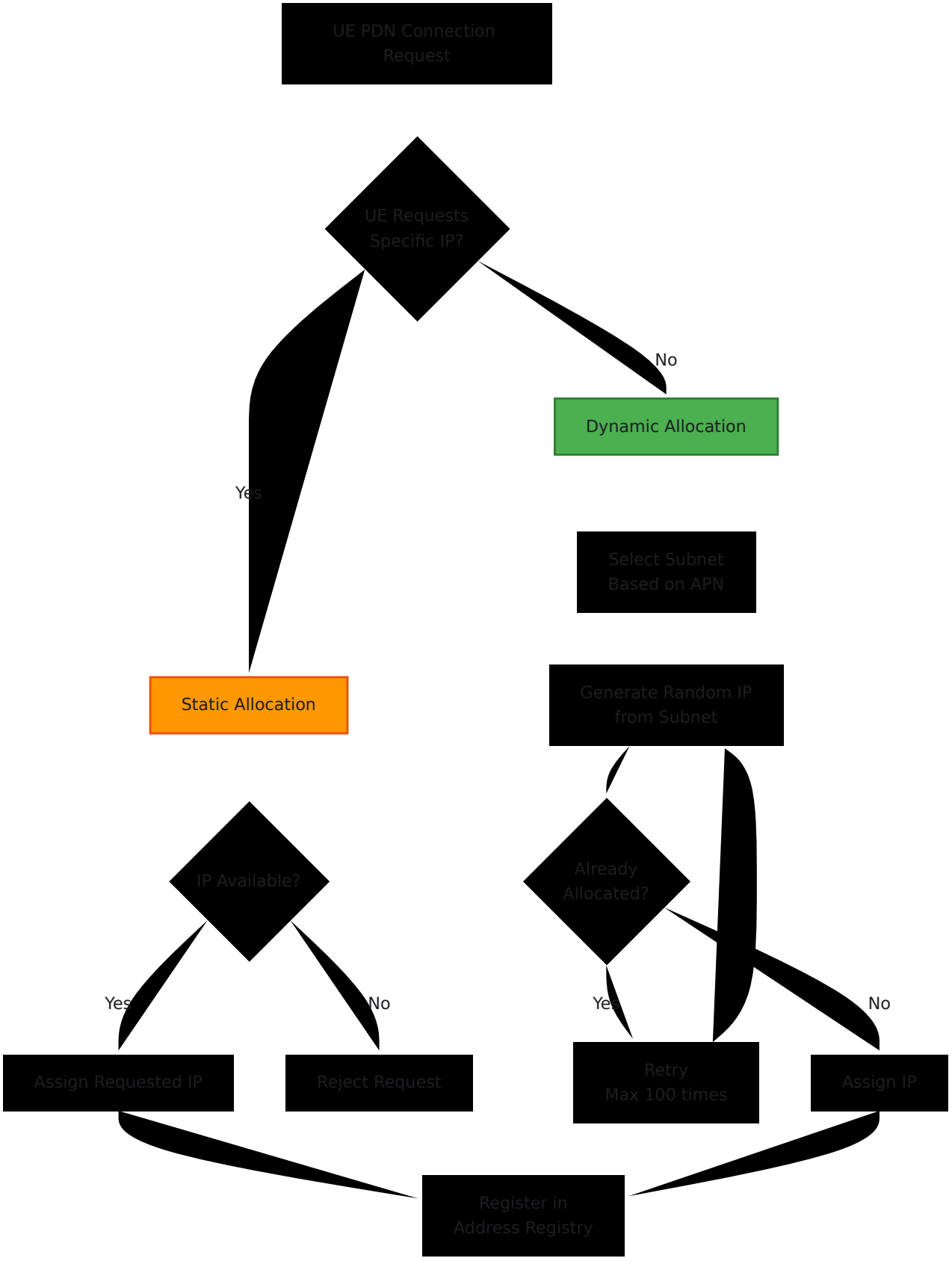
Tipo de PDN

Quando uma UE solicita uma conexão PDN, ela especifica um **Tipo de PDN**:

Tipo de PDN	Descrição	Endereços Alocados
IPv4	Conexão apenas IPv4	Endereço IPv4 único
IPv6	Conexão apenas IPv6	Prefixo IPv6 (ex: /64)
IPv4v6	Conexão dual-stack	Endereço IPv4 e prefixo IPv6

Métodos de Alocação

O PGW-C suporta dois métodos de alocação de IP:



1. Alocação Dinâmica (Mais Comum):

- O PGW-C seleciona IP do pool configurado
- Seleção aleatória para evitar previsibilidade

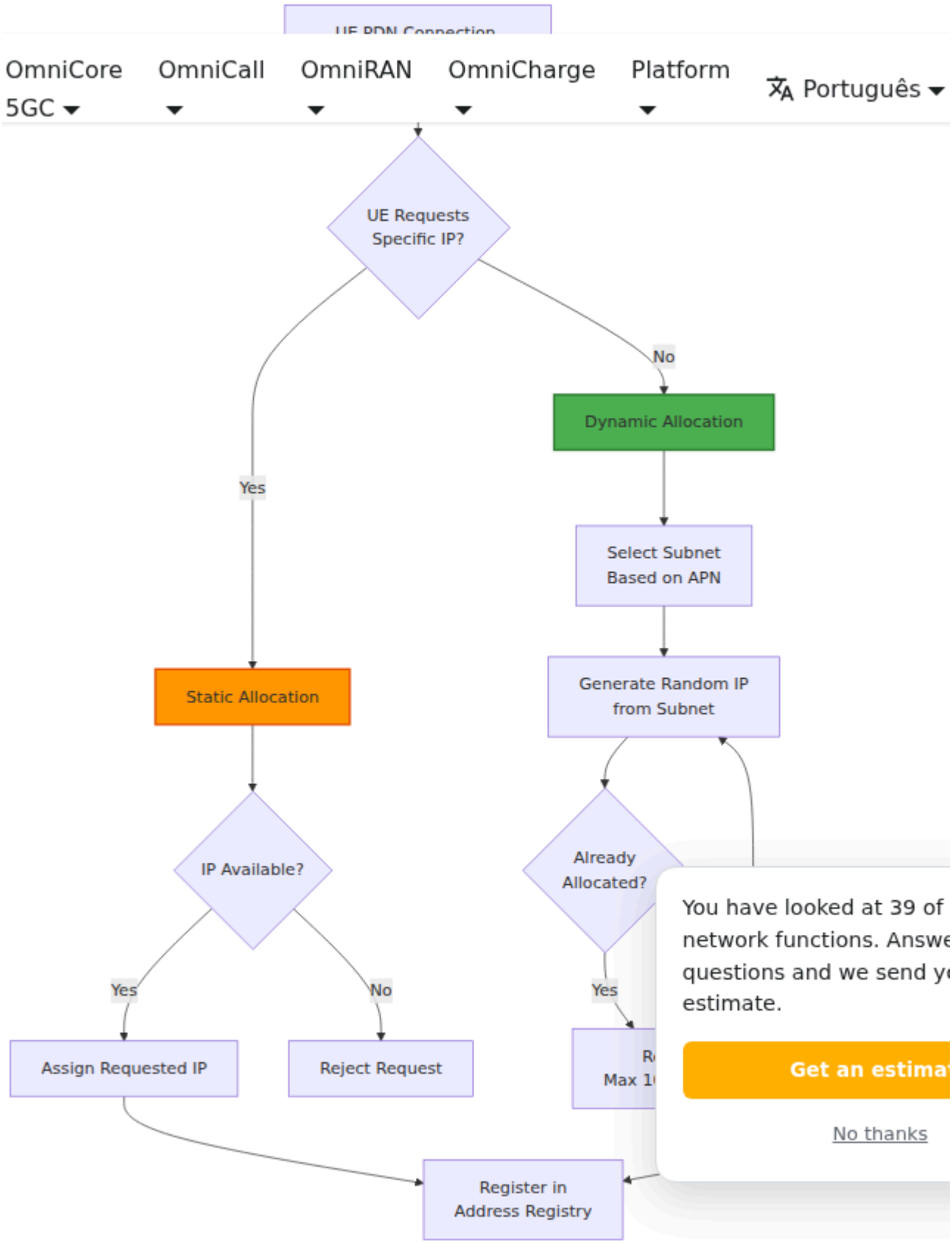
- Detecção de colisão garante exclusividade

2. Alocação Estática:

- UE solicita IP específico na mensagem GTP-C
- O PGW-C valida a disponibilidade
- Útil para dispositivos empresariais com IPs fixos

Seleção de Sub-rede Baseada em APN

Diferentes **APNs (Nomes de Ponto de Acesso)** podem usar diferentes pools de IP:



Benefícios:

- **Segregação de Tráfego** - Diferentes APNs roteiam para diferentes redes
- **Diferenciação de Políticas** - Aplicar políticas diferentes por APN

- **Planejamento de Capacidade** - Dimensionar pools com base no uso esperado
- **Cobrança** - Rastrear uso por tipo de serviço

Registro de Endereços

O **Registro de Endereços** rastreia IPs alocados:

Função	Descrição
Registro	Mapeia IP da UE → PID do Processo de Sessão
Consulta	Encontrar sessão pelo IP da UE
Desregistro	Liberar IP quando a sessão termina
Detecção de Colisão	Prevenir alocações duplicadas

Configuração

Configuração Básica

Edite `config/runtime.exs`:

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      # APN "internet" usa duas sub-redes
      "internet" => [
        "100.64.1.0/24",    # 254 IPs utilizáveis
        "100.64.2.0/24"     # 254 IPs utilizáveis
      ],

      # APN "ims" usa uma sub-rede
      "ims" => [
        "100.64.10.0/24"
      ],

      # Pool padrão para APNs desconhecidas
      default: [
        "42.42.42.0/24"
      ]
    }
  }
```

Correspondência de Padrões Regex para APNs

Para cenários onde múltiplas APNs compartilham o mesmo pool de sub-rede, você pode usar **padrões regex** em vez de nomes exatos de APN. Isso é útil para correspondência de APN com curinga.

Regras de Padrão:

- Chaves que começam com `^` são tratadas como padrões regex
- Chaves sem `^` são correspondidas exatamente (compatibilidade retroativa)
- Padrões são avaliados em ordem - o primeiro a corresponder vence
- Retorna ao `default` se nenhum padrão corresponder

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      # Regex: APNs começando com "ims" (ex: "ims", "ims.apn",
"ims.something.else")
      "^ims" => [
        "100.64.10.0/24"
      ],

      # Regex: APNs começando com "m2m." (ex: "m2m.test",
"m2m.prod")
      "^m2m\." => [
        "100.64.20.0/24"
      ],

      # Correspondência exata apenas - "enterprise.corp"
exatamente
      "enterprise.corp" => [
        "10.100.0.0/16"
      ],

      # Pool padrão para APNs não correspondidas
      default: [
        "42.42.42.0/24"
      ]
    }
  }
```

Notas Importantes:

- Padrões regex usam a sintaxe padrão de regex do Elixir/Erlang
- As barras invertidas devem ser escapadas em strings Elixir (use `\` para `\`)
- O `^` no início é necessário para indicar um padrão regex
- Correspondências exatas e padrões regex podem ser misturados na mesma configuração
- A ordem importa para padrões regex - coloque padrões mais específicos primeiro

Exemplos Comuns de Padrões:

Tipo de Padrão	Chave Regex	Correspondências
Começa com	`"^ims"`	`ims`, `ims.apn`, `ims.anything`
Termina com	`"^.*\.corp$"`	`foo.corp`, `bar.corp`
Contém	`"^.*test.*"`	`test`, `foo.test.bar`, `testing`
Exato (com pontos)	`"^internet\.apn$"`	`internet.apn` apenas

Exemplo de Correspondência de Sufixo:

Para corresponder APNs que terminam com um sufixo específico (como `.corp`), use `^.*\.suffix$`:

```
subnet_map: %{
  # Correspondência de APNs que terminam com ".corp"
  "^.*\.corp$" => ["10.100.0.0/16"],

  # Correspondência de APNs que terminam com ".iot"
  "^.*\.iot$" => ["10.200.0.0/16"],

  default: ["42.42.42.0/24"]
}
```

Exemplo de Correspondência de Padrão:

Solicitação de APN	Chave Correspondente	Pool Usado
`ims`	`^ims`	100.64.10.0/24
`ims.apn`	`^ims`	100.64.10.0/24
`ims.something.else`	`^ims`	100.64.10.0/24
`m2m.test`	`^m2m\.`	100.64.20.0/24
`m2m`	*default*	42.42.42.0/24
`enterprise.corp`	`enterprise.corp`	10.100.0.0/16
`foo.corp`	`^.*\.corp$`	10.100.0.0/16
`unknown.apn`	*default*	42.42.42.0/24

Notação de Sub-rede

Notação CIDR: `<rede>/<comprimento_do_prefixo>`

CIDR	IPs Utilizáveis	Faixa de Exemplo
/24	254	100.64.1.1 - 100.64.1.254
/23	510	100.64.0.1 - 100.64.1.254
/22	1022	100.64.0.1 - 100.64.3.254
/20	4094	100.64.0.1 - 100.64.15.254
/16	65534	100.64.0.1 - 100.64.255.254

Notas:

- O endereço de rede (ex: 100.64.1.0) não é alocado
- O endereço de broadcast (ex: 100.64.1.255) não é alocado
- O PGW-C aloca de `<rede> + 1` a `<broadcast> - 1`

Múltiplas Sub-redes por APN

Balanceamento de Carga Entre Sub-redes:

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => [
        "100.64.1.0/24",
        "100.64.2.0/24",
        "100.64.3.0/24",
        "100.64.4.0/24"
      ]
    }
  }
```

Método de Seleção:

- O PGW-C seleciona aleatoriamente uma sub-rede da lista
- Fornece balanceamento básico de carga
- Cada sessão seleciona uma sub-rede independentemente

Benefícios:

- Distribuir carga entre múltiplas sub-redes
- Expansão de capacidade mais fácil (adicionar novas sub-redes)
- Flexibilidade para políticas de roteamento

Exemplo do Mundo Real

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      # Acesso geral à internet
      "internet" => [
        "100.64.0.0/20"     # 4094 IPs para uso geral
      ],

      # IMS (Voz sobre LTE)
      "ims" => [
        "100.64.16.0/22"    # 1022 IPs para IMS
      ],

      # APN Empresarial
      "enterprise.corp" => [
        "10.100.0.0/16"     # 65534 IPs para empresas
      ],

      # Dispositivos IoT (baixa taxa de bits)
      "iot.m2m" => [
        "100.64.20.0/22"    # 1022 IPs para IoT
      ],

      # Fallback padrão
      default: [
        "42.42.42.0/24"     # 254 IPs para APNs desconhecidas
      ]
    }
  }
```

Configuração de IPv6

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => [
        # Pools IPv4
        "100.64.1.0/24"
      ],
      "internet.ipv6" => [
        # Pools IPv6 (delegação de prefixo)
        "2001:db8:1::/48"
      ],
      default: [
        "42.42.42.0/24"
      ]
    }
  }
```

Delegação de Prefixo IPv6:

- A UE normalmente recebe um prefixo /64
- Permite que a UE atribua múltiplos IPs (ex: para tethering)
- Exemplo: UE recebe `2001:db8:1:a::/64`

Configuração Dual-Stack (IPv4v6)

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => [
        "100.64.1.0/24",        # Pool IPv4
        "2001:db8:1::/48"       # Pool IPv6 (será usado para
alocação IPv6)
      ]
    }
  }
```

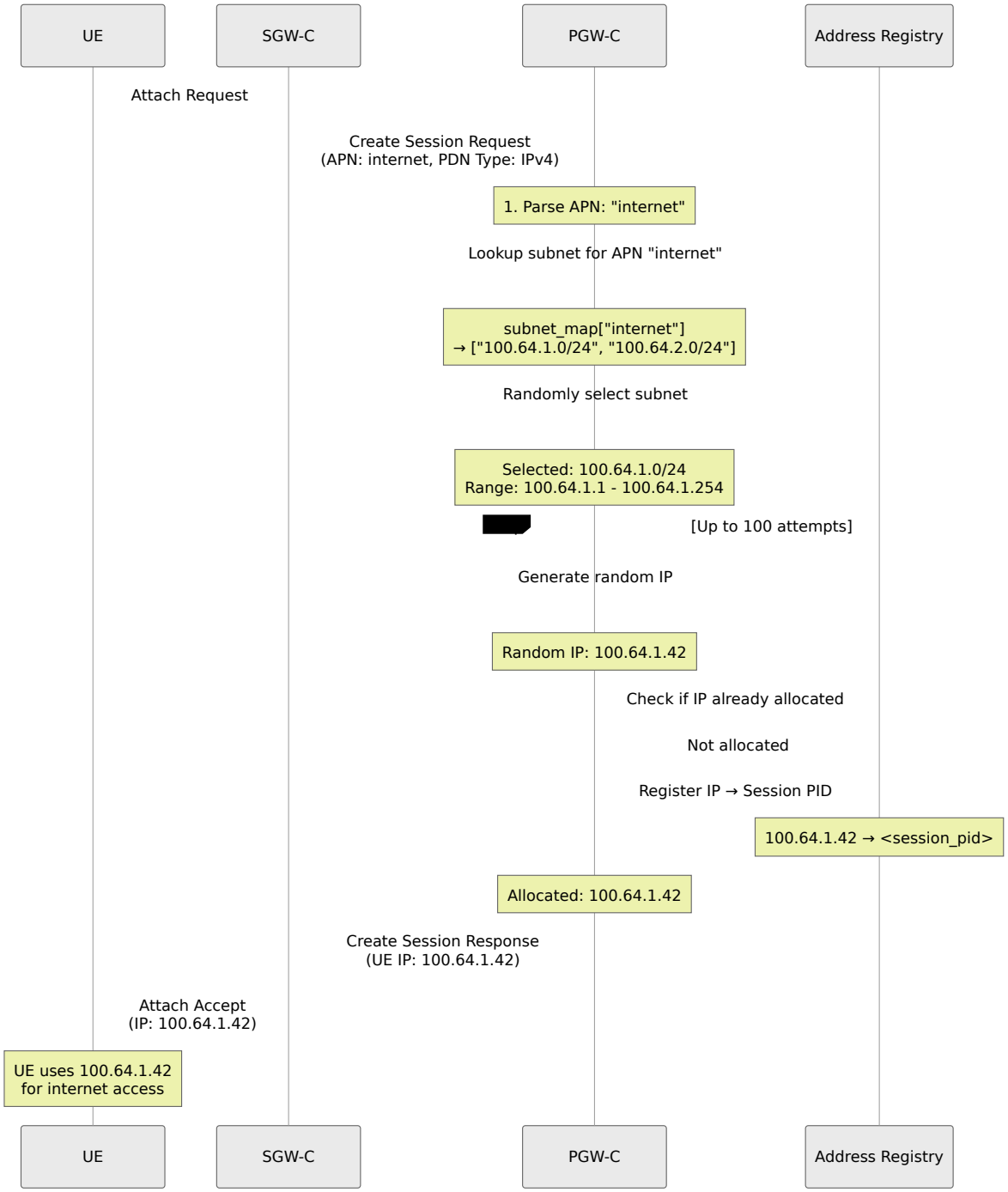
Alocação Dual-Stack:

- A UE solicita Tipo de PDN: IPv4v6
- O PGW-C aloca tanto o endereço IPv4 quanto o prefixo IPv6
- Ambos os endereços ativos simultaneamente

Processo de Alocação

A alocação de IP ocorre durante a criação da sessão quando o PGW-C recebe um Pedido de Criação de Sessão via a interface S5/S8. Veja Interface S5/S8 para detalhes da mensagem GTP-C e Gerenciamento de Sessão para o ciclo de vida da sessão.

Passo a Passo: Alocação Dinâmica de IPv4



Como Funciona

Processo de Alocação Dinâmica:

1. **Consulta de Sub-rede:** O sistema recupera sub-redes configuradas para a APN solicitada

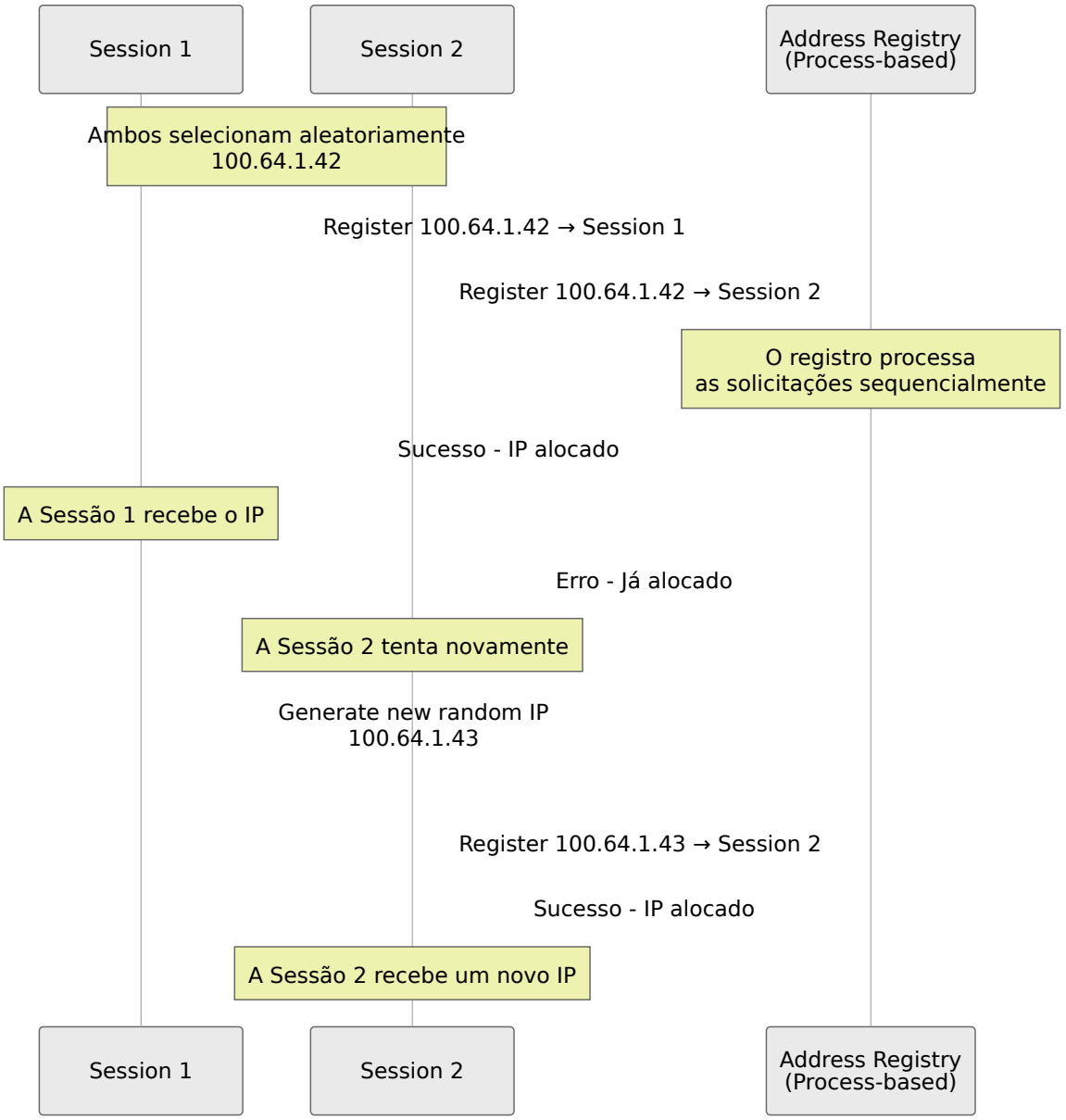
2. **Seleção Aleatória:** Uma sub-rede é selecionada aleatoriamente da lista disponível
3. **Geração de IP:** Um IP aleatório é gerado dentro da faixa da sub-rede
4. **Verificação de Exclusividade:** O sistema verifica se o IP não foi alocado
5. **Lógica de Retentativa:** Se uma colisão for detectada, tenta novamente até 100 vezes com um novo IP aleatório
6. **Registro:** Uma vez encontrado um IP único, ele é registrado na sessão

Pontos de Design Chave:

- **Máximo de 100 tentativas:** Previne loops infinitos quando o pool está quase esgotado
- **Seleção aleatória:** Evita padrões de atribuição de IP previsíveis para segurança
- **Operações atômicas:** O registro baseado em processos garante que não haja alocações duplicadas
- **Retorno ao padrão:** Se a APN não for encontrada na configuração, usa o pool padrão

Manipulação de Colisões

Cenário: Duas sessões tentam alocar o mesmo IP simultaneamente



Como a Prevenção de Colisões Funciona:

- O registro processa as solicitações uma a uma (serializado)
- Não há condições de corrida possíveis
- A primeira solicitação a registrar um IP tem sucesso
- Solicitações subsequentes para o mesmo IP são rejeitadas
- Sessões rejeitadas tentam automaticamente novamente com um novo IP aleatório

Retorno ao Pool Padrão

Cenário: UE solicita APN desconhecida

Exemplo de Configuração:

```
# Config
subnet_map: %{
  "internet" => ["100.64.1.0/24"],
  default: ["42.42.42.0/24"]
}
```

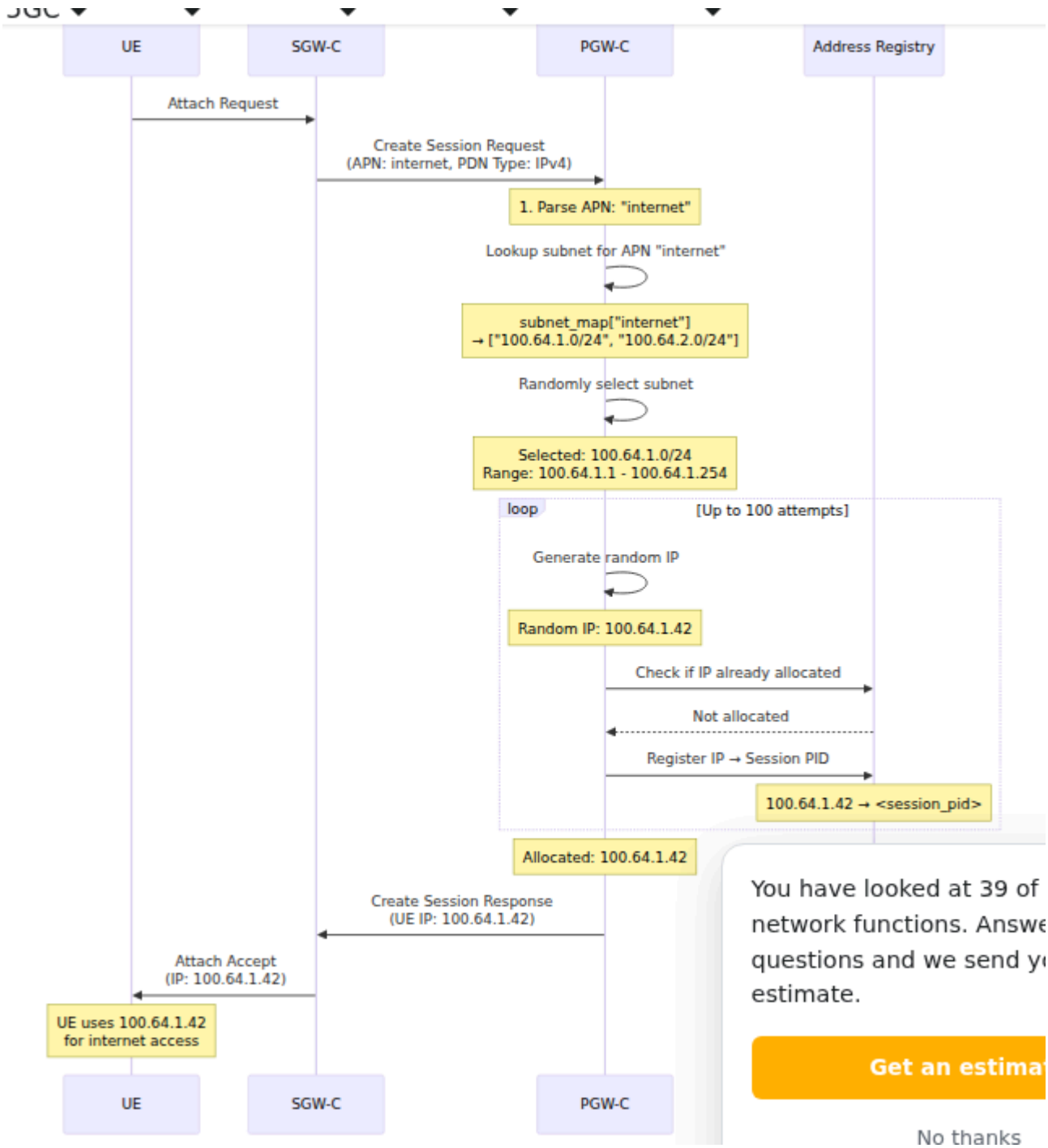
Comportamento:

- UE solicita APN: "unknown.apn"
- O sistema procura "unknown.apn" no subnet_map
- Não encontrado, então retorna ao pool padrão
- Aloca IP do 42.42.42.0/24

Lógica de Retorno:

1. Primeiro, tenta encontrar o pool específico da APN na configuração
2. Se não encontrado, usa o pool `default`
3. Se nenhum padrão configurado, a alocação falha

Desalocação na Término da Sessão



Limpeza Automática:

- Quando o processo da sessão termina, o registro faz a limpeza
- O IP fica imediatamente disponível para novas alocações
- Nenhuma intervenção manual necessária

Tópicos Avançados

Esgotamento do Pool

Cenário: Todos os IPs no pool estão alocados

```
Pool: 100.64.1.0/24 (254 IPs utilizáveis)
Alocados: 254 IPs
Nova solicitação chega → Esgotamento
```

O que Acontece:

1. O PGW-C tenta 100 alocações aleatórias
2. Todas as tentativas encontram IP já alocado
3. Retorna: `{:error, :ue_ip_address_allocation_failed}`
4. Estabelecimento da sessão falha
5. O SGW-C recebe a resposta de erro

Prevenção:

```
# Monitorar a utilização do pool
address_registry_count / total_pool_size > 0.8  # Alerta em 80%

# Expandir o pool antes do esgotamento
"internet" => [
  "100.64.1.0/24",
  "100.64.2.0/24",  # Adicionar sub-rede adicional
  "100.64.3.0/24"
]
```

Alocação Estática de IP

Caso de Uso: Dispositivo empresarial precisa de IP fixo

Formato da Mensagem GTP-C:

```
Create Session Request
├── IMSI: 310260123456789
├── APN: enterprise.corp
├── PDN Address Allocation (IE)
│   └── PDN Type: IPv4
│   └── IPv4 Address: 10.100.0.50  ← UE solicita IP específico
```

Processamento do OmniPGW:

1. **Extrair IP Solicitado:** Analisar IE de Alocação de Endereço PDN da solicitação
2. **Validar IP:** Verificar se o IP solicitado está no pool configurado para esta APN
3. **Verificar Disponibilidade:** Confirmar se o IP não está alocado a outra sessão
4. **Alocar ou Rejeitar:**
   - Se disponível: Alocar IP solicitado a esta sessão
   - Se indisponível: Rejeitar a sessão com o código de causa apropriado

Resultados Possíveis:

- **Sucesso:** UE recebe exatamente o endereço IP que solicitou
- **Falha (IP em uso):** Sessão rejeitada - IP já alocado
- **Falha (IP não no pool):** Sessão rejeitada - IP não está na faixa configurada

Delegação de Prefixo IPv6

UE solicita IPv6:

```
Create Session Request
├── PDN Type: IPv6
```

PGW-C aloca prefixo /64:

```
Prefixo Alocado: 2001:db8:1:a::/64

UE pode usar:
- 2001:db8:1:a::1
- 2001:db8:1:a::2
- ... (18 quintilhões de endereços)
```

Benefícios:

- UE pode atribuir múltiplos IPs (ex: tethering)
- Suporta SLAAC (Configuração Automática de Endereço Stateless)
- Elimina a necessidade de NAT

Alocação Dual-Stack

UE solicita IPv4v6:

```
Create Session Request
├── PDN Type: IPv4v6
```

PGW-C aloca ambos:

```
IPv4: 100.64.1.42
IPv6: 2001:db8:1:a::/64
```

Manipulação de Tráfego:

- O tráfego IPv4 usa o endereço IPv4
- O tráfego IPv6 usa o prefixo IPv6
- Ambos ativos simultaneamente
- Túneis GTP separados (ou túnel dual-stack)

Quando a APN não tem pool IPv6:

Se um PDN IPv4v6 for solicitado, mas a APN correspondente tiver apenas um CIDR IPv4 (sem faixa IPv6), o alocador de IPv6 retorna `{:error,`

`:no_ipv6_pool}` em vez de falhar, e o procedimento da sessão decide pelo tipo de APN:

Tipo de APN	Resultado
APN de Emergência (SOS) (nome contém `"sos"`)	**Degradar para apenas IPv4** — mantém o endereço IPv4, limpa o endereço IPv6 e define o tipo de PDN negociado como IPv4. O PAA, PFCP e Gx saem todos apenas IPv4, e o P-CSCF ainda é retornado no PCO para que o registro IMS de emergência possa prosseguir (TS 29.274 / TS 23.401 bearer de endereço único / downgrade de tipo de PDN).
APN Regular	**Rejeitar** — a solicitação não é degradada silenciosamente; a má configuração (ou solicitação não suportada) é exposta.

Solicitações apenas IPv6 contra uma APN sem pool IPv6 também são rejeitadas. Veja Solicitação IPv4v6 quando a APN não tem pool IPv6 no guia IPv6 / Dual-Stack para a tabela completa de comportamento.

Endereços IP Privados vs. Públicos

Pools de IP Privados (RFC 1918):

```
# Não roteável na internet pública
subnet_map: %{
  "internet" => [
    "10.0.0.0/8",
    "172.16.0.0/12",
    "192.168.0.0/16"
  ]
}
```

Requer NAT no PGW-U para acessar a internet

Pools de IP Públicos:

```
# IPs públicos roteáveis (apenas exemplo)
subnet_map: %{
  "internet" => [
    "203.0.113.0/24"  # Bloco de IP público
  ]
}
```

Nenhum NAT necessário - roteamento direto para a internet

Recomendação:

- Use **IPs privados (RFC 6598)**: `100.64.0.0/10` (NAT de Grau de Transportadora)
- Reserve IPs públicos apenas para serviços especiais

Monitoramento

Gerenciamento de Pool de IP

A alocação e utilização do pool de IP são monitoradas através da API REST OAM (HTTPS/TLS na porta `8443`, rotas sob o prefixo `/api`). A documentação interativa (Swagger UI) está disponível em `https://<omnipgw-ip>:8443/api/docs`.

```
GET /api/ip_pools            # todos os pools + utilização
GET /api/ip_pools/<name>     # um pool pelo nome
```

Exemplo:

```
curl -k https://<omnipgw-ip>:8443/api/ip_pools
curl -k https://<omnipgw-ip>:8443/api/ip_pools/Internet
```

Informações por Pool retornadas para cada pool configurado:

Campo	Descrição
Nome	Identificador do Pool / APN (ex: "default", "ims.something.else", "Internet")
Total	Número de IPs no pool
Usado	IPs atualmente alocados
Livre	IPs restantes disponíveis para alocação
Utilização %	Percentual do pool atualmente alocado

Casos de Uso:

- Verificação rápida de capacidade antes da manutenção
- Identificar pools se aproximando do esgotamento
- Verificar configuração do pool
- Monitorar padrões de alocação por APN

Métricas Chave

Contagem do Registro de Endereços:

```
# IPs atualmente alocados
address_registry_count

# Utilização do pool (requer cálculo)
address_registry_count / <total_pool_size> * 100
```

Exemplo:

```
Pool: 100.64.1.0/24 (254 IPs)
Alocados: 150 IPs
Utilização: 150 / 254 = 59%
```

Alertas

```
# Alerta sobre alta utilização do pool
- alert: UEIPPoolUtilizationHigh
  expr: address_registry_count > 200  # Para pool /24
  for: 10m
  annotations:
    summary: "Utilização do pool de IP da UE acima de 80%"
    description: "Atual: {{ $value }} / 254 IPs alocados"

# Alerta sobre esgotamento do pool
- alert: UEIPPoolExhausted
  expr: address_registry_count >= 254  # Para pool /24
  for: 1m
  annotations:
    summary: "Pool de IP da UE esgotado - nenhum IP disponível"

# Alerta sobre falhas de alocação
- alert: UEIPAllocationFailures
  expr: rate(ue_ip_allocation_failures_total[5m]) > 0
  for: 5m
  annotations:
    summary: "Falhas de alocação de IP da UE ocorrendo"
```

Painel do Grafana

Painel 1: Utilização do Pool de IP

```
# Gauge mostrando porcentagem
(address_registry_count / 254) * 100
```

Painel 2: IPs Alocados ao Longo do Tempo

```
# Série temporal
address_registry_count
```

Painel 3: Taxa de Alocação

```
# Taxa de novas alocações
rate(address_registry_count[5m])
```

Painel 4: Risco de Esgotamento do Pool

```
# Dias até o esgotamento (com base na taxa atual)
(254 - address_registry_count) / rate(address_registry_count[1h])
```

Solução de Problemas

Problema 1: Estabelecimento de Sessão Falha (Nenhum IP Disponível)

Sintomas:

- Resposta de Criação de Sessão: Causa "Solicitação rejeitada"
- Log: "Falha na alocação do endereço IP da UE"

Possíveis Causas:

1. **Pool Esgotado**

   ```
   # Verificar alocação atual
   curl http://<pgw_c_ip>:42069/metrics | grep
   address_registry_count
   ```

2. **Erro de Configuração**

```
# Verificar configuração da sub-rede
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => [
        "100.64.1.0/24"  # Garantir CIDR válido
      ]
    }
  }
```

3. **Má Configuração de APN**

```
# Se a APN não for encontrada, retorna ao padrão
# Garantir que o pool padrão exista
subnet_map: %{
  default: ["42.42.42.0/24"]
}
```

Resolução:

- **Expandir pool:** Adicionar mais sub-redes
- **Limpar sessões obsoletas:** Reiniciar o PGW-C para liberar IPs vazados
- **Verificar configuração:** Checar `runtime.exs` em busca de erros de digitação

Problema 2: Colisão de Endereço IP

Sintomas:

- Duas UEs recebem o mesmo IP (muito raro)
- Problemas de roteamento

Causa:

- Bug no Registro de Endereços (não deveria acontecer)

Depuração:

```
# Verificar IPs duplicados nos logs
grep "already_registered" /var/log/pgw_c.log
```

Resolução:

- Deve se corrigir automaticamente (segunda sessão tenta novamente)
- Se persistente, reportar bug

Problema 3: Pool de IP Errado Usado

Sintomas:

- UE recebe IP de sub-rede inesperada
- APN "internet" obtém IP do pool "ims"

Causa:

- Configuração incorreta do subnet_map

Verificar:

```
# Verificar correspondência exata da string APN
subnet_map: %{
  "internet" => [...],    # Sensível a maiúsculas
  "Internet" => [...]     # APN diferente!
}
```

Resolução:

- Garantir que os nomes das APNs correspondam exatamente (sensível a maiúsculas)
- Usar pool padrão para captura

Problema 4: Falha na Alocação de IPv6

Sintomas:

- UE solicita IPv6, recebe erro
- Uma UE solicita um **IPv4v6** PDN contra uma APN sem faixa IPv6 e recebe uma sessão **apenas IPv4** (APN SOS) ou tem a sessão **rejeitada** (APN regular)

Possíveis Causas:

1. **Nenhum pool IPv6 configurado**

```
# Sub-redes IPv6 ausentes
subnet_map: %{
  "internet" => [
    "100.64.1.0/24"  # Apenas IPv4
  ]
}
```

2. **Prefixo IPv6 inválido**

```
# Prefixo muito pequeno (deve ser /48 ou maior)
"internet" => [
  "2001:db8::/128"  # Errado - sem espaço para alocação
]
```

IPv4v6 sem pool IPv6 (degradar vs. rejeitar):

Quando uma solicitação **IPv4v6** atinge uma APN sem faixa IPv6, `register_new_ipv6/1` retorna `{:error, :no_ipv6_pool}` (não falha). A sessão então:

- **APN de Emergência (SOS)** (nome contém `"sos"`): degrada para **apenas IPv4** — atribui o endereço IPv4, limpa o endereço IPv6, define o tipo de PDN negociado como IPv4 e ainda retorna o P-CSCF no PCO. Registrado como um `warning`.
- **APN Regular**: **rejeita** com `{:error, :no_ipv6_pool}` — sem downgrade silencioso. Registrado como um `error`.

Veja Solicitação IPv4v6 quando a APN não tem pool IPv6.

Resolução:

```
# Adicionar pool IPv6 (para uma APN regular que deve oferecer
dual-stack)
subnet_map: %{
  "internet" => [
    "100.64.1.0/24",
    "2001:db8:1::/48"  # Pool IPv6
  ]
}
```

Para uma APN de emergência (SOS) em uma implantação apenas IPv4, a degradação para apenas IPv4 é intencional — adicione um CIDR IPv6 apenas se PDNs SOS devem ser dual-stack.

Problema 5: Alta Utilização do Pool

Sintomas:

- Aproximando-se do esgotamento do pool
- `address_registry_count` se aproximando do máximo

Medidas Proativas:

1. **Adicionar Sub-redes:**

   ```
   "internet" => [
     "100.64.1.0/24",   # Existente
     "100.64.2.0/24",   # Nova sub-rede (adiciona 254 IPs)
     "100.64.3.0/24"    # Nova sub-rede (adiciona 254 IPs)
   ]
   ```

2. **Usar Sub-redes Maiores:**

   ```
   # Substituir /24 por /22
   "internet" => [
     "100.64.0.0/22"  # 1022 IPs utilizáveis
   ]
   ```

3. **Limpeza de Sessões:**
    - Monitorar sessões obsoletas
    - Garantir o manuseio adequado da Solicitação de Exclusão de Sessão

---

Melhores Práticas

Planejamento de Capacidade

Calcular o tamanho do pool necessário:

```
Usuários simultâneos esperados: 10.000
Pico de concorrência: 30% (3.000 sessões simultâneas)
Buffer de crescimento: 50%
IPs necessários: 3.000 * 1.5 = 4.500 IPs

Sub-rede: /20 (4.094 IPs utilizáveis) - Muito pequeno
Sub-rede: /19 (8.190 IPs utilizáveis) - Suficiente
```

Seleção de Sub-rede

Recomendado:

- Usar `100.64.0.0/10` (RFC 6598 - NAT de Grau de Transportadora)
- Fornece 4 milhões de IPs
- Reservado para NAT de provedores de serviços

Evitar:

- IPs públicos (caros, limitados)
- Faixas privadas comuns que conflitam com VPNs empresariais

Layout de Configuração

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      # APN principal de internet - pool grande
      "internet" => [
        "100.64.0.0/18"  # 16.382 IPs
      ],

      # IMS - pool dedicado menor
      "ims" => [
        "100.64.64.0/22"  # 1.022 IPs
      ],

      # Empresa - pool médio
      "enterprise.corp" => [
        "100.64.68.0/22"  # 1.022 IPs
      ],

      # IoT - pool grande para muitos dispositivos
      "iot.m2m" => [
        "100.64.72.0/20"  # 4.094 IPs
      ],

      # Padrão - fallback pequeno
      default: [
        "100.64.127.0/24"  # 254 IPs
      ]
    }
  }
```

Documentação Relacionada

Configuração

- **Guia de Configuração** - Configuração do pool de IP da UE, mapeamento de sub-rede APN

- **Configuração PCO** - DNS, P-CSCF, MTU entregues com endereço IP
- **Gerenciamento de Sessão** - Ciclo de vida da sessão, alocação de IP durante a configuração do PDN
- **Interface PFCP** - Atribuição de endereço UE via PFCP para UPF

Planejamento de Rede

- **Interface S5/S8** - Entrega de endereço IP via GTP-C
- **Interface Diameter Gx** - Controle de políticas para alocação de IP

Operações

- **Guia de Monitoramento** - Métricas de utilização do pool de IP, rastreamento de alocação
- **Formato de CDR de Dados** - Endereços IP da UE em CDRs para correlação de cobrança

---

Voltar ao Guia de Operações