

OmniPGW

runtime.exs □□□□□□

□ Omnitouch □□□□□□

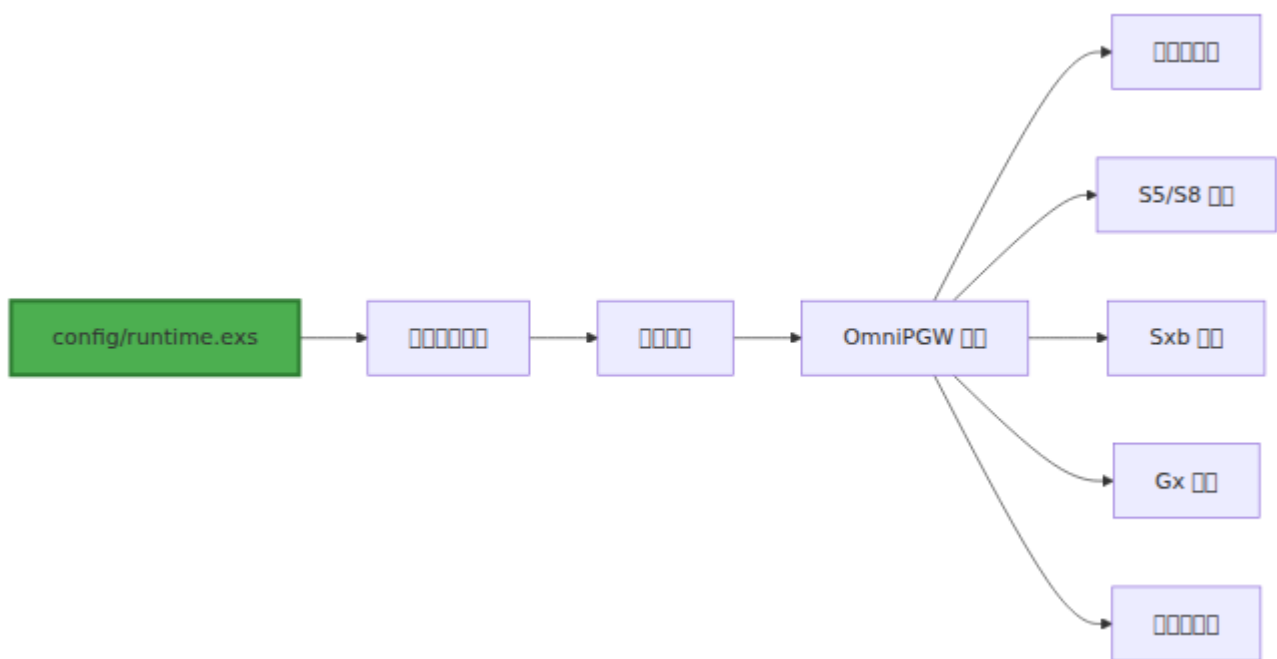
11

1. **□□**
2. **□□□□□□**
3. **□□□□**
4. **Diameter/Gx □□**
5. **S5/S8 □□**
6. **Sxb/PFCP □□**
 - **UPF □□□□**
 - **□□ UPF □□□□□□**
 - **□□ DNS □□□**
 - **□□□□□□**
7. **UE IP □□□**
8. **PCO □□**
9. **Web UI □□**
10. **□□□□**
11. **□□□□**

11

```
OmniPGW 0000 config/runtime.exe 000000 00000000 0000000000000000
0000000000000000
```

□□□□



□□□□□

- □□□□□□ - □□□□□□□□□□□□
- □□□□□ - □□□□□□□□
- □□□□□□ - □□□□□□□□□□□□
- □□□□□□□□ - □□□□□□□□□□□□□□

□□□□□□□□

□□□□□

```
pgw_c/
├─ config/
│   ├── config.exs      # □□□□□□□□ runtime.exs□
│   ├── dev.exs        # □□□□□□□□
│   ├── prod.exs       # □□□□□□□□
│   └─ runtime.exs     # ← □□□□□□□□
```

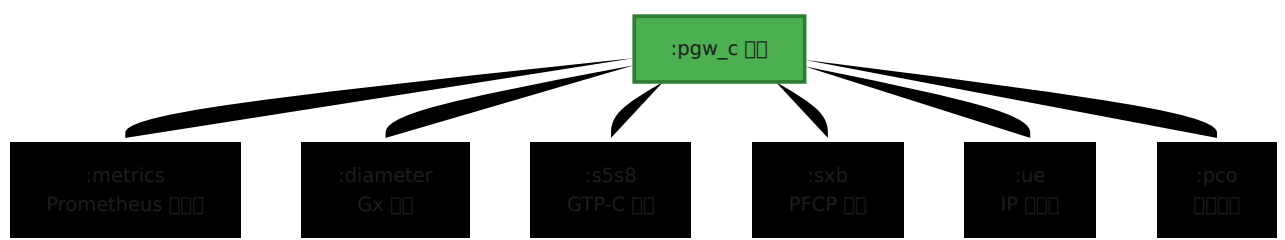
配置

```
# config/runtime.exs
import Config

config :logger, level: :info

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

架构图



部署

部署

部署 Prometheus 和 OmniPGW

配置

```
config :pgw_c,
  metrics: %{
    # 是否开启指标
    enabled: true,

    # 监听 HTTP 请求 IP 地址
    ip_address: "0.0.0.0",

    # 监听端口
    port: 9090,

    # 注册中心轮询周期
    registry_poll_period_ms: 10_000
  }
```

参数

参数	类型	默认值	说明
<code>enabled</code>	布尔	<code>true</code>	是否开启指标
<code>ip_address</code>	字符串 IP地址	<code>"0.0.0.0"</code>	监听地址 0.0.0.0 = 所有地址
<code>port</code>	整数	<code>9090</code>	<code>/metrics</code> 监听 HTTP 端口
<code>registry_poll_period_ms</code>	整数	<code>10_000</code>	注册中心轮询周期

部署

编译 - 生成可执行 IP

```
metrics: %{
  enabled: true,
  ip_address: "10.0.0.20", # 10.0.0.20
  port: 9090,
  registry_poll_period_ms: 5_000 # 5 000ms
}
```

10 - 10.0.0.20

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

10.0.0.1

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

10.0.0.1

```
# 10.0.0.1
curl http://<ip_address>:<port>/metrics

# 10
curl http://10.0.0.20:9090/metrics
```

10.0.0.1 10.0.0.20 10.0.0.1

Diameter/Gx 配置

配置

配置 Diameter 服务器 Gx 接口 PCRF 服务器

配置

```
config :pgw_c,  
  diameter: %{  
    # Diameter 服务器 IP 地址  
    listen_ip: "0.0.0.0",  
  
    # OmniPGW 与 Diameter 服务器 Origin-Host  
    host: "omnipgw.epc.mnc001.mcc001.3gppnetwork.org",  
  
    # OmniPGW 与 Diameter 服务器 Origin-Realm  
    realm: "epc.mnc001.mcc001.3gppnetwork.org",  
  
    # PCRF 服务器  
    peer_list: [  
      %{  
        # PCRF Diameter 服务器  
        host: "pcrf.epc.mnc001.mcc001.3gppnetwork.org",  
  
        # PCRF 服务器  
        realm: "epc.mnc001.mcc001.3gppnetwork.org",  
  
        # PCRF IP 地址  
        ip: "10.0.0.30",  
  
        # 是否 PCRF 服务器  
        initiate_connection: true  
      }  
    ]  
  }  
}
```

表

項目	項目	項目	項目
listen_ip	PCRF IP	項目	Diameter 項目項目項目
host	PCRF FQDN	項目	OmniPGW 項目 Origin-Host項目項目 FQDN
realm	項目項目項目	項目	OmniPGW 項目 Origin-Realm
peer_list	項目	項目	PCRF 項目項目項目

項目項目項目

項目	項目	項目	項目
host	PCRF FQDN	項目	PCRF Diameter 項目
realm	項目項目項目	項目	PCRF 項目
ip	PCRF IP	項目	PCRF IP 項目
initiate_connection	項目	項目	OmniPGW 項目項目項目 PCRF

FQDN 項目

Diameter 項目項目項目 FQDN

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

# 項目
host: "omnipgw" # 項目 FQDN
host: "10.0.0.20" # 項目項目項目 IP
```

3GPP 項目

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

例

- omnigw.epc.mnc001.mcc001.3gppnetwork.org (MCC=001, MNC=001)
- pgw-c.epc.mnc260.mcc310.3gppnetwork.org (MCC=310, MNC=260 - 例 T-Mobile)

例

例 **PCRF**

```
diameter: %{  
  listen_ip: "0.0.0.0",  
  host: "omnigw.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  
    }  
  ]  
}
```

例 **PCRF**例

```
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
    }
  ]
}
```

PCRF 配置

```
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 # PCRF 不连接
    }
  ]
}
```

配置 Diameter Gx 接口

S5/S8

GTP-C SGW-C

```
config :pgw_c,  
  s5s8: %{  
    # S5/S8 IPv4  
    local_ipv4_address: "10.0.0.20",  
  
    # IPv6  
    local_ipv6_address: nil,  
  
    # GTP-C 2123  
    local_port: 2123,  
  
    # GTP-C 500  
    # GTP-C  
    request_timeout_ms: 500,  
  
    # GTP-C 3  
    # = request_timeout_ms * request_attempts  
    request_attempts: 3  
  }
```

이

필드	타입	값	설명
local_ipv4_address	문자열 IPv4		S5/S8 노드 IPv4 주소
local_ipv6_address	문자열 IPv6	nil	S5/S8 노드 IPv6 주소 (선택)
local_port	정수	2123	GTP-C 포 UDP 포트 번호 2123
request_timeout_ms	정수	500	요청 시간 제한 (ms)
request_attempts	정수	3	요청 시도 횟수

이

- GTP-C 포트 2
- UDP
- 포트 2123
- 노드 SGW-C

이

이 IPv4

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

IPv4 + IPv6

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

配置示例

```
s5s8: %{\n  local_ipv4_address: "10.0.0.20",\n  local_port: 2124 # 端口\n}
```

配置项

```
s5s8: %{\n  local_ipv4_address: "10.0.0.20",\n  request_timeout_ms: 1500, # 请求超时 1.5 秒\n  request_attempts: 3      # 请求次数 4.5 秒\n}
```

说明

S5/S8 配置项与 GTP-C 配置项/参数项对应关系如下

计算公式

请求次数 = request_timeout_ms × request_attempts
1500ms × 3 = 4.5 秒

对应关系

配置项	配置项	配置项
请求次数 < 50ms	200-300ms	600-900ms
请求次数 50-150ms	500ms	1.5s
请求次数 > 150ms	1000-2000ms	3-6s
配置项/参数	2000-3000ms	6-9s

Wireshark

- Wireshark 安裝及設定
- Wireshark 抓包及分析
- Wireshark 抓包及分析

PCRF

- PCRF 抓包及分析
- PCRF 抓包及分析
- PCRF 抓包及分析

IPsec

IPsec

- IPsec 抓包及分析
- IPsec 抓包及分析
- IPsec 抓包及分析

iptables

```
# iptables -A INPUT -p udp --dport 2123 -s <sgw_c_network> -j ACCEPT
```

Sxb/PFCP

PF

PF PFCP 抓包及分析

配置

```
config :pgw_c,
  sxb: %{
    # PFCP 服务器 IP 地址
    local_ip_address: "10.0.0.20",

    # 服务器 PFCP 端口8805
    local_port: 8805
  }
```

表

变量	类型	默认值	描述
local_ip_address	字符串IP		PFCP 服务器
local_port	整数	8805	PFCP UDP 端口

配置

- 配置 **UPF** 服务器地址 `upf_selection` 格式为 `IP + 端口`
- 配置 UPF 服务器名称
 - 格式为 `"UPF-<ip>:<port>"`
 - 配置 PFCP 服务器 UPF 名称
 - 5 个字节
- UPF 服务器地址 `upf_selection` 格式为 `IP 端口`
- 配置 UPF 服务器名称 DNS 名称 UPF

表

配置

```
sxb: %{  
  local_ip_address: "10.0.0.20"  
}  
  
# [] upf_selection [] UPF []  
# - []`"UPF-10.0.0.21:8805"`  
# - [] PCFP [] UPF []  
# - 5 []
```

[] **PCFP** []

```
sxb: %{  
  local_ip_address: "10.0.0.20",  
  local_port: 8806 # [] PCFP []  
}
```

[] **UPF** []

```
sxb: %{
  local_ip_address: "10.0.0.20"
},
upf_selection: %{
  rules: [
    %{
      name: "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}
  ]
}
# 3 UPF 10.0.1.21, 10.0.1.22, 10.0.2.21
```

DNS

```
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}
  ]
}
# DNS UPF
```

UPF 部署

UPF 部署架构图 UPF 部署 `upf_selection` 部署图

部署

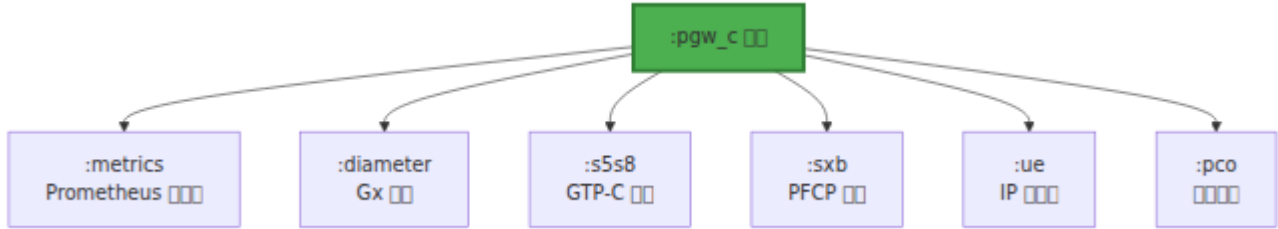
UPF 部署 `upf_selection` 部署图

1. 部署 - 部署图
2. **DNS** 部署 - 部署图 UPF 部署
3. 部署 - 部署图 DNS 部署图

部署图

1. 部署图 - 部署图
2. 部署 **DNS** 部署图 - 部署图 UPF 部署
3. 部署图 - 部署图 DNS 部署图

UPF 部署图



部署图

部署图

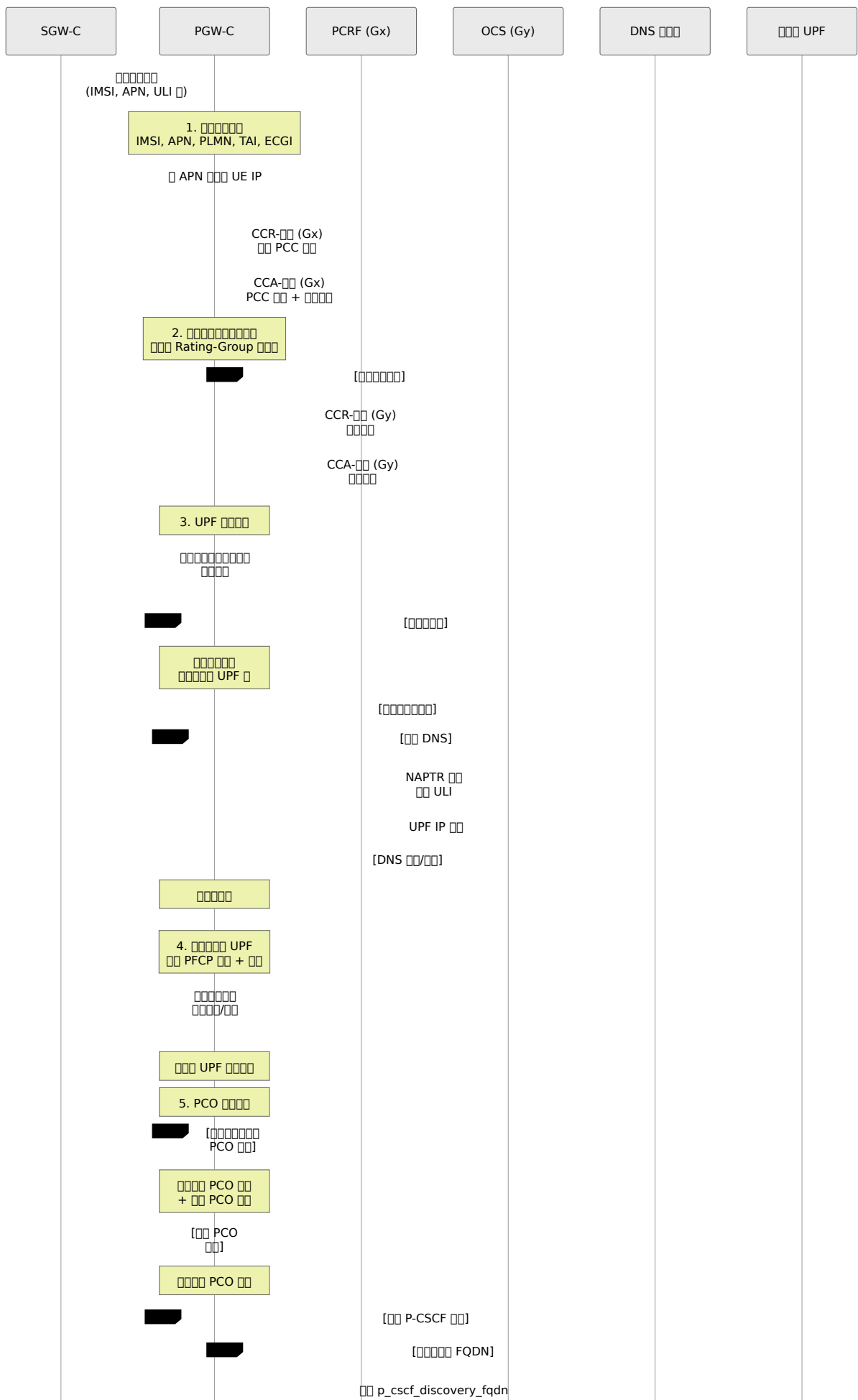
項目名	説明	形式
:imsi	IMSI	^313380.*
:apn	APN / DNN	^internet\. ^ims\.
:serving_network_plmn_id	PLMN ID	^313380\$
:sgw_ip_address	SGW IP	^10\.100\..*
:uli_tai_plmn_id	PLMN ID	^313.*
:uli_ecgi_plmn_id	E-UTRAN PLMN ID	^313.*

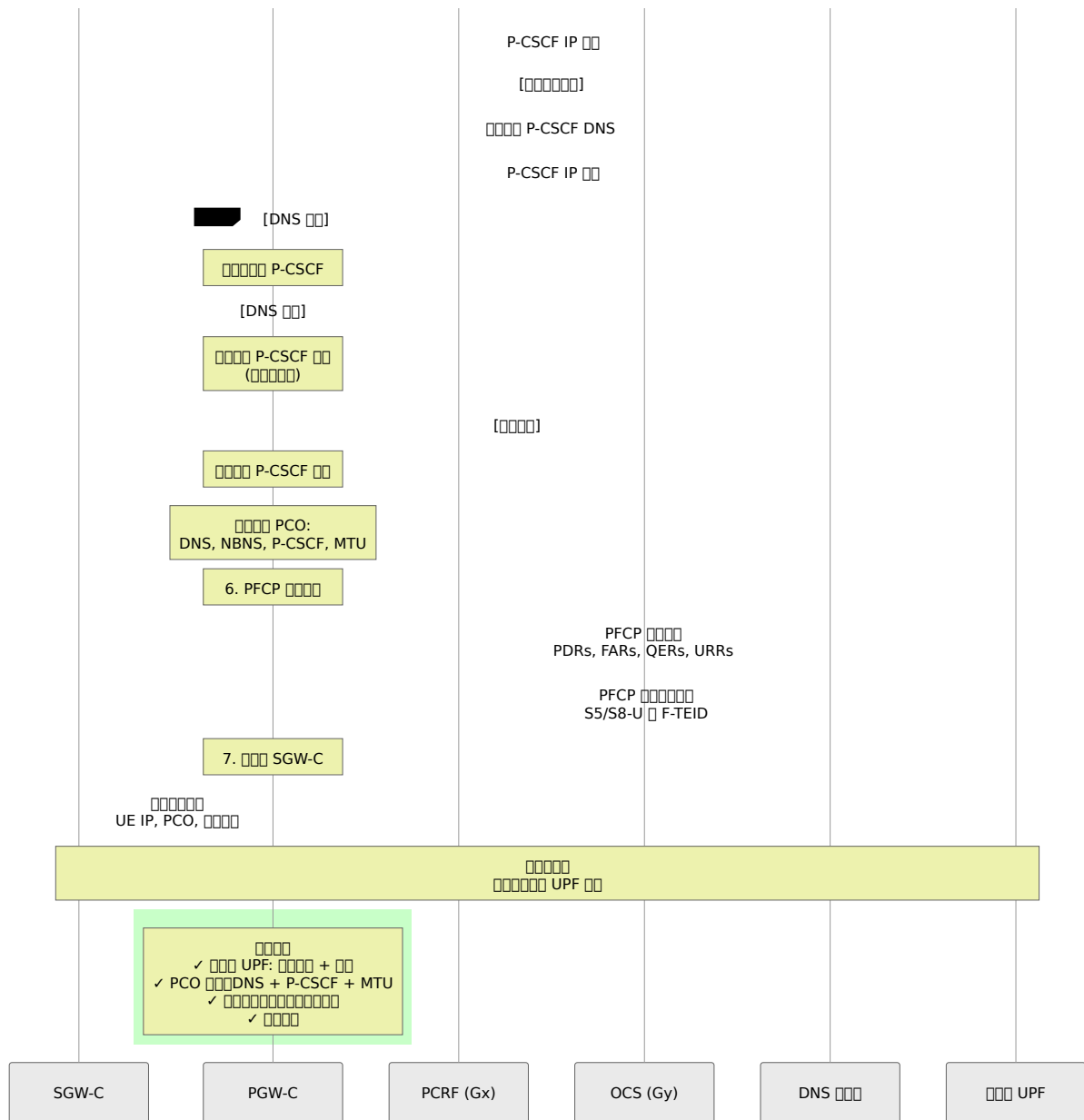
設定項目

項目名	説明	形式	値
UPF	UPF	IPアドレス	UPF
APN	APN	IMS/	
IMSI	IMSI		
DNS	MEC/		DNS
		UPF	

□□□□□□□□

□□□□□□□□□□□□□□□□□□ UPF □□□ PCO □□□





프로세스

1. UPF 프로세스

- 프로세스 흐름 → DNS → UPF
- DNS → UPF
- UPF/프로세스 흐름
- UPF PFCP → UPF 프로세스

2. PCO 프로세스

- PCO → P-CSCF DNS → PCO
- 프로세스 흐름

- 返回 PCO 给 终端 PCO 给

3. P-CSCF 返回

- 返回 FQDN → 给 DNS 给 → 返回 PCO → 返回 PCO
- 返回 P-CSCF 给 终端

4. 返回

- PCRF 返回 Rating-Group + Online=1
- OCS 返回
- PGW-C 返回 CCR-Update 给
- 返回 Diameter Gx 给 Diameter Gy 给 终端

返回

返回 UPF 返回

```

config :pgw_c,
  # PFCP [] - [] UPF [] upf_selection []
  sxb: %{
    local_ip_address: "127.0.0.20"
  },

  # UPF [] - [] UPF []
  upf_selection: %{
    # [] DNS []
    dns_enabled: false,
    dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],
    dns_suffix: "epc.3gppnetwork.org",
    dns_timeout_ms: 5000,

    # []
    rules: [
      # [] 1 IMS [] - []
      %{
        name: "IMS []",
        priority: 20,
        match_field: :apn,
        match_regex: "^ims",
        upf_pool: [
          weight: 80,
          %{remote_ip_address: "10.100.2.21", remote_port: 8805,
          weight: 20}
          %{remote_ip_address: "10.100.2.22", remote_port: 8805,
          weight: 20}
        ]
      },

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

      # [] 3 [] - []

```

```

    %{
      name: "互联网",
      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}
      ]
    },

    # 备用 - 本地 DNS 服务器
    fallback_pool: [
      %{remote_ip_address: "127.0.0.21", remote_port: 8805,
weight: 100}
    ]
  }

```

配置

配置

- 在 `upf_selection` 中 UPF 配置
- 在 `UPF` 中配置
- 在 `upf_pool` 中配置 UPF
- 在 `fallback_pool` 中配置
- 在 **DNS** 中配置 DNS 服务器
- 在 `DNS` 中配置 UPF 配置
- 在 `UPF` 中配置 5 配置

配置

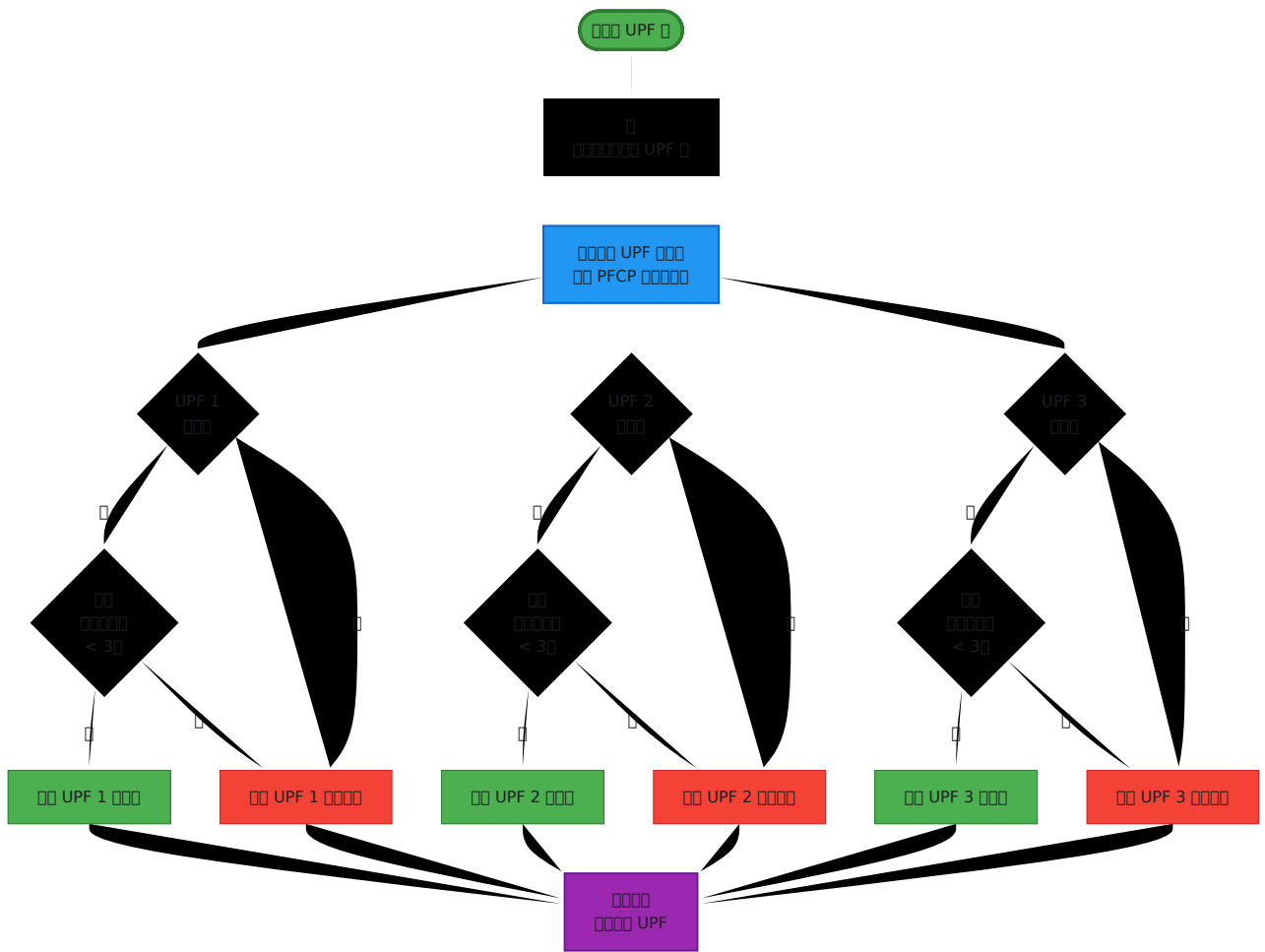
- 在 `sxb.peer_list` 中配置
- 在 `selection_list` 中配置

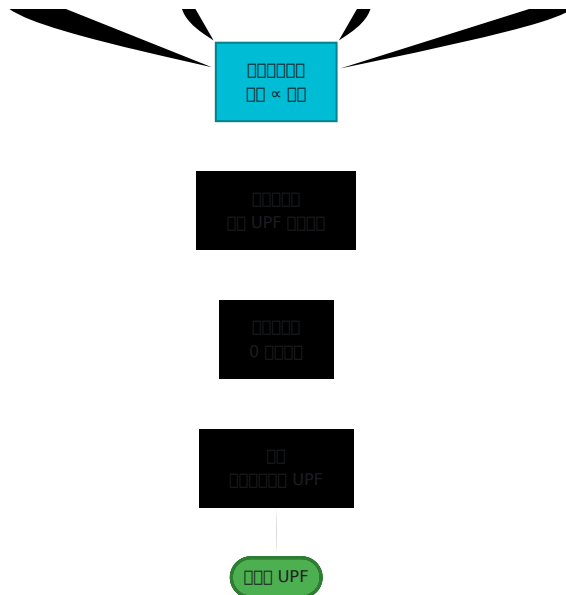
- 5G UPF 5QoS 5QI `upf_selection` 5QI

UPF □□□□□□□□

1. **UPF 部署**
 - **UPF = PCF + 3 个 UPF**
 - **UPF 部署**
 - **UPF 部署**
2. **UPF 部署** `weight: 0` **UPF**
 - **UPF 部署** `> 0`
 - **UPF 部署** `= 0` **UPF 部署**
 - **UPF 部署** `weight: 1`
3. **UPF 部署**
 - **UPF 部署** `70%` **UPF 部署** `.21` `20%` **UPF 部署** `.22` `10%` **UPF 部署** `.23`
 - **UPF 部署** **UPF 部署**
 - **UPF 部署** `=` **UPF 部署**
4. **UPF 部署**
 - **UPF 部署** `"UPF-<ip>:<port>"`
 - **UPF 部署** **PCF 部署** `5` **UPF 部署**
 - **UPF 部署** **UPF 部署**

□□□□□□□□/□□





透射率透射率透射率

透射率: [
UPF-A: 透射率 50, 透射率 ✓
UPF-B: 透射率 30, 透射率 ✓
UPF-C: 透射率 20, 透射率 ✓
]

透射率: 50 + 30 + 20 = 100

透射率:
UPF-A: 0-49 (50%)
UPF-B: 50-79 (30%)
UPF-C: 80-99 (20%)

透射率: 63 → 透射率 UPF-B
透射率: 15 → 透射率 UPF-A
透射率: 91 → 透射率 UPF-C

透射率/透射率透射率透射率

```
UPF: [
  UPF-A: 100,  ✓  ( )
  UPF-B: 0,    ✓  ( )
]
```

1: UPF-A

- : [UPF-A: 100]
- UPF-A

2: UPF-A

- UPF
- : [UPF-B: 1]
- UPF-B
- UPF

3:

- UPF
- : [UPF-A: 100, UPF-B: 0]
-
- UPF

```

# 25%
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}
]

# 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}
]

# 100% 0%
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: 0}
]

#
upf_pool: [
    %{remote_ip_address: "10.0.1.1", remote_port: 8805, weight: 100}, #
    %{remote_ip_address: "10.0.1.2", remote_port: 8805, weight: 0},
    # 1
    %{remote_ip_address: "10.0.1.3", remote_port: 8805, weight: 0}
    # 2
]
# 100% 50/50%

# A/B 50% / 50%
upf_pool: [
    %{remote_ip_address: "10.0.1.100", remote_port: 8805, weight: 50}, #
    %{remote_ip_address: "10.0.1.200", remote_port: 8805, weight: 50} #
]

```

000

- [illegible]

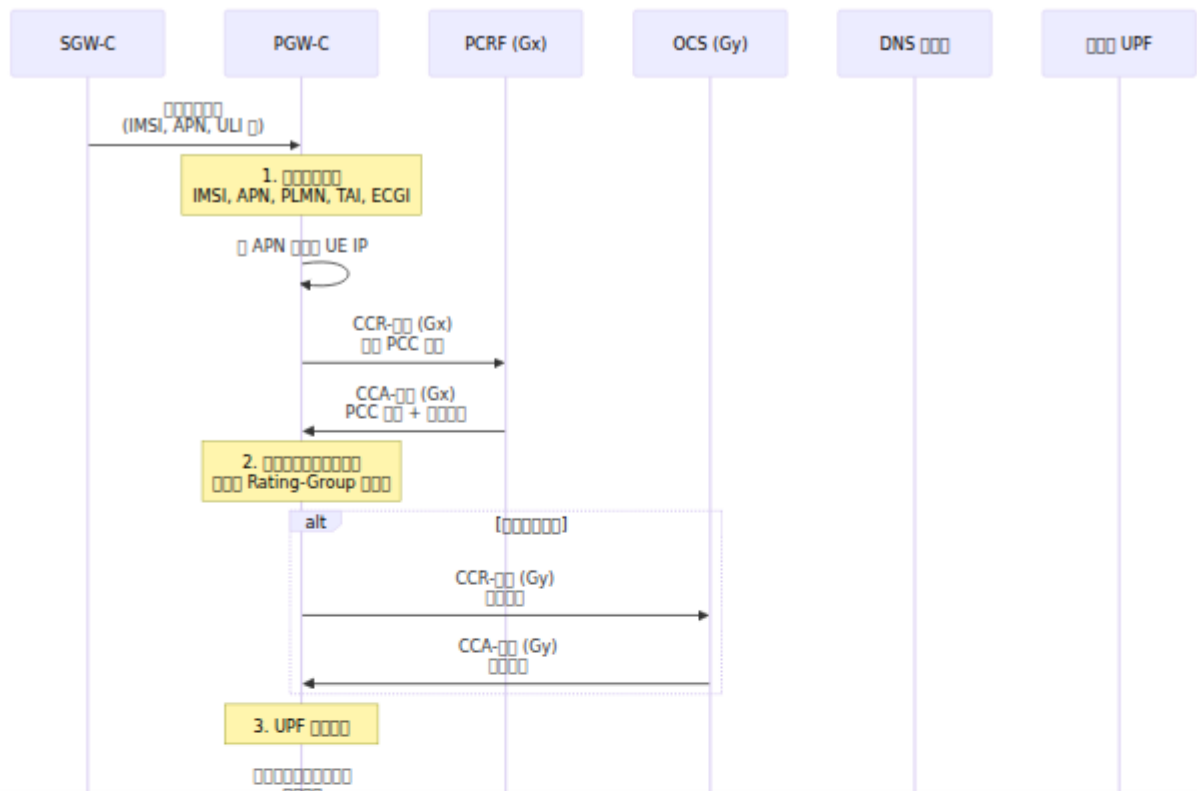
PCO□□□□□□□□□□

00 UPF 0000000000000000 PCO 0000000000000000 PCO 0000000000 APN 0000000000
 000000

PCO

1. 環境問題の解決には PCO が必要
2. 環境問題の解決には **pco** が必要
3. 環境問題の解決には PCO が必要
4. 環境問題の解決には PCO が必要、PCO が必要

PCO □□□□



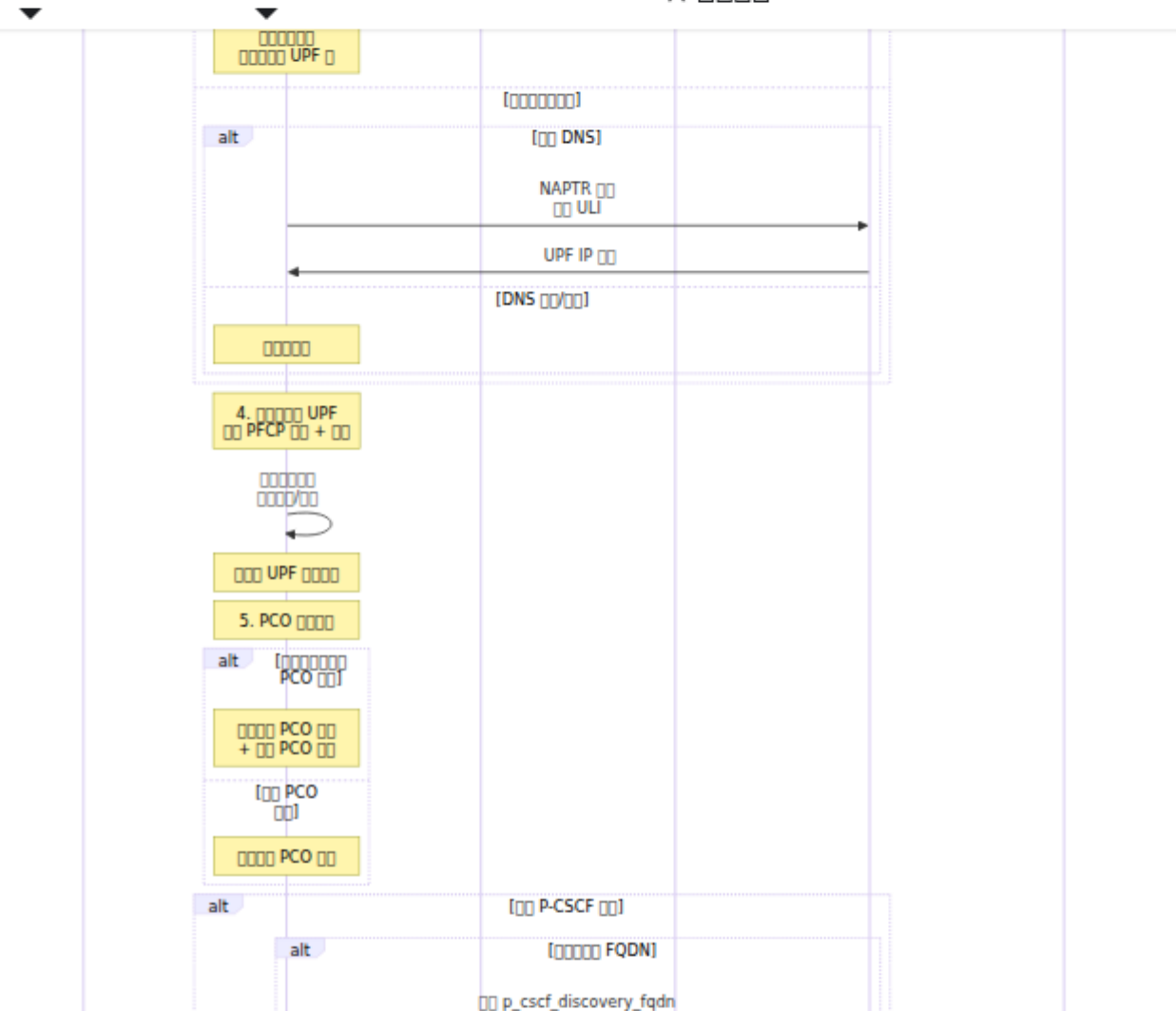
OmniCharge

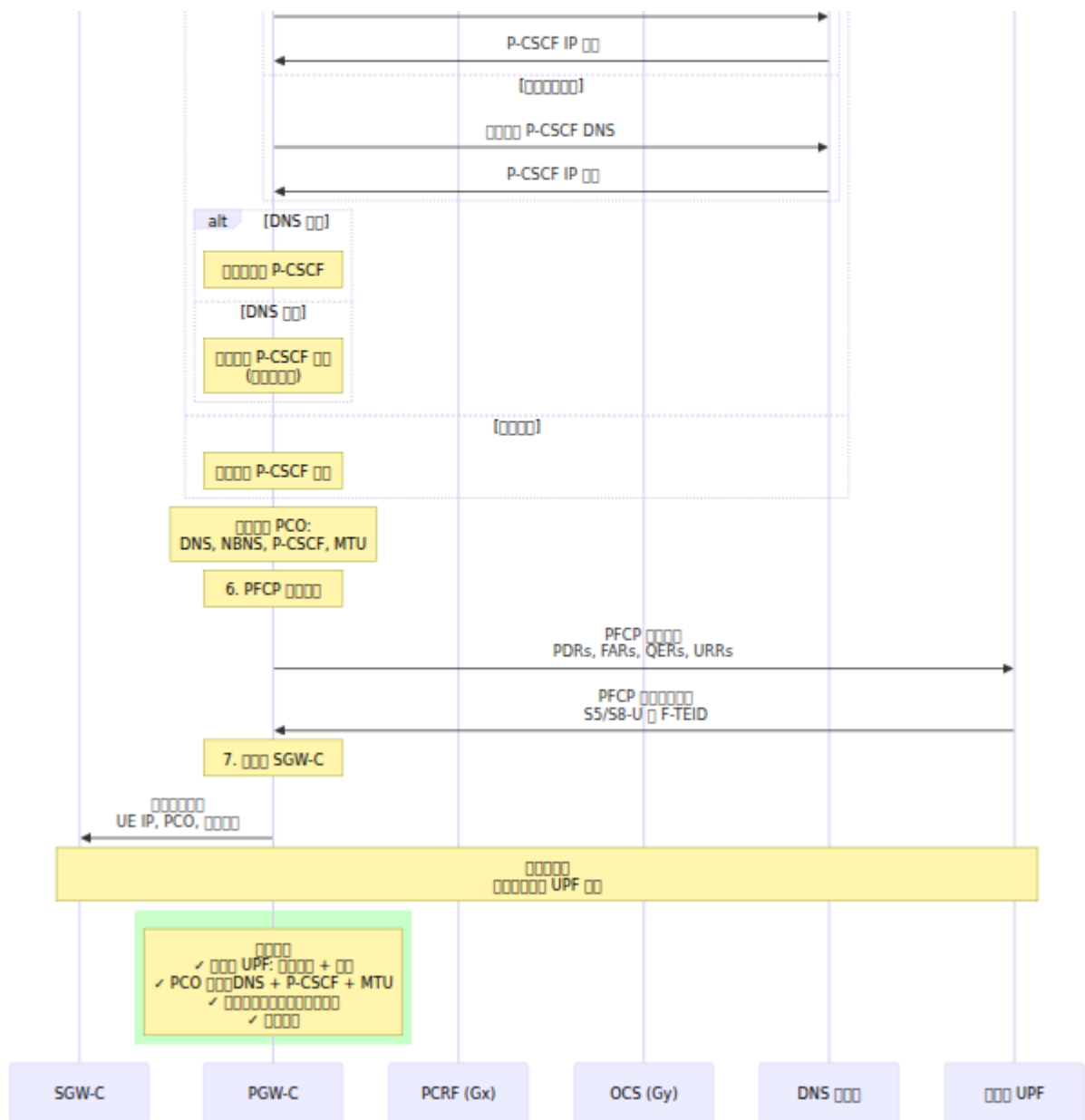
OmniRAN

Downloads

⌕ ☰ ☰ ☰ ▼

Omnitouch Website [↗](#)





PCO ()

1. PCO ()
2. P-CSCF DNS () P-CSCF ()
3. PCO () / ()

IMS () DNS ()

IMS 配置“IMS 配置”配置

- └ DNS 配置
- └ P-CSCF 配置 DNS 配置 p_cscf_discovery_fqdn
 - └ 配置 DNS 配置
- └ MTU 配置

配置“配置”配置

- └ DNS 配置 192.168.1.10, 192.168.1.11
- └ P-CSCF 配置
- └ MTU 配置 1500

配置配置

- └ DNS 配置
- └ P-CSCF 配置 DNS 配置
- └ MTU 配置

配置 **PCO** 配置

- primary_dns_server_address - DNS 配置 IP
- secondary_dns_server_address - DNS 配置 IP
- primary_nbns_server_address - WINS 配置 IP
- secondary_nbns_server_address - WINS 配置 IP
- p_cscf_ipv4_address_list - P-CSCF 配置 IP 配置 IMS - 配置 PCO 配置 P-CSCF 配置 P-CSCF 配置
- ipv4_link_mtu_size - MTU 配置

配置 **P-CSCF** 配置

PCO 通過UPF 與P-CSCF 通信

- `p_cscf_discovery_fqdn` - 通過DNS 獲 P-CSCF 的 FQDN
如 `"pcscf.mnc380.mcc313.3gppnetwork.org"`

過程如下

1. PGW-C 通過DNS 獲 FQDN
2. DNS 獲 P-CSCF IP
3. 通過 P-CSCF 與 PCO 通信 UE
4. 通過 DNS 獲 PCO 的 `p_cscf_ipv4_address_list` PCO 的 IP
5. 通過 **P-CSCF** 與 PCO 通信

過程如下

- **IMS APN** - 通過IMS 與 P-CSCF 通信
- 通過 - 通過 P-CSCF 通信
- 通過 - DNS 獲 UE 通過 P-CSCF
- 通過 - DNS 獲 P-CSCF 的 IP

通過**IMS** 與 **P-CSCF**

```

rules: [
  %{
    name: "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}
    ],
    # P-CSCF  DNS  P-CSCF 
    # DNS  FQDN  P-CSCF IP
    p_cscf_discovery_fqdn: "pcscf.mnc380.mcc313.3gppnetwork.org",
    # IMS  P-CSCF  DNS 
    pco: %{
      p_cscf_ipv4_address_list: ["10.101.2.100", "10.101.2.101"]
      # DNSNBNSMTU  pco 
    }
  }
]

```

DNS

```

rules: [
  %{
    name: "企业",
    priority: 15,
    match_field: :apn,
    match_regex: "^(enterprise|corporate)\.apn",
    upf_pool: [
      %{remote_ip_address: "10.100.3.21", remote_port: 8805,
weight: 100}
    ],
    # 企业 DNS 企业 MTU
    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 企业 pco 企业
    }
  }
]

```

企业 PCO 企业

```

rules: [
  %{
    name: "IoT APN - 物联网",
    priority: 10,
    match_field: :apn,
    match_regex: "^iot\\.m2m",
    upf_pool: [
      %{remote_ip_address: "10.100.5.21", remote_port: 8805,
weight: 100}
    ],
    # IoT 物联网 PC0
    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: [], # IoT 物联网 P-CSCF
      ipv4_link_mtu_size: 1280      # 物联网 MTU
    }
  }
]

```

物联网

- **IMS/VoLTE** 物联网 P-CSCF 物联网
- 物联网 **APN** 物联网 DNS 物联网
- **IoT/M2M** 物联网 DNS 物联网 MTU 物联网
- 物联网 DNS 物联网
- 物联网

物联网 **DNS** 物联网 **UPF** 物联网

物联网 ULI 物联网 UPF 物联网 DNS NAPTR 物联网 DNS 物联网 `upf_selection` 物联网

物联网 UPF 物联网 **PF** 物联网 UPF 物联网 **PF** 物联网

```
upf_selection: %{
  # DNS
  dns_enabled: true,

  #
  dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],

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

  # DNS
  dns_timeout_ms: 5000,

  # ...
}
```

DNS

1. DNS
2. UE DNS NAPTR
 - ECGI `eci-<hex>.ecgi.epc.mnc<MNC>.mcc<MCC>.epc.3gppnetwork.org`
 - TAI `tac-lb<hex>.tac-hb<hex>.tac.epc.mnc<MNC>.mcc<MCC>.epc.3gppnetwork.org`
 - RAI/SAI/CGI 3GPP TS 23.003
- 3.
4. DNS
- 5.

DNS DNS

```
; NAPTR PLMN 313-380 TAC 100
tac-lb64.tac-hb00.tac.epc.mnc380.mcc313.epc.3gppnetwork.org IN
NAPTR 10 50 "a" "x-3gpp-upf:x-sxb" "" upf-edge-1.example.com.

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

000

- **MEC** **UPF**
- **UPF** **PGW-C** **UPF**
- **UPF**
- **UPF**

UPF ☐ ☐ ☐ ☐

□□□□□□□□PGW-C □□□□□□ UPF □□□□□□□□□□□□□□ UPF□

□ □ □ □ □ □

□□□□□□□□□□UPF □□□ □□□

1. **PFCP** 介面UPF 介面 PFCP 介面
2. 網路側IP地址與埠號碼 3 個
3. 網路側UPF 介面 GenServer 介面

□□□□□□□□□□ UPF □□□ □□□□

- PFCP `associated: false`
- 3 `...`
- UPF `...`

□ □ □ □

UPF `upf_selection`

- 网络拥塞
- 网络 PFCP 问题
- 5 网络拥塞
- missed_heartbeats_consecutive 网络拥塞
- 网络拥塞 UPF 网络拥塞

DNS **UPF**

- [illegible]

- 000000000000
- 000000000000

0000

00/00000000 weight: 0 000

1. 000000 UPF
2. 000 000000 > 000 000000 == 00
3. 000000000000 UPF00000000 UPF
4. 0000000 UPF 0000000000 UPF000000 10
5. 00000000 UPF0000000000

000000000000 > 000

1. 0000000 UPF
2. 0000 UPF 000000000000
3. 00000000 UPF0000000000

000

```
[debug] 0000 UPF 002/3 00 UPF01 000
[info] 0000 UPF 00000000 UPF01 00 UPF0000 0 00 10
[warning] 00000000 UPF000 3 00000000000000
```

00 **UPF** 00

000000

```
# 確認 UPF 状態
iex> PGW_C.PFCP_Node.is_peer_healthy?({10, 100, 1, 21})
true

# 詳細状態取得
iex> PGW_C.PFCP_Node.get_peer_health({10, 100, 1, 21})
%{
  associated: true,
  missed_heartbeats: 0,
  healthy: true,
  registered: true
}
```

Web UI

- 確認画面 `/upf_selection`
- 確認 UPF 状態
- 確認 10-UP 10-100 10-100 100
- 確認 100 > 0 100 == 0 DNS 確認
- 確認 UPF 状態

確認

1. `upf_selection` 確認 **UPF** 100 UPF 状態

#####

DNS##### UPF

1. ## **PFCP** ##### UPF #####
2. ## **PFCP** ##### UPF ## PFCP ##
3. ##### UPF #####
4. #####10 #####
5. ##### UPF #####

UPF

UPF

```
%{
  name: "Dynamic-UPF-<IP>",                                # ""Dynamic-UPF-10-100-
  1-21"
  remote_ip_address: <discovered_ip>,                      # ## DNS ##### IP
  remote_port: 8805,                                       # ## PFCP #####
  initiate_pfcip_association_setup: true,                 # PGW-C #####
  heartbeat_period_ms: 10_000                             # 10 #####
}
```

UPF ##### `upf_selection`

#####**DNS** ##### **UPF**

```
# DNS #####upf-edge-2.example.com -> 10.200.5.99
# ## UPF ##### upf_selection #####

# #####
# 1. ##### UPF 10.200.5.99
# 2. ##“UPF {10, 200, 5, 99} #####...”
# 3. ## 10.200.5.99:8805 ## PFCP #####
# 4. ## UPF #####
# 5. ## UPF #####
```

##

DNS UPF
 PGW-C UPF
 UPF
 UPF
 UPF

UPF PFCP

[error] UPF {10, 200, 5, 99} PFCP :timeout
 [error] UPF {10, 200, 5, 99} :timeout
 UPF upf_selection

PGW-C

UPF	upf_selection
DNS UPF	
/ UPF	
UPF	upf_selection
/ UPF	UPF

UPF

UPF

[info] PFCP UPF {10, 200, 5, 99}{10, 200, 5, 99}:8805
 [info] UPF Dynamic-UPF-10-200-5-99 PID #PID<0.1234.0>

PGW-C에 등록된 UPF의 수를 반환합니다.

```
# PGW-C에 등록된 UPF의 수를 반환합니다
PGW_C.PFCP_Node.registered_peer_count()

# PGW-C에 등록된 UPF의 정보를 반환합니다
PGW_C.PFCP_Node.get_peer({10, 200, 5, 99})
# 반환값: {ok, #PID<0.1234.0>} 또는 {error, ...}
```

UPF에 대한 정보를 반환합니다.

UPF에 대한 정보를 반환합니다. PFCP 프로토콜을 사용하여 UPF에 정보를 등록합니다.

```
# PGW-C에 등록된 UPF의 정보를 반환합니다
PGW_C.PFCP_Node.register_dynamic_peer({10, 200, 5, 99}, 9999)
```

UPF에 대한 정보를 반환합니다. DNS를 사용하여 UPF의 IP 주소를 찾습니다. 8805 포트에서 PFCP 메시지를 보냅니다.

UPF에 대한 정보를 반환합니다.

UPF에 대한 정보를 반환합니다. UPF에 정보를 등록합니다.

□□□□□□□□ (CDR) □□

PGW-C □□□□□

OmniPGW □ Omnitouch □□□□□□

11

1. ☐ ☐
2. CDR ☐ ☐ ☐ ☐
3. CDR ☐ ☐
4. CDR ☐ ☐
5. ☐ ☐ ☐ ☐
6. ☐ ☐
7. CDR ☐ ☐ ☐ ☐
8. ☐ ☐ ☐ ☐ ☐ ☐
9. ☐ ☐
10. ☐ ☐

11

CDR PGW-C CDR

SGW-C CDR EPC

□ □ □ □

- **CSV** - Comma Separated Values
- **JSON** - JavaScript Object Notation
- **XML** - eXtensible Markup Language

- 0000 - 0000000000000000
- 00 **3GPP**  - 00 3GPP TS 32.251 PS 00000 TS 32.298 CDR 000

00

00	00
0000	00000000 CDR
00	00000000
0000	0000000000
0000	0000000000
0000	0000000000

CDR 0000

0000000

<epoch_timestamp>

000

1726598022

0000000000 Unix 00000000000000

0000

00000

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

cdr_directory config/runtime.exs

CDR

```
# CDR
# HH:MM:SS (unix_timestamp)
# HH:MM:SS (unix_timestamp)
# <gateway_name>
#
epoch,imsi,event,charging_id,msisdn,ue_imei,timezone_raw,plmn,tac,eci
```

- CDR Unix
 - Unix
 - PGW-C pgw_name
 - CSV
-

CDR 表

表名

項目	データ型	長さ	説明
0	epoch	整数	UNIX 時刻
1	imsi	文字列	IMSI
2	event	整数	CDR イベント番号“default_bearer_start”
3	charging_id	整数	充電 ID
4	msisdn	文字列	MSISDN 番号
5	ue_imei	文字列	UE IMEI
6	timezone_raw	文字列	UE 時刻領域
7	plmn	文字列	PLMN
8	tac	文字列	TAC
9	eci	文字列	E-UTRAN セル ID
10	sgw_ip	文字列	SGW-C S5/S8 インターフェイス IP
11	ue_ip	文字列	UE IP アドレスIPv4 IPv6
12	pgw_ip	文字列	PGW-C S5/S8 インターフェイス IP
13	apn	文字列	APN
14	qci	整数	QoS 値

項目	フィールド名	型	説明
15	octets_in	uint64_t	受信バイト数
16	octets_out	uint64_t	送信バイト数

CDR 形式

形式

CDR 形式

項目	フィールド名	型	説明
1	<type>_bearer_start	uint64_t	開始時刻
2	<type>_bearer_update	uint64_t	更新時刻
3	<type>_bearer_end	uint64_t	終了時刻/秒

形式

- default - デフォルト PDN 形式
- dedicated - デフォルト PDN 形式

形式

- default_bearer_start - デフォルト
- default_bearer_update - デフォルト
- default_bearer_end - デフォルト
- dedicated_bearer_start - デフォルト
- dedicated_bearer_update - デフォルト
- dedicated_bearer_end - デフォルト

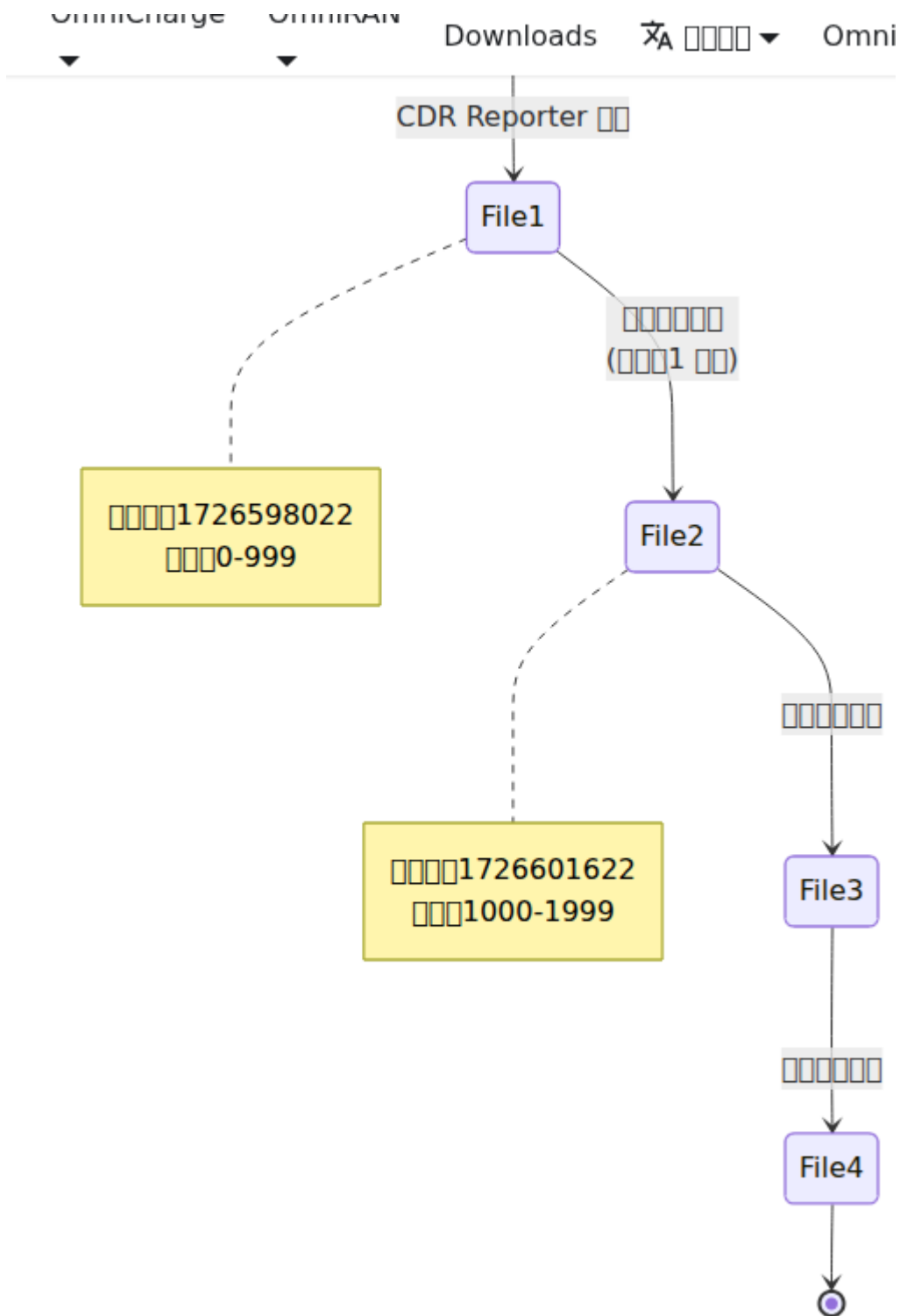
□□□□

□□ **CDR** □□

```
# □□ CDR □□□
# □□□□□□18:53:42 (1726598022)
# □□□□□□19:53:42 (1726601622)
# □□□□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
```

□□□□

CDR □□□□□□□□□□□□□□□□



□□□□□

1. □□□□ CDR □□
2. □□□□□□□□□□□□□□
3. □□□□□□□□
4. □□□ CDR □□□□□□□

PGW

PGW-C

PGW-C CDR `config/runtime.exs` `PGW-C`

項目	単位	説明	初期値	コメント
pgw_name	文字列	PGW 識別子 PGW 識別子 CDR 識別子	"omni- pgw01"	PGW識別子 ID
cdr_file_duration	整数	CDR ファイル 長さ	3600000	3600000以上 1 単位
cdr_directory	文字列	CDR ファイル 保存先	"/tmp/pgw_c"	/var/log/pgw_c/cdrs
usage_report_interval	整数	URR 報告 間隔 PGW- U 単位 報告 間隔	60000	60000以上 1 単位

PGW-C

PGW-C (`config/runtime.exs`):

```

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

  # URR PGW-U
  usage_report_interval: 60_000          # 60

```

CDR:

```

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

```

CDR:

```

config :pgw_c,
  pgw_name: "pgw-c-dev",
  cdr_file_duration: 300_000,            # 5
  cdr_directory: "/tmp/pgw_c_cdrs",
  usage_report_interval: 30_000          # 30

```

CDR:

```

config :pgw_c,
  pgw_name: "pgw-c-prod-heavy",
  cdr_file_duration: 1_800_000,          # 30
  cdr_directory: "/mnt/fast-storage/cdrs",
  usage_report_interval: 300_000         # 5

```

URR

PGW-C **PFCEP URRs** PGW-U URR PGW-U CDR

URR 配置

1. `usage_report_interval` PFCP 配置
2. PGW-C 配置 URR
3. PGW-U 配置
4. `bearer_update` CDR
5. `bearer_end` CDR

`usage_report_interval: 60_000`

- PGW-U 60
- 60 CDR
-

URR 定义

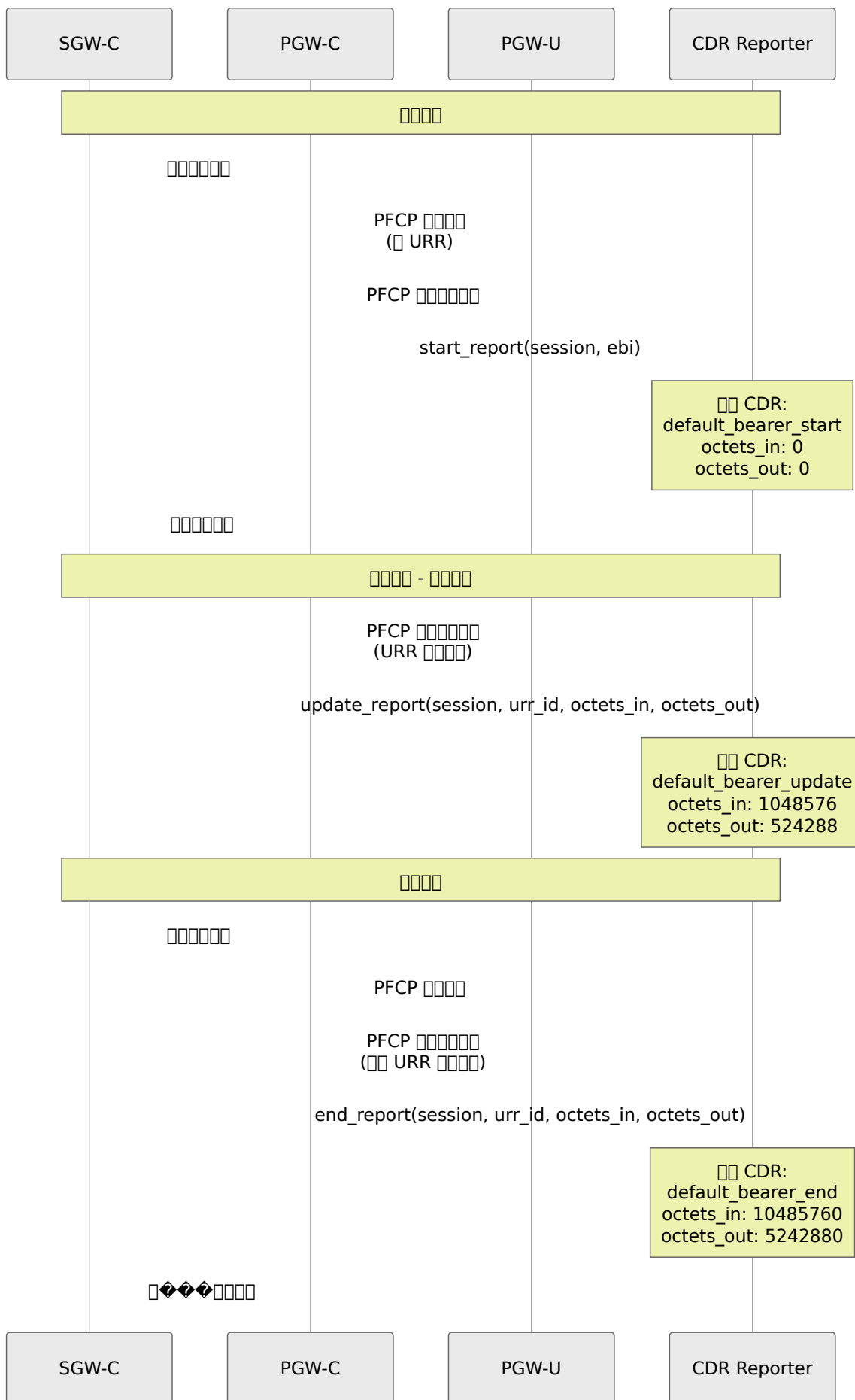
```
# lib/core/session/types.ex
defmodule PGW_C.Session.Types.URR do
  typedstruct do
    field :urr_id, non_neg_integer()
    field :measurement_method, :duration | nil
    field :reporting_triggers, :time_threshold | nil
    field :time_threshold, non_neg_integer() | nil #
  end
end
```

PFCP 配置 URR PFCP 配置 lib/core/session/impl/procedures.ex:468
配置 URR

CDR 配置

配置 CDR

PGW-C CDR



CDR 設定

1. 基本設定

- 名前 設定
- 単位 設定
- **octets_in** 0
- **octets_out** 0

2. 詳細設定

- 名前 PGW-U 単位 PFCP 単位 URR 単位
- 単位 設定
- **octets_in** 単位 設定
- **octets_out** 単位 設定
- 単位 URR 単位 `usage_report_interval` 単位

3. 追加設定

- 名前 PGW-U 単位 PFCP 単位
- 単位 設定
- **octets_in** 単位 設定
- **octets_out** 単位 設定

epoch 設定

1. epoch 設定

単位 Unix 単位

単位 CDR 単位

単位

2. imsi

15

MCCMNC + MSIN

310260123456789

MCC	MNC	MSIN	
(310)	(260)	(123456789)	

UE

3. eventCDR

<bearer_type>_bearer_<event>

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

- EBI EPS ID LBI ID default
- EBI LBI dedicated

EBI LBI

4. charging_id

32

12345

PGW-C

- SGW PGW
- Gy/Gz
-

5. msisdn

E.164

ISDN

+

15551234567



CC

(1) (5551234567)

UE MME HSS

6. ue_imei

15

TAC8+ SNR6+ Spare1

123456789012345



TAC

SNR S

UE MME

7. timezone_raw

/

UE

CSV

, ()

8. plmn

MCC: 505, MNC: 57

↓

"50557"

↓

"055570"

↓

0x055570 = 349552

349552 → MCC: 505, MNC: 57

MME UE

9. tac

16

UE

0 - 65535

1234

UE MME

-
-
- TAI

10. eciE-UTRAN

28

E-UTRAN UE

eNodeB ID20 + ID8

0 - 268,435,455

5678

MME UE

-
-
-

11. sgw_ipSGW IP

IPv4 IPv6

SGW-C S5/S8 IP F-TEID

IPv4IPv6

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

S5/S8

12. ue_ipUE IP

IPv4|IPv6

UE IP PDN

<ipv4>|<ipv6>

172.16.1.100| (IPv4)
|2001:db8::1 (IPv6)
172.16.1.100|2001:db8::1 ()

PGW-C PDN PAA

- IPv4IPv4
- IPv6IPv6
- PDN

13. pgw_ipPGW IP

IPv4 IPv6

PGW-C S5/S8 IP F-TEID

IPv4IPv6

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

PGW-C

14. apn

100

PDN

DNS

internet
ims
mms
enterprise.corporate

MME

-
-
- IP

15. qciQoS

8

QoS

1 - 9128-254

QCI

QCI					
1	GBR	2	100 ms	10^-2	
2	GBR	4	150 ms	10^-3	
3	GBR	3	50 ms	10^-3	
4	GBR	5	300 ms	10^-6	
5	Non-GBR	1	100 ms	10^-6	IMS
6	Non-GBR	6	300 ms	10^-6	
7	Non-GBR	7	100 ms	10^-3	
8	Non-GBR	8	300 ms	10^-6	
9	Non-GBR	9	300 ms	10^-6	

9 →

PGW-C QoS

16. octets_in

64

→ UE

1048576 → 1 MB

PGW-U PFCP URR

- update
- end
- start 0
- URR `usage_report_interval`

17. octets_out

64

UE →

524288 → 512 KB

PGW-U PFCP URR

CDR 000

1726570800,3102602222222222,default_bearer_start,10002,15552222222,222
1726571100,3102602222222222,default_bearer_update,10002,15552222222,22
1726571400,3102602222222222,default_bearer_update,10002,15552222222,22
1726571700,3102602222222222,default_bearer_update,10002,15552222222,22
1726572000,3102602222222222,default_bearer_end,10002,15552222222,22222

00 30000000000000

0000

1. 00000000QCI 9
2. 000000000000QCI 6
3. 0000000000
4. 000000
5. 000000

CDR 000

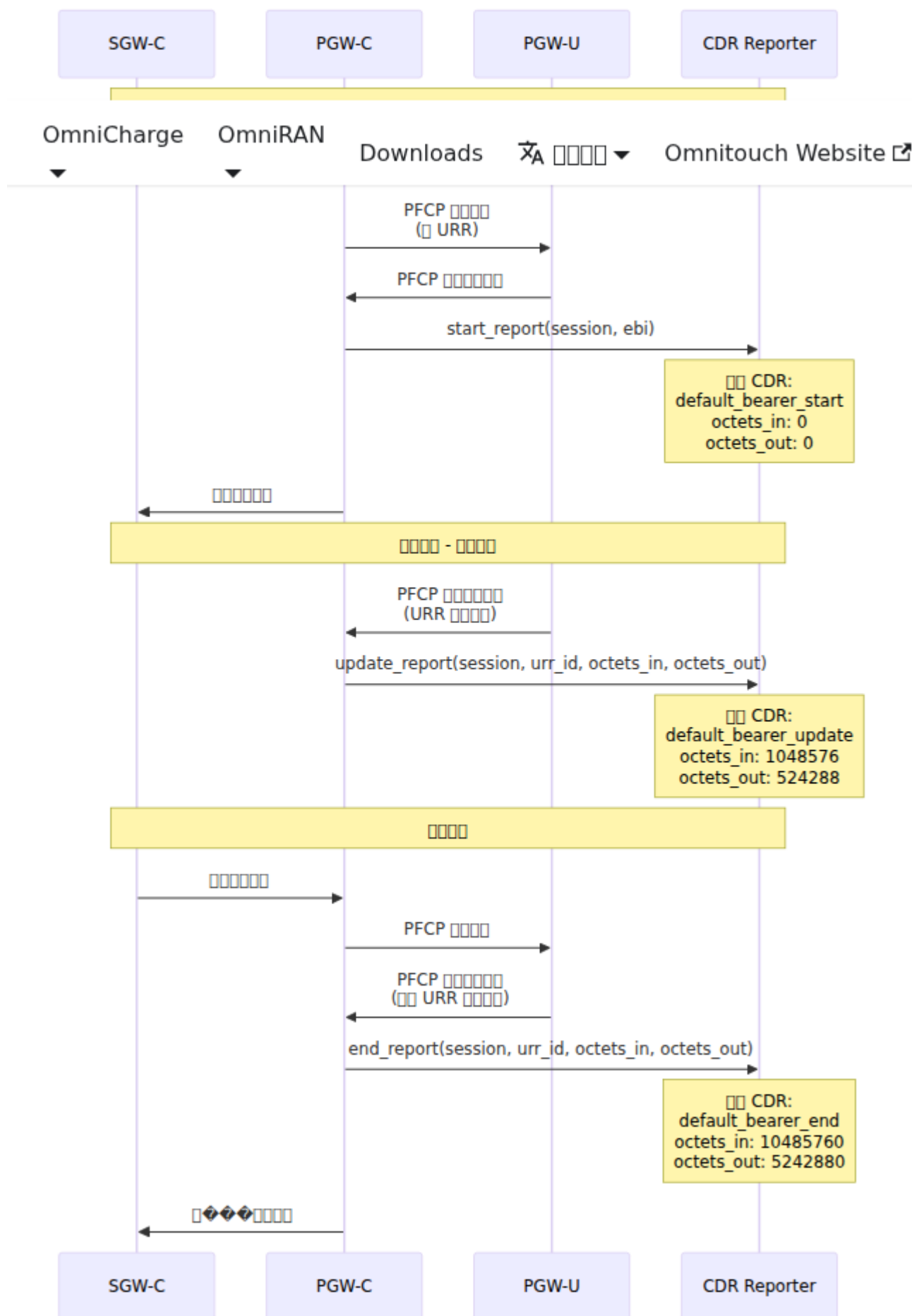
1726570800,3102603333333333,default_bearer_start,10003,15553333333,333
1726571100,3102603333333333,dedicated_bearer_start,10004,15553333333,3
1726571400,3102603333333333,default_bearer_update,10003,15553333333,33
1726571400,3102603333333333,dedicated_bearer_update,10004,15553333333,
1726571700,3102603333333333,dedicated_bearer_end,10004,15553333333,333
1726572000,3102603333333333,default_bearer_end,10003,15553333333,33333

000

- 0000001000300000000000 10 MB000 4 MB
- 0000001000400000000000 200 MB000 2 MB
- 000 QCI 009 60000000 QoS 00

□□

CDR □□□□



CDR 处理

1. 守护进程

```
# 守护 CDR 进程PGW-C
inotifywait -m /var/log/pgw_c/cdrs/ -e close_write | while read
path action file; do
    # 处理 CDR
    process_cdr "$path$file"
done
```

2. 脚本

```
# 实时监控 CDR
tail -F /var/log/pgw_c/cdrs/* | process_cdr_stream
```

CDR 类型

- **CDR** - 记录 CDR 信息
- **PFCP** - 记录 URRs 的 PGW-U 消息
- **CDR** - CDR 记录
- **CDR** - CDR 的 URR 记录
- **Gx** - CDR 的 QCI 记录
- **Gy** - 记录

3GPP

- TS 32.251 - 记录 PS 记录
- TS 29.274 - 3GPP 记录 EPS 的 GTP-C 记录
- TS 29.244 - CP 的 UP 记录 PFCP - **URR** 记录
- TS 32.298 - CDR 记录

CDR 00 - PGW-C 00000000

Omnitouch 000000

000000 1.0 000000 2025-12-28

Diameter Gx

Policy and Charging Rules Function (PCRF)

Overview

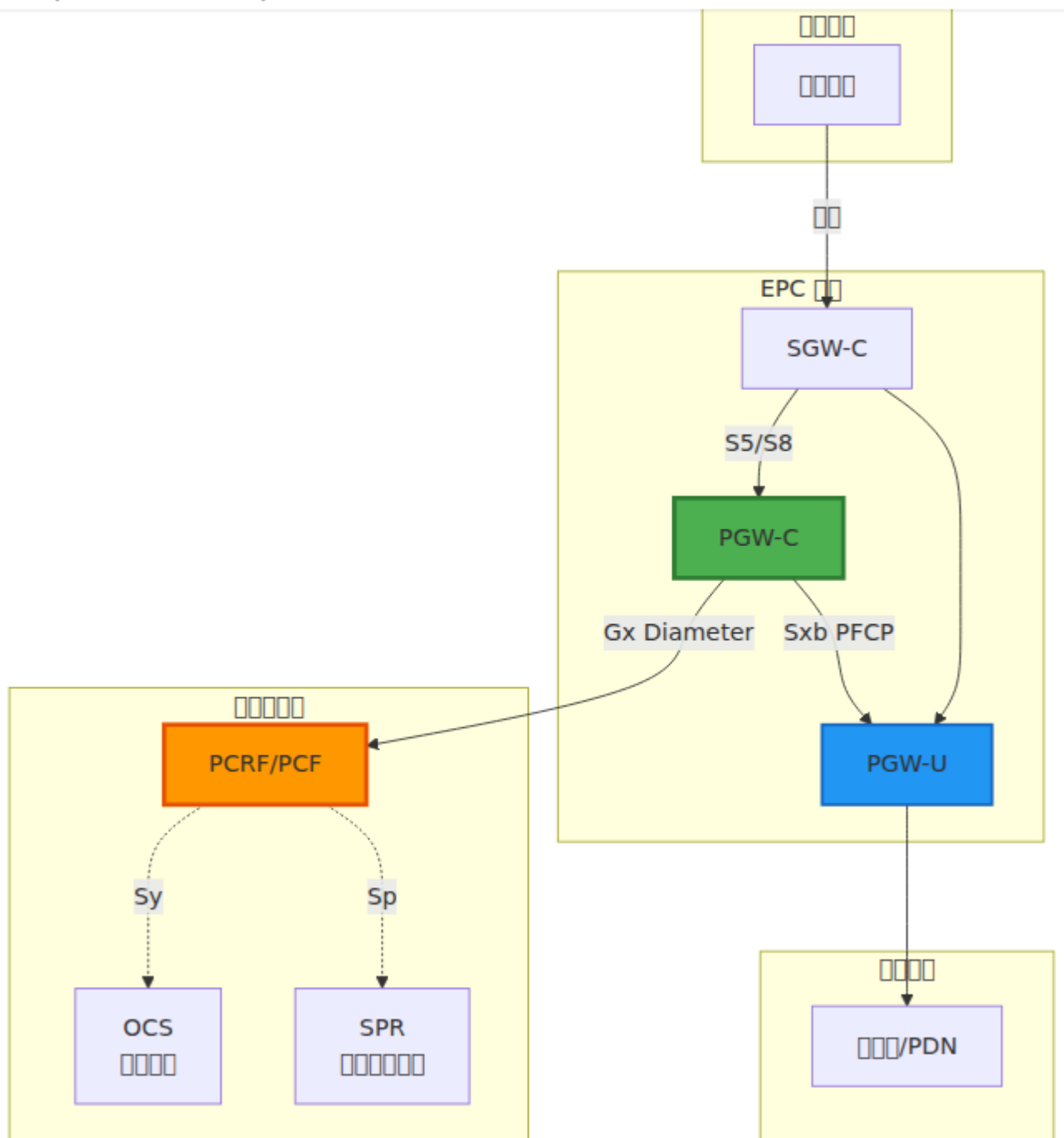
- Introduction
- Gx Interface
- Diameter Protocol
- Network Elements
- PCRF Functions
- QoS Management
- Charging Management
- Policy Management
- Session Management
- Security

Architecture

Gx interface connects PGW-C to PCRF. PCRF is responsible for policy and charging control in 5G networks.

- QoS Management - QoS flows
- Charging - Charging data collection
- Policy Management - Policy rules
- Session Management - Session establishment

网络架构 Gx



□□□□

□□	□□
□□□□	PCRF □□ PCC □□□□□□□□□□
QoS □□	□□□□□□□□ QoS □□
□□□□	□□□/□□□□□□□□□□
□□□□	□□□□□□/□□□□
□□□□	□□□□□□□□□□

Gx □□□□

3GPP □□

- □□□ 3GPP TS 29.212
- **Diameter** □□ **ID**□ 16777238 (Gx)
- □□□ Diameter □□□□ (RFC 6733)

□□□□

□□ UE PDN □□□□□□□□□ **Gx** □□□□ **Session-ID** □□□□□□□□

- □ UE □□□□□ (CCR-Initial)
- □□□□□□□□□□ (CCR-Update) - □□
- □ UE □□□□□ (CCR-Termination)

Session ID

Session-ID: <Origin-Host>;<high32>;<low32>[;<optional>]

Example: omni-pgw_c.epc.mnc999.mcc999.3gppnetwork.org;1234567890;98765

Fields

- **Origin-Host**: PGW-C or Diameter peer
- **high32**: high 32 bits
- **low32**: low 32 bits

Diameter

Fields

Diameter peer or PGW-C

```

Diameter Header (20 bytes)
├─ Version (1 byte) = 1
├─ Message Length (3 bytes)
├─ Flags (1 byte)
│   ├── R: Request (1) / Answer (0)
│   ├── P: Proxiable
│   ├── E: Error
│   └─ T: Potentially retransmitted
├─ Command Code (3 bytes)
├─ Application ID (4 bytes) = 16777238 (Gx)
├─ Hop-by-Hop ID (4 bytes)
└─ End-to-End ID (4 bytes)

```

```

AVPs (00-00)
├─ AVP Header
│   ├── AVP Code
│   ├── Flags (V, M, P)
│   ├── AVP Length
│   └─ Vendor ID (optional)
└─ AVP Data

```

00 Diameter 00

AVP0000-0000

- Diameter 00000000
- 00000000
- 00000000 AVP

000

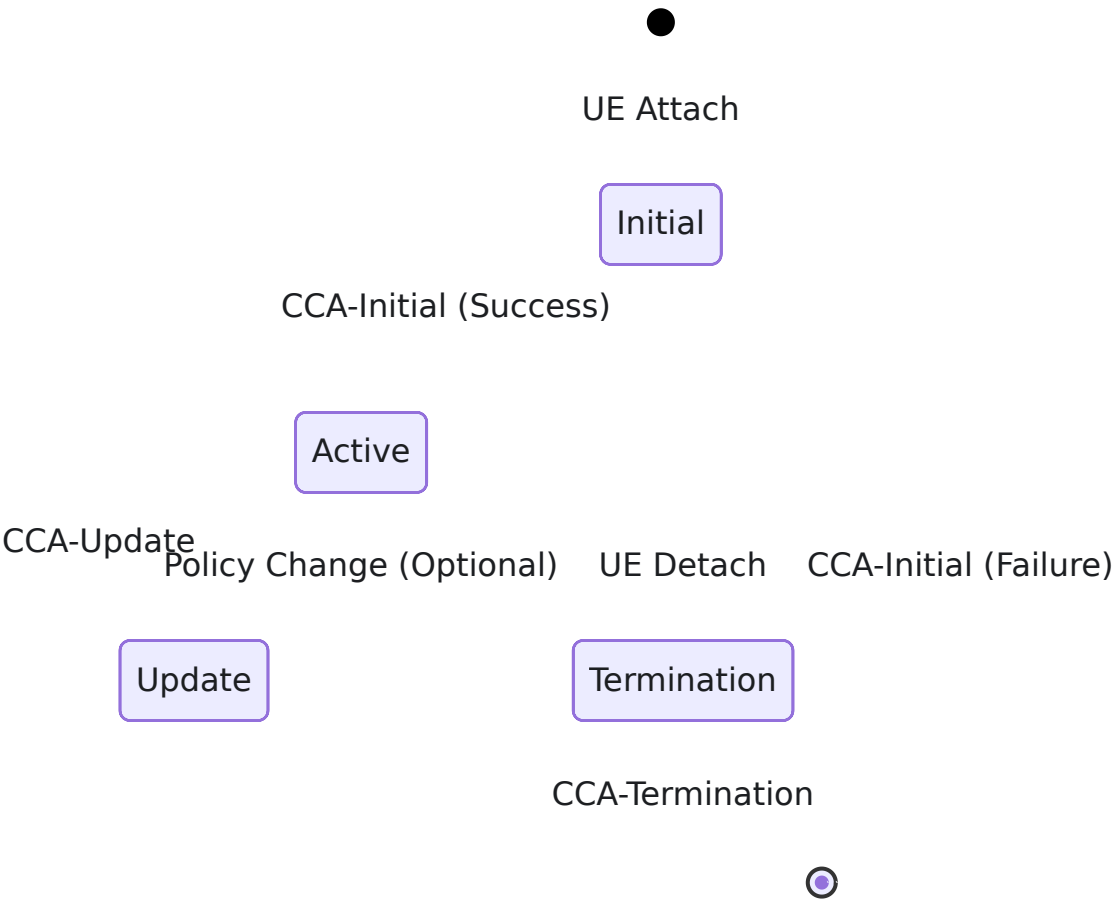
- 00/000
- CCR00000000/ CCA00000000

00000

- 2001 - DIAMETER_SUCCESS
- 3xxx - 0000
- 4xxx - 0000

- 5xxx -

PGW-C Diameter RFC 4006 Gx



CCR-Initial -

UE PDN

-
- PCRF UE

- QoS

PGW-C AVP

AVP	AVP		
Session-Id	263	UTF8String	Gx
Auth-Application-Id	258	Unsigned32	16777238 (Gx)
Origin-Host	264	DiamIdent	PGW-C Diameter
Origin-Realm	296	DiamIdent	PGW-C Diameter
Destination-Realm	283	DiamIdent	PCRF
CC-Request-Type	416	Enumerated	1 = INITIAL_REQUEST
CC-Request-Number	415	Unsigned32	0
Subscription-Id	443	Grouped	UE IMSI/MSISDN
Called-Station-Id	30	UTF8String	APN
Framed-IP-Address	8	OctetString	UE IPv4
IP-CAN-Type	1027	Enumerated	5 = 3GPP-EPS
RAT-Type	1032	Enumerated	1004 = EUTRAN
QoS-Information	1016	Grouped	QoS (AMBR)
Network-Request-Support	1024	Enumerated	
Supported-Features	628	Grouped	Gx

CCR-I

```
CCR (Command Code: 272, Request)
├─ 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 (Grouped)
│   └─ 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 (Grouped)
│   └─ APN-Aggregate-Max-Bitrate-UL: 100000000 (100 Mbps)
│       └─ APN-Aggregate-Max-Bitrate-DL: 50000000 (50 Mbps)
├─ Network-Request-Support: 1
└─ Supported-Features: [...]
```

CCA-Initial -

PCRF CCR-I

-
- PCC
- QoS

PGW-C AVP

AVP 名称	AVP 代码	描述
Result-Code	268	结果 (2001) 成功
Experimental-Result	297	实验性结果
QoS-Information	1016	QoS 信息
Charging-Rule-Install	1001	安装 PCC 规则
Charging-Rule-Definition	1003	规则定义
Default-EPS-Bearer-QoS	1049	默认 EPS 承载 QoS

命令示例

```
CCA (Command Code: 272, Answer)
├─ 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 (Grouped)
│   ├─ APN-Aggregate-Max-Bitrate-UL: 50000000 (50 Mbps - reduced)
│   └─ APN-Aggregate-Max-Bitrate-DL: 100000000 (100 Mbps -
increased)
├─ Charging-Rule-Install (Grouped)
│   ├─ Charging-Rule-Name: "default_internet_rule"
│   └─ Charging-Rule-Name: "video_streaming_rule"
└─ Charging-Rule-Definition (Grouped)
    ├─ Charging-Rule-Name: "default_internet_rule"
    ├─ QoS-Information: {...}
    └─ Precedence: 1000
```

CCR-Termination 消息 - 终止

UE 向 PDN 发起

PCRF

- PCRF
- /

CCR-I

- CC-Request-Type: TERMINATION_REQUEST (3)
-
- AVP

CCR-T

```
CCR (Command Code: 272, Request)
├─ 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

PCRF CCR-T

-
-

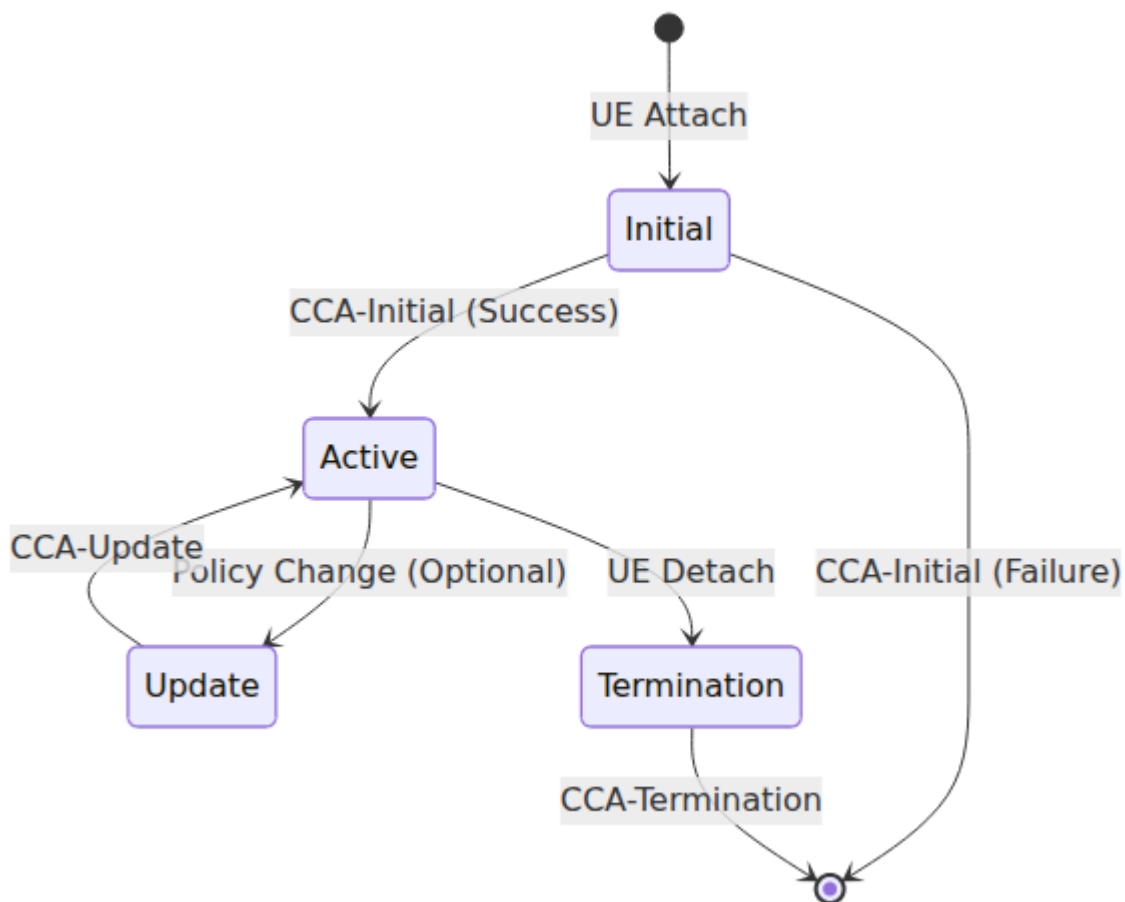
CCA-T

```
CCA (Command Code: 272, Answer)
├─ 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
```

□□□□□□□□

PCC □□□□

PCC□□□□□□□□□□ □□□□□□□□□□□□



□□□□

1. □□□□

- □□□□□□□□
- □□□ "video_streaming_rule"

2. □□□□

- □□□□ = □□□□□□
- □□□0-65535
- □□□□□□□□□□

3. □□□□TFT - □□□□□□□

- □□□□□□□□□□□□
- □□□
 - IP 5 □□□□□□□□/□□ IP□□/□□□□
 - "permit out ip from any to 8.8.8.8 80"

4. QoS □□□

- **QCI**□**QoS** □□□□□□ 1-9□□□□□□□128-254□□□□□□□□
 - QCI 1□□□□□□
 - QCI 5□IMS □□
 - QCI 9□□□□□□□
- **ARP**□□□□□□□□□□□□ □□□□
- **MBR/GBR**□ □□/□□□□□□

5. □□□□□

- □□□□□ □□□□□□□□□ OCS □□ - □ **Diameter Gy** □□□
- □□□□□□ □□□□□□□□□□□□
- □□/□□□□□□ OCS□□□□ **Diameter Gy** □□□□□□□□□□ CDR□□□□□ - □ □□ **CDR** □□□

6. □□□□□

- □□□□ □□□□□

- 3GPP 3GPP

3GPP 3GPP

PCRF 3GPP 3GPP

1. 3GPP 3GPP

Charging-Rule-Install (Grouped)

└─ Charging-Rule-Name: "gold_subscriber_internet"

└─ Charging-Rule-Name: "video_qos_boost"

2. 3GPP 3GPP

Charging-Rule-Definition (Grouped)

└─ Charging-Rule-Name: "dynamic_rule_123"

└─ Precedence: 100

└─ Flow-Information (Grouped)

| └─ Flow-Description: "permit out ip from any to 192.0.2.0/24"

| └─ Flow-Direction: DOWNLINK

└─ QoS-Information (Grouped)

| └─ QoS-Class-Identifier: 5

| └─ Max-Requested-Bandwidth-UL: 100000000

| └─ Max-Requested-Bandwidth-DL: 500000000

└─ Rating-Group: 1000

QoS 3GPP AVP

APN-AMBR 3GPP 3GPP

3GPP APN 3GPP GBR 3GPP

QoS-Information (Grouped)

└─ APN-Aggregate-Max-Bitrate-UL: 100000000 # 100 Mbps

└─ APN-Aggregate-Max-Bitrate-DL: 200000000 # 200 Mbps

PGW-C 3GPP

- AMBR
 - PGW-U QER
-

Gx

config/runtime.exs

```

config :pgw_c,
  diameter: %{
    # Diameter IP
    listen_ip: "0.0.0.0",

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

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

    # PCRF
    peer_list: [
      %{
        # PCRF Diameter
        host: "pcrf.epc.mnc999.mcc999.3gppnetwork.org",

        # PCRF PGW-C
        realm: "epc.mnc999.mcc999.3gppnetwork.org",

        # PCRF IP
        ip: "10.0.0.30",

        # PGW-C PCRF
        # true = PGW-C PCRF
        # false = PCRF
        initiate_connection: true
      }
    ]
  }
}

```

PCRF

```

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
      }
    ]
  }
}

```

□□□□□

- Diameter □□□□□□□□□□
- □□□□□□□□□□
- □□□□□□□□□□□□□□

□□□□□□

Diameter □□□□□ **FQDN**□□□□□□□□□□

```

# □□ - FQDN □□
host: "pgw_c.epc.mnc999.mcc999.3gppnetwork.org"

# □□□ - □□□□□ Diameter □□
host: "pgw_c"
host: "10.0.0.20" # □□□ IP □□

```

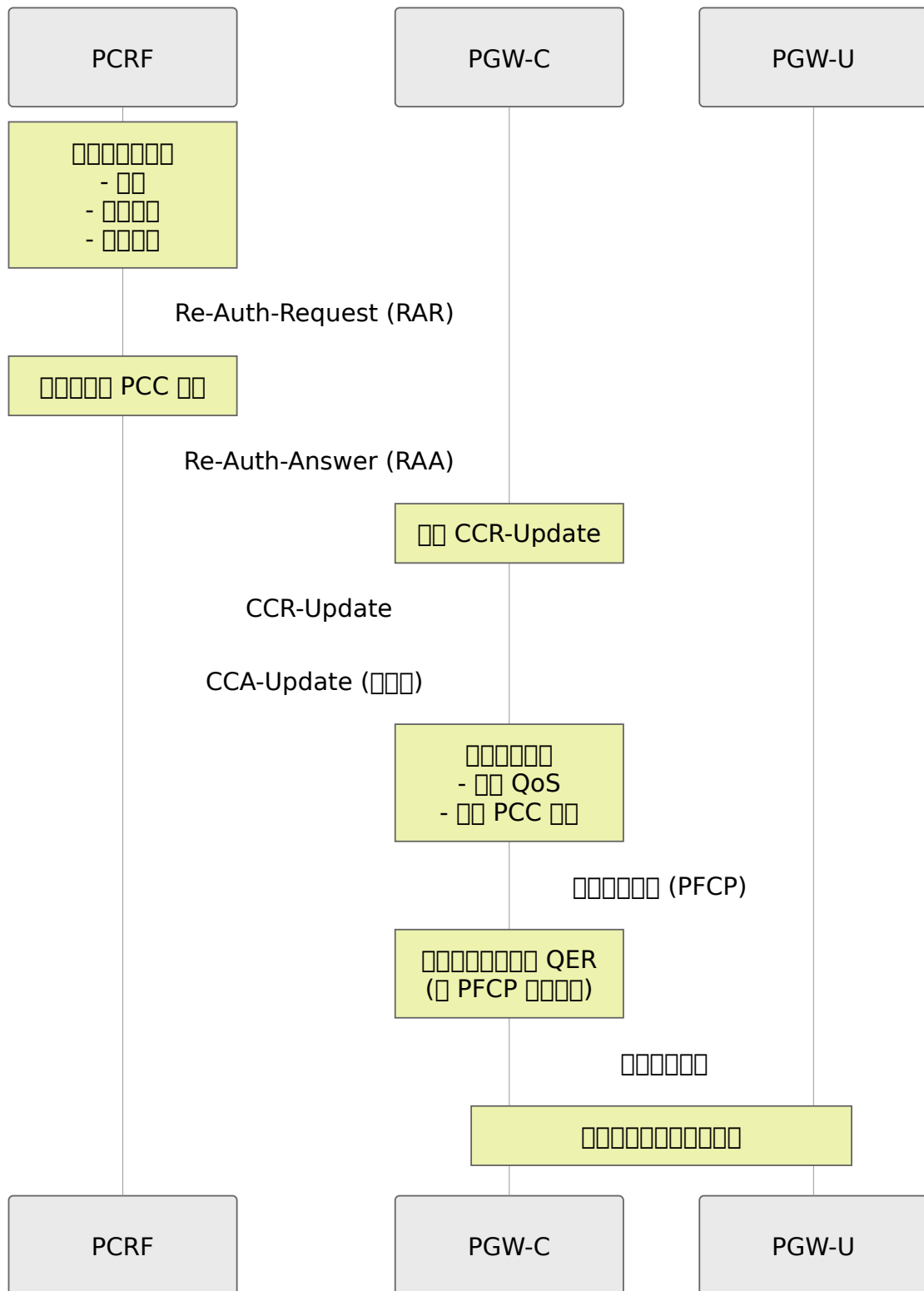
□□□□□

- `0000000000`
 - `3GPP PLMN 000000 epc.mncXXX.mccYYY.3gppnetwork.org`
-

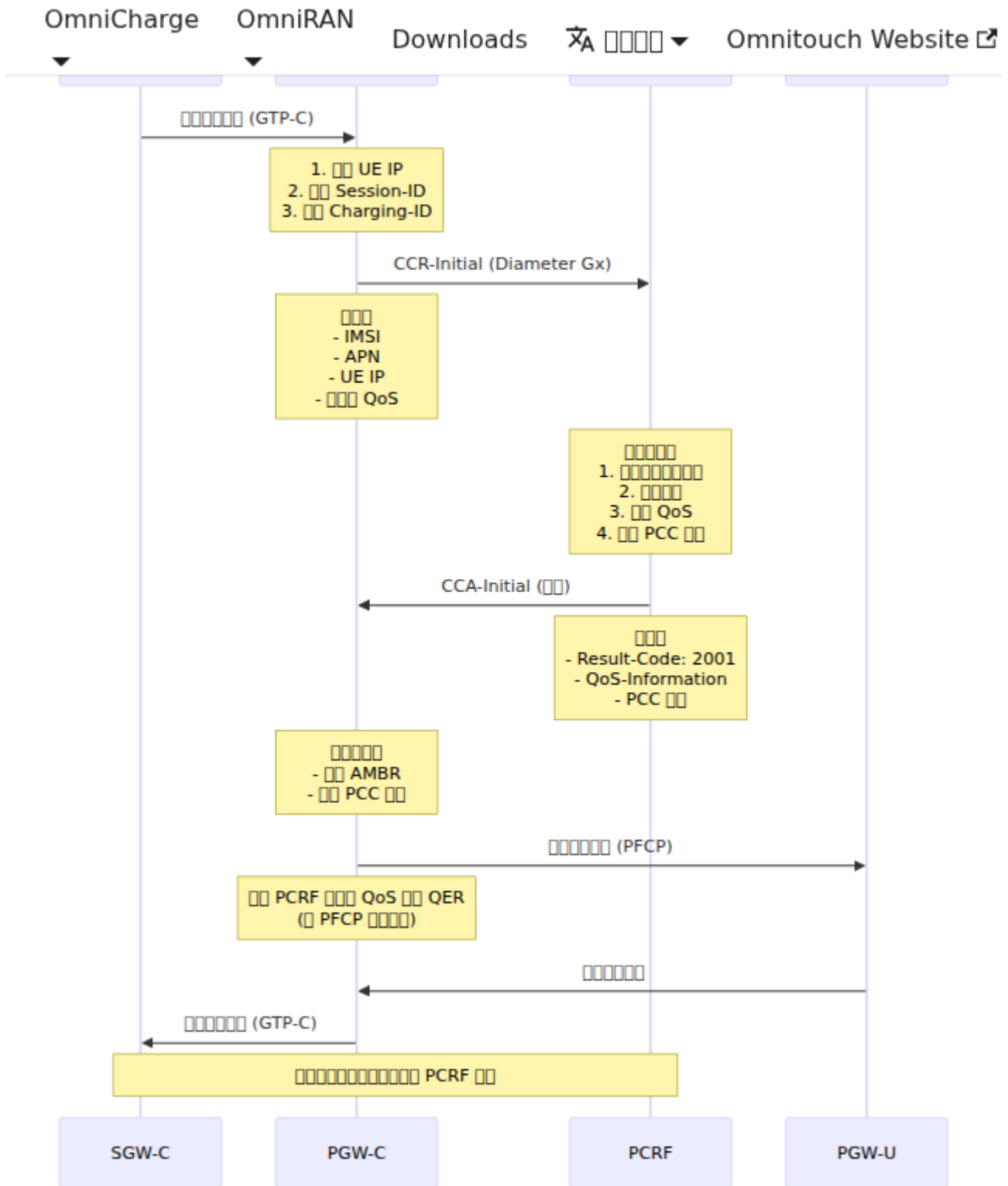
--	--	--	--	--	--	--



□□□□□□□□□□



□□□□



PGW-C

PGW-C

PGW-C CCA Diameter

PGW-C

PGW-C	PGW-C	PGW-C
2001	DIAMETER_SUCCESS	PGW-C

PGW-C (5xxx)

PGW-C	PGW-C	PGW-C
5002	DIAMETER_UNKNOWN_SESSION_ID	PGW-C
5030	DIAMETER_USER_UNKNOWN	PGW-C
5140	DIAMETER_ERROR_INITIAL_PARAMETERS	PGW-C
5003	DIAMETER_AUTHORIZATION_REJECTED	PGW-C

PGW-C (4xxx)

PGW-C	PGW-C	PGW-C
4001	DIAMETER_AUTHENTICATION_REJECTED	PGW-C
4010	DIAMETER_TOO_BUSY	PGW-C
4012	DIAMETER_UNABLE_TO_COMPLY	PGW-C

3GPP

Experimental-Result (Grouped)

Experimental-Result (Grouped)

└ Vendor-Id: 10415 (3GPP)

└ Experimental-Result-Code: <vendor-specific code>

3GPP

Code	Message	Meaning
5065	IP_CAN_SESSION_NOT_AVAILABLE	PCRF is not available
5143	INVALID_SERVICE_INFORMATION	Invalid service information

CCR-I

CCR-I

PCRF CCR-Initial

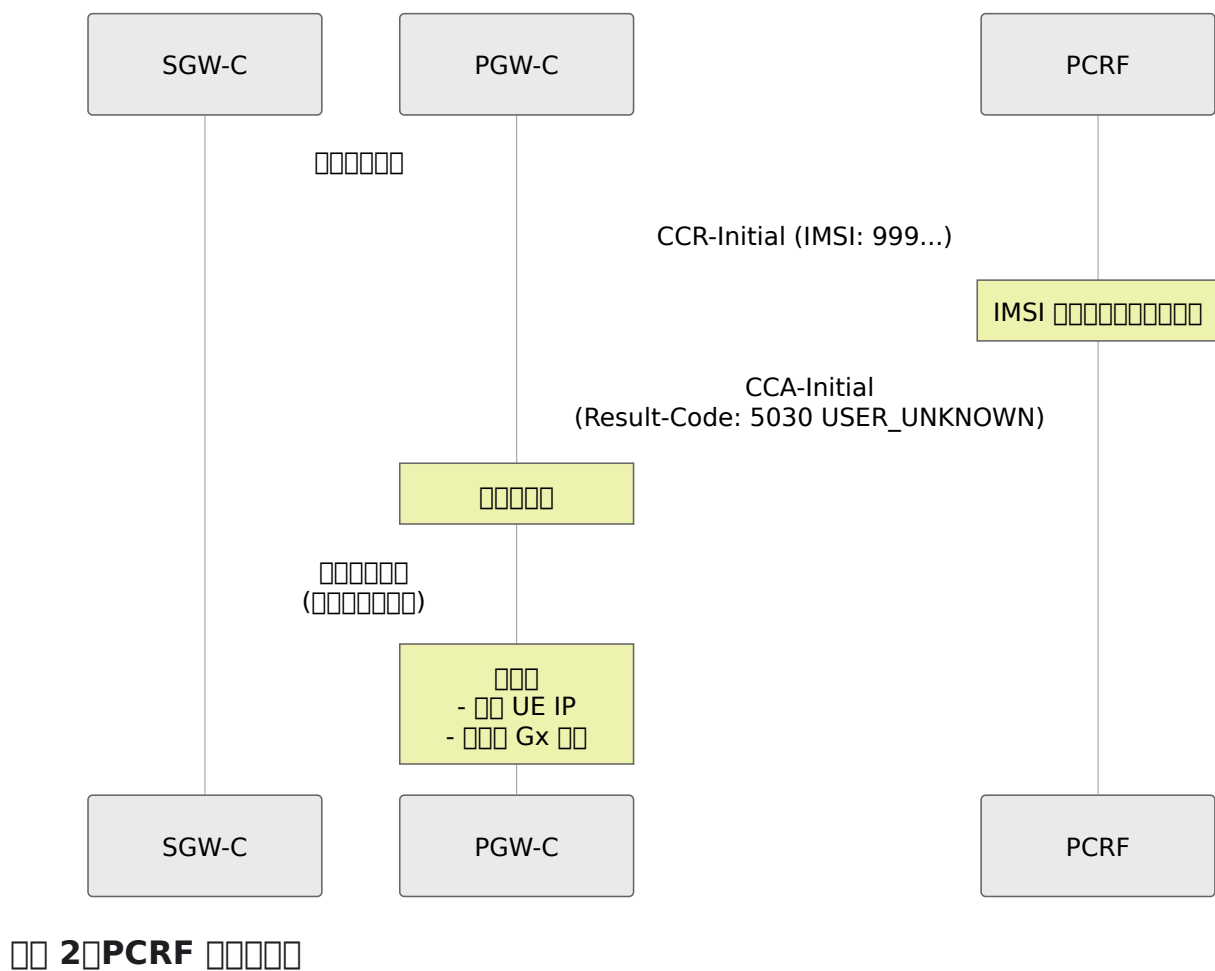
- PGW-C CCR-Initial
- CCA
 - “Session-ID CCR-Initial ...”
 - SGW-C
 -
- SGW-C

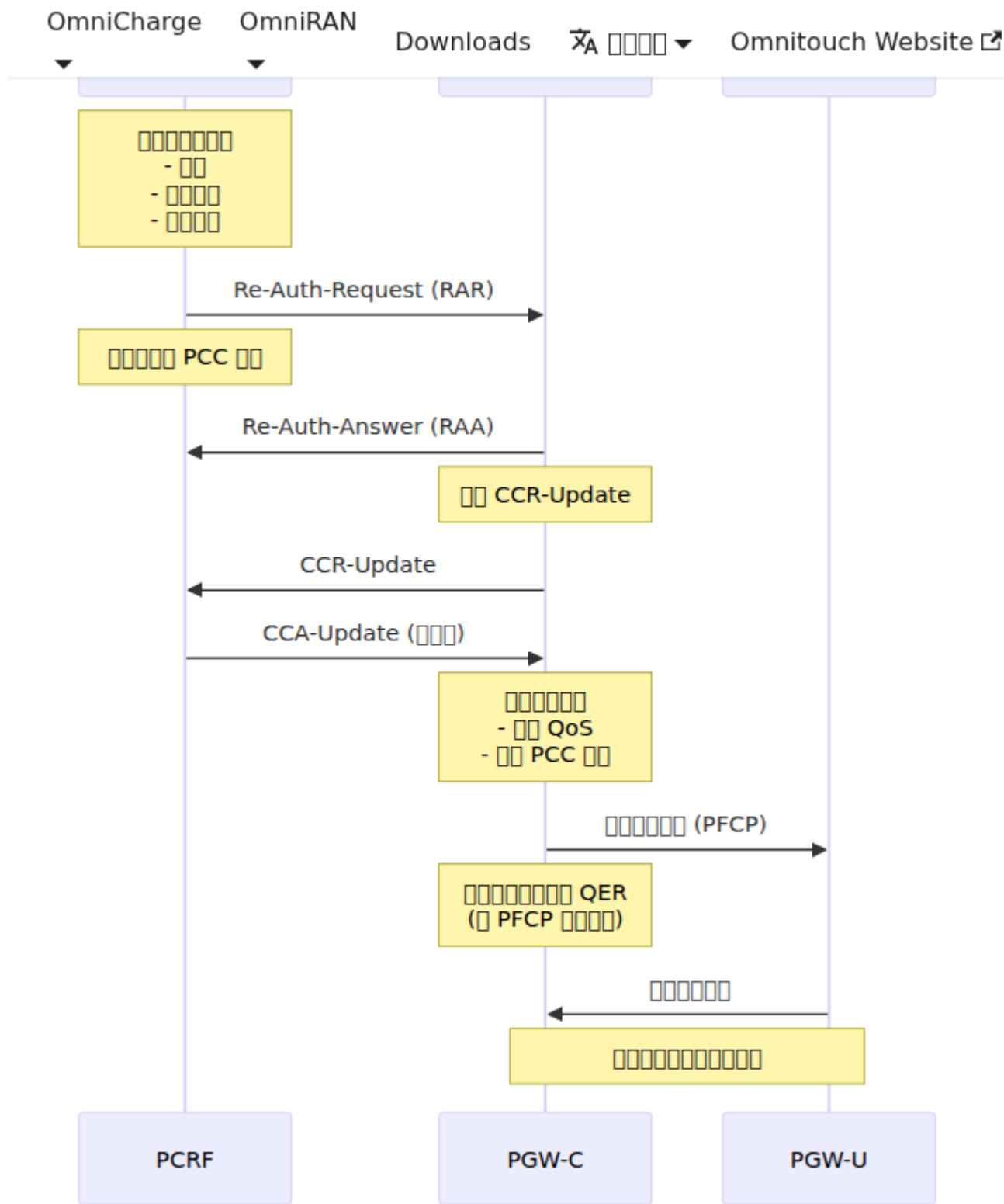
SGW-C

CCR-Initial PGW-C SGW-C
:remote_peer_not_responding

1

1 PCRF





□□□□

□□□□

1. Diameter □□□□□□

□□□

- □□□“Diameter □□□□□□”
- □□□ CCR-Initial

□□□□□

- PCRF □□□□
- □□□ PCRF IP □□□
- □□□□□ Diameter □□□3868□
- Diameter □□□□□/□□□□□□

□□□□□

```
# □□□□□□  
ping <pcrf_ip>  
  
# □□ Diameter □□□TCP 3868□  
telnet <pcrf_ip> 3868  
  
# □□ Diameter □□□□  
# □□□□□□□□ FQDN□□□□ IP
```

□□□□□

```

config :pgw_c,
  diameter: %{
    # 是否 FQDN 还是 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. CCR-Initial 消息

消息

- 消息内容
- 消息“CCR-Initial 消息”

消息内容

- PCRF 消息
- 消息
- PCRF 消息 Session-ID

消息内容

1. 消息 PCRF 消息内容
2. 消息 PCRF 消息内容
3. 消息内容 `ping <pcrf_ip>`
4. 消息内容

3. 消息 PCRF 消息

消息

- CCA-Initial 消息 Result-Code != 2001
- 消息内容

□□□□□□

□□□□	□□□□	□□□□
5030	IMSI □□□□□□□□	□ HSS/SPR □□□□□
5003	□□□□□	□□□□□□
4010	PCRF □□□□	□□□□□ PCRF □□

□□□□□

```
# PGW-C □□□□□  
[error] Diameter Gx □□□□□□□ 5030 (DIAMETER_USER_UNKNOWN)  
[error] IMSI 3102609999999999 □ PCRF □□
```

4. QoS □□□

□□□

- □□□□□□ QoS □□
- □□□□□□□□□□

□□□□□

1. □□ CCA-Initial□

- □□ QoS-Information AVP □□□□
- □□ APN-Aggregate-Max-Bitrate-UL/DL □

2. □□ PFCP □□□□□

- □□ QER □□□□□□□ MBR □□□
- □□ PGW-U □□□□□ QER □□

3. □□ PCRF □□□

- □□ PCRF □□

- 网络资源分配 QoS

5. Diameter 网络

网络

- Diameter 网络 PCRF
- 网络“网络资源分配”

网络

- 网络资源分配

网络

网络

```
# 网络
config :pgw_c,
  diameter: %{
    realm: "epc.mnc999.mcc999.3gppnetwork.org", # PGW-C 网络
    peer_list: [
      %{
        realm: "epc.mnc999.mcc999.3gppnetwork.org" # PCRF 网络
      }
    ]
  }
}
```

CCR-Initial 网络

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

网络 Gx 网络

网络

```

# Gx 消息
rate(gx_inbound_messages_total{message_type="gx_CCA"}[5m])
rate(gx_outbound_messages_total{message_type="gx_CCR"}[5m])

# Gx 错误
rate(gx_inbound_errors_total[5m])

# Gx 成功率
sum(rate(gx_outbound_responses_total{result_code_class="2xxx"}[5m])) /
sum(rate(gx_outbound_responses_total[5m])) * 100

# 每个 PCRF 的 Gx 成功率
rate(gx_outbound_responses_total{result_code_class!="2xxx"}[5m])
by (diameter_host)

# Gx 会话数
session_id_registry_count

# Gx 处理时长
histogram_quantile(0.95,
rate(gx_inbound_handling_duration_bucket[5m]))

```

数据源和聚合函数

`gx_outbound_responses_total` 聚合函数 `rate` 聚合函数 `sum` 聚合函数 `by` 聚合函数 `histogram_quantile` 聚合函数

- `message_type` 聚合函数 `gx_RAA` 聚合函数 `gx_CCA` 聚合函数
- `result_code_class` 聚合函数 `2xxx` 聚合函数 `3xxx` 聚合函数 `4xxx` 聚合函数 `5xxx` 聚合函数
- `diameter_host` 聚合函数 `PCRF` 聚合函数

聚合函数

- **2001** `DIAMETER_SUCCESS` - 成功
- **3001** `DIAMETER_COMMAND_UNSUPPORTED` - 不支持命令
- **5012** `DIAMETER_UNABLE_TO_COMPLY` - 无法遵守
- **5030** `DIAMETER_USER_UNKNOWN` - 用户未知

聚合函数

```

# Gx 错误率
- alert: GxErrorRateHigh
  expr: rate(gx_inbound_errors_total[5m]) > 0.1
  for: 5m
  annotations:
    summary: "Gx 错误率"

# Gx 响应失败率
- 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: "Gx 响应失败率"
    description: "10% Gx 响应失败率"

# PCRF 失败率
- 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 }} 失败率"
    description: "PCRF 失败率"

# 会话拒绝率
- alert: GxSessionRejection
  expr: rate(gx_inbound_errors_total{result_code="5030"}[5m]) >
0.01
  for: 5m
  annotations:
    summary: "PCRF USER_UNKNOWN"

```

错误率

响应失败率 Diameter 失败率

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

# 確認
iex> Logger.configure(level: :debug)
```

確認

- [debug] CCR-InitialSession-ID: ...
 - [debug] CCA-InitialResult-Code 2001
 - [error] Diameter ...
-

Web UI - Diameter 確認

OmniPGW 確認 Web UI 確認 Diameter 確認

Diameter 確認

確認 <http://<omnipgw-ip>:<web-port>/diameter>

PCRF Diameter Gx PCRF

1. PCRF

- PCRF - PCRF
- PCRF - PCRF
- 1 PCRF

2. PCRF

- - Diameter (Origin-Host)
- **IP** - PCRF IP
- - Diameter 3868
- - /
- - TCP SCTP
- - PGW PCRF
- - Diameter
- - PCRF
- **ID** - Diameter Gx = 16777238

3. PCRF

-
- CER/CEA
-
-

PCRF

1. 設定する Diameter 関
2. 設定する PCRF 設定する“名前”
3. 設定する名前
4. 設定する ID 関 Gx16777238

設定する名前**Gx** 設定

1. 設定する名前“PCRF 関”関
2. 関 Diameter 関
3. 設定する名前
 - 設定
 - 設定
 - 関 PCRF 設定
 - 関 TCP 3868 設定
 - 設定
 - 設定
 - PCRF 設定

関 **Diameter** 関

1. 設定する PCRF 関
2. 関 Diameter 関
3. 設定する名前
4. 設定する名前“**000**関”
5. 設定する名前
 - 設定
 - 関 ID 関 Gx
 - 設定する PCRF 関

設定

- 設定する PCRF 関
1. Diameter 設定する“名前”
 2. 設定する PCRF 関“名前”
 3. 設定する名前
 4. 設定する名前“名前”

関 **Diameter** 関

- 配置“名称”参数
- ID 为 Gx16777238
- 配置 PCRF 参数

配置参数

```
Web UI 参数
配置“名称”
参数
initiate_connection: true
```

- 配置
- OmniPGW 参数
 - PCRF 参数
 - 配置参数

配置

- 配置 - 1 参数
- 配置 - 参数/参数
- 配置 **Diameter** 参数 - 配置 Diameter CLI 参数
- 配置 - 配置参数
- 配置 - 配置 Diameter 参数
- 配置 - 配置 Diameter 参数

配置

配置 Web UI 配置 Prometheus 参数

- Gx 参数
- CCR/CCA 参数
- 配置

Web UI = “配置参数” 参数 = “配置参数”

PCRF

PCRF 功能

- **PCRF** - Diameter 协议 PCRF 功能
- **PFCP** 协议 - 用于 PCC 策略 QER 和 QoS 控制
- **PCRF** - 策略控制功能
- **QoS** 策略 - 用于 QoS 控制

PCRF 接口

- **Diameter Gy** 接口 - 用于 PCC 策略控制
- **CDR** 接口 - 用于计费数据记录
- **PCO** 接口 - IMS 网络 P-CSCF 接口

PCRF 部署

- **PCRF** - Gx 接口 PCRF 部署
- **S5/S8** 接口 - 用于策略控制

PCRF 部署

Diameter (Gy/Ro)

(OCS)

- 1.
2. 3GPP
3. Gy/Ro
- 4.
- 5.
- 6.
- 7.
- 8.
- 9.
- 10.
11. Gx
- 12.

Gy IMS Ro PGW-C (OCS)

- -
- -
- -
- -
- -

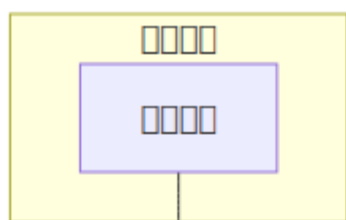
Table 1

Item	Parameter (Gy/Ro)	Parameter (Gz/Rf)
1	Parameter 1	Parameter 2
2	Parameter 3	Parameter 4
3	Parameter 5	Parameter 6
4	Parameter 7	Parameter 8
5	OCS (Parameter 9)	CGF/CDF (Parameter 10)
6	Parameter 11	Parameter 12
7	Parameter 13	Parameter 14
8	Parameter 15	Parameter 16

Table 2 CDR Table 1 Parameter 1

Table 3 Parameter 1 PDN Parameter 2

□□□□□ Gy



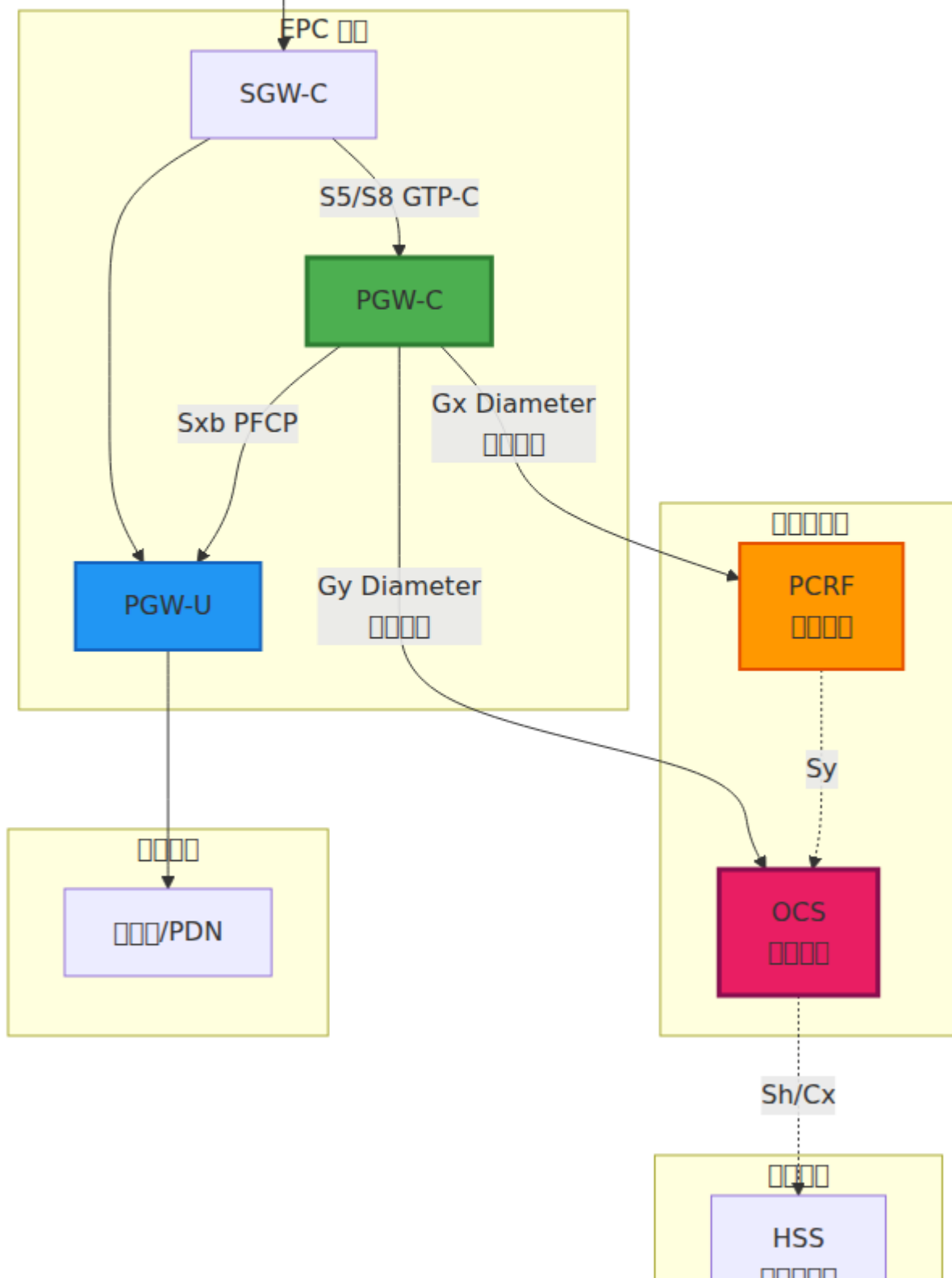
OmniCharge

OmniRAN

Downloads

🔍 ☐☐☐☐ ▼

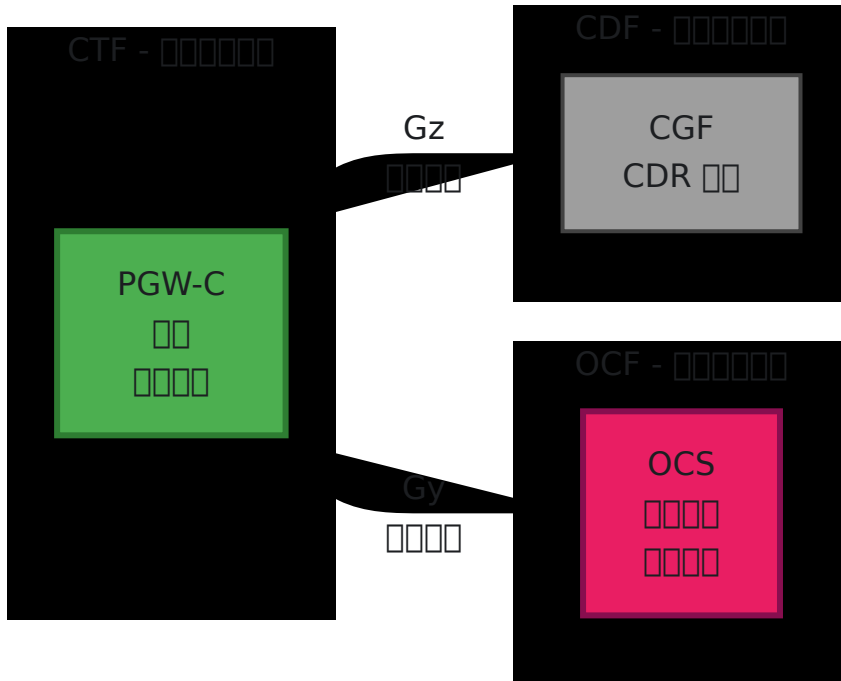
Omnitouch Websi



0000

00	00
0000	00000000 OCS 0000
0000	000000000000000000
000000	000000
0000	00000000000000
0000	000000000000
0000	0000000000000000

3GPP



(CTF)

PGW-C 는 CTF ()

1. -
2. -
3. -
4. -
5. -

(OCF)

OCS 는 OCF ()

1. -
2. - MB
3. -

4. Gy/Ro - 3GPP TS 32.299 (NAS)
5. Gy/Ro - 3GPP TS 32.251 (PS NAS)

Gy/Ro 消息

3GPP 消息

- 3GPP TS 32.299 (NAS)
- 3GPP TS 32.251 (PS NAS)
- Diameter** 消息 ID 4 (Gy/Ro - NAS)
- RFC 4006 (Diameter 消息)

消息

UE PDN 消息 **Session-ID** 消息 **Gy/Ro** 消息

- CCR-Initial
- CCR-Update
- CCR-Termination

消息 ID

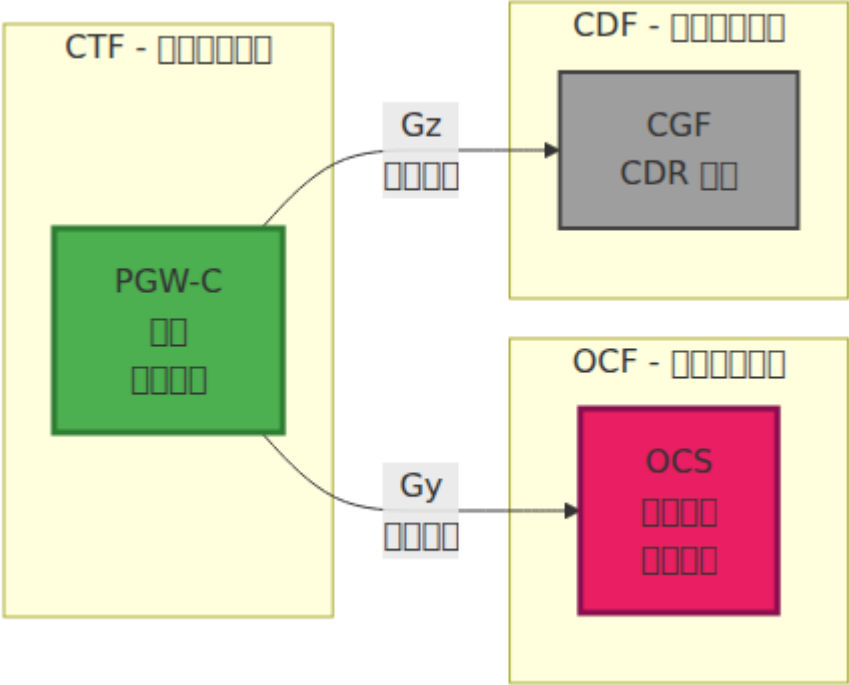
Session-ID: <Origin-Host>;<high32>;<low32>[;<optional>]
例子: omni-
pgw_c.epc.mnc999.mcc999.3gppnetwork.org;9876543210;12345;gy

消息

- Origin-Host:** PGW-C 消息 Diameter 消息
- high32:** 32 位
- low32:** 32 位
- optional:** 消息 "gy" 消息 Gx

□□□□□□

□□□□



CCR-Initial (□□□□□□ - □□)

□□□ UE □□ PDN □□□□□□□□□□□□

□□□

- □ OCS □□□□□□□□
- □□□□□□□□□□
- □□ Gy/Ro □□

PGW-C □□□□□ AVP□

AVP 이름	AVP 번호	타입	비고
Session-Id	263	UTF8String	0000 Gy 000000
Auth-Application-Id	258	Unsigned32	4 (0000)
Origin-Host	264	DiamIdent	PGW-C 0 Diameter 0
Origin-Realm	296	DiamIdent	PGW-C 0 Diameter 0
Destination-Realm	283	DiamIdent	OCS 000
CC-Request-Type	416	Enumerated	1 = INITIAL_REQUEST
CC-Request-Number	415	Unsigned32	000000 0 000
Subscription-Id	443	Grouped	UE 000 (IMSI/MSISDN)
Service-Context-Id	461	UTF8String	000000000
Multiple-Services-Credit-Control	456	Grouped	000000000
Requested-Service-Unit	437	Grouped	00000000000000
Used-Service-Unit	446	Grouped	0000000000 0
Service-Identifier	439	Unsigned32	0000000
Rating-Group	432	Unsigned32	0000000

00 CCR-I 000

```
CCR (Sequence: 272, Seq)
├─ 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 (Seq)
│   ├─ Subscription-Id-Type: END_USER_IMSI (1)
│   └─ Subscription-Id-Data: "310260123456789"
├─ Subscription-Id (Seq)
│   ├─ Subscription-Id-Type: END_USER_E164 (0)
│   └─ Subscription-Id-Data: "15551234567"
├─ Service-Context-Id: "32251@3gpp.org"
├─ Multiple-Services-Credit-Control (Seq)
│   ├─ Service-Identifier: 1
│   ├─ Rating-Group: 100
│   └─ Requested-Service-Unit (Seq)
│       └─ CC-Total-Octets: 100000000 (Seq 10 MB)
└─ Used-Service-Unit (Seq)
    └─ CC-Total-Octets: 0 (SeqSeq)
```

CCA-Initial (Sequence - Seq)

Sequence OCS Seq CCR-I Seq

Seq

- SequenceSequence
- SequenceSequence
- SequenceSequence

PGW-C Sequence **AVP**

AVP 名称	AVP 代码	说明
Result-Code	268	结果 (2001) 成功
Multiple-Services-Credit-Control	456	多业务信用控制
Granted-Service-Unit	431	授予的服务单元
Validity-Time	448	有效期
Result-Code	268	结果 (2001) 成功
Final-Unit-Indication	430	最终单元指示
Volume-Quota-Threshold	-	流量配额阈值

消息内容

```
CCA (消息 ID: 272, 类型)
├─ 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 (AVP)
│   ├─ Result-Code: DIAMETER_SUCCESS (2001)
│   ├─ Service-Identifier: 1
│   ├─ Rating-Group: 100
│   ├─ Granted-Service-Unit (AVP)
│   │   └─ CC-Total-Octets: 10000000 (约 10 MB)
│   ├─ Validity-Time: 3600 (有效期 1 小时)
│   └─ Volume-Quota-Threshold: 8000000 (约 8 MB 流量配额 80%)
```

CCR-Update (消息 ID - 类型)

消息

- 80%
-
-
-

-
-
-

CCR-I

- CC-Request-Type: UPDATE_REQUEST (2)
- CC-Request-Number
- Used-Service-Unit
- Requested-Service-Unit

CCR-U

```
CCR (272, )
├─ 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 ( )
    ├─ Service-Identifier: 1
    ├─ Rating-Group: 100
    ├─ Used-Service-Unit ( )
    │   └─ CC-Total-Octets: 8000000 (8 MB)
    └─ Requested-Service-Unit ( )
        └─ CC-Total-Octets: 10000000 (10 MB)
```

CCA-Update (信用更新 - 更新)

信用更新 OCS と CCR-U 関係

更新

- 信用更新の仕組み
- 更新の種類
- 更新の条件

更新の種類

1. 信用更新

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

2. 信用更新の仕組み

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

3. 信用更新

CCA ()

- └ 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 (-)

- UE
- PDN
-

-
- Gy/Ro
-

- CC-Request-Type: TERMINATION_REQUEST (3)
- Used-Service-Unit
- Requested-Service-Unit
- Termination-Cause

CCR-T

```
CCR (seq: 272, id)
├─ 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 (id)
    ├─ Service-Identifier: 1
    ├─ Rating-Group: 100
    └─ Used-Service-Unit (id)
        └─ CC-Total-Octets: 18500000 (id 18.5 MB)
```

CCA-Termination (seq - id)

seq OCS id CCR-T id

seq

- seq
- seq
- seq

seq CCA-T

```
CCA (seq: 272, id)
├─ 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
```

1. 背景

2. 需求

OCS 系统

名称	AVP	单位	说明
时间	CC-Time	秒	计费时间
流量	CC-Total-Octets	字节+KB	总流量
输入输出流量	CC-Input-Octets, CC-Output-Octets	字节	输入/输出流量
业务单元	CC-Service-Specific-Units	业务单元	业务单元API 接口
其他	-	其他	其他

3. 部署

PGW-C 系统

OCS 系统 **Volume-Quota-Threshold** **Time-Quota-Threshold** PGW-C 系统 PGW-U 系统 PFCP 系统 **PFCP** 系统

系统

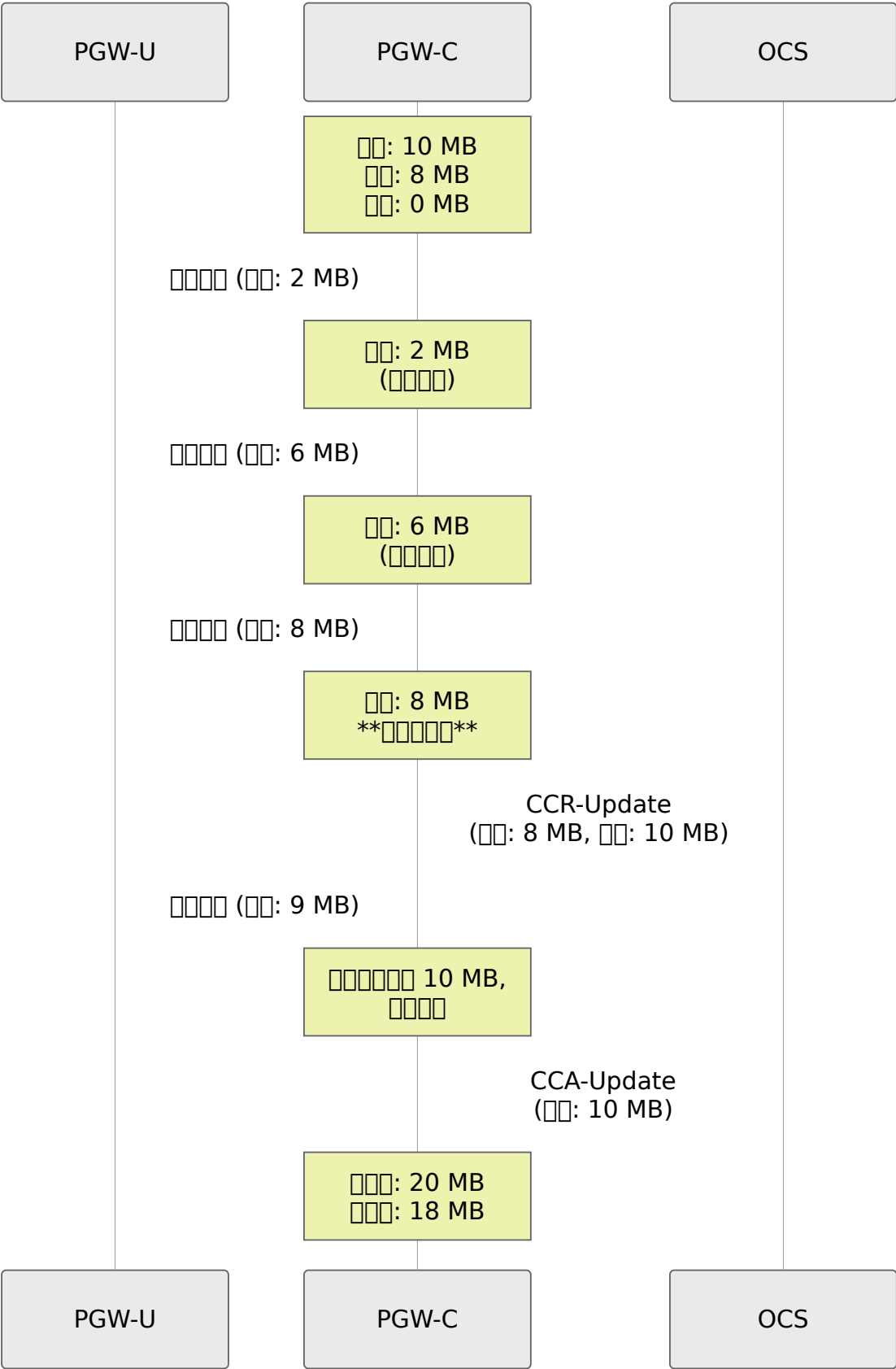
1. OCS 10 MB 80% 8 MB
2. PGW-C PGW-U PFCP
3. 8 MB
 - PGW-C CCR-Update
 -
4. OCS
5. CCR-Update
 - PGW-C

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

8 MB → CCR-Update
2 MB OCS

PGW-C

PGW-C PGW-U PFCP



PGW-U

PGW-C

OCS → CCA → **Final-Unit-Indication** AVP →

Final-Unit-Action		PGW-C →
TERMINATE	0	
REDIRECT	1	
RESTRICT_ACCESS	2	

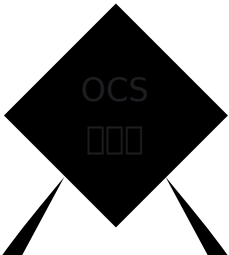
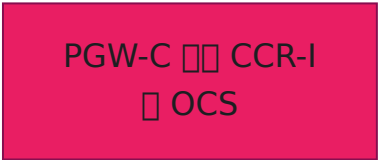
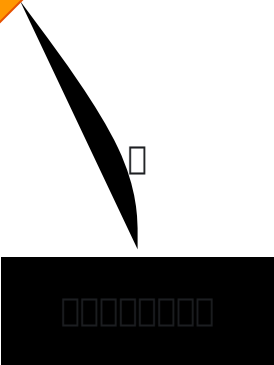
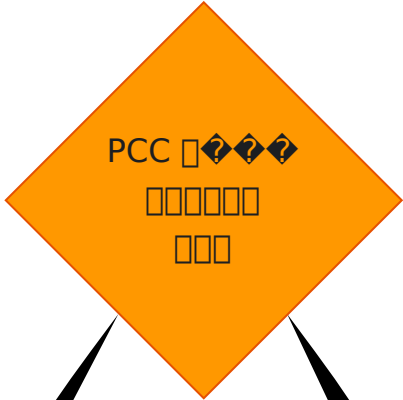
```
CCA (→)
└─ Multiple-Services-Credit-Control
    └─ Result-Code: DIAMETER_SUCCESS (2001)
    └─ Granted-Service-Unit
        └─ CC-Total-Octets: 1000000 (→ 1 MB)
    └─ Final-Unit-Indication
        └─ Final-Unit-Action: REDIRECT (1)
            └─ Redirect-Server (→)
                └─ Redirect-Address-Type: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

PGW-C →

- 1. **TERMINATE:** → CCR-T→
- 2. **REDIRECT:** → PCFP → HTTP → URL
- 3. **RESTRICT_ACCESS:** → PCFP → IP

3GPP TS 23.002: PCRF, Gx, PCC, Diameter Gx, PCC

3GPP TS 23.002





PCRf (Gx CCA-I)

PCC □□□□□□□□ **AVP**□

AVP 名称	AVP 值	数据类型	说明
Rating-Group	432	Unsigned32	计费策略组 OCS 计费
Online	1009	0=否, 1=是	在线计费 (Gy)
Offline	1008	0=否, 1=是	离线计费 (Gz)
Metering-Method	1007	0=预付费, 1=后付费, 2=混合	计费方式
Reporting-Level	1011	0=否, 1=是	计费报告

计费策略组

计费策略组	计费策略	Rating-Group	说明
1	0	计费策略	预付费策略
0	1	计费策略	后付费策略
1	1	计费策略	混合策略
0	0	-	无策略

计费策略

计费 PDN 列表 计费策略组

计费策略

PGW-C Gy

└─ Rating-Group: 100 ()
└─ Online: 1

PGW-C Gy

└─ Rating-Group: 200 ()
└─ Online: 1

PGW-C Gy

└─ Rating-Group: 300 ()
└─ Online: 1

PGW-C Gy

- CCR-I MSCC

CCR-Initial

└─ Session-Id: "..."
└─ Multiple-Services-Credit-Control
└─ [Rating-Group: 100] →
└─ [Rating-Group: 200] →
└─ [Rating-Group: 300] →

OCS

CCA-Initial

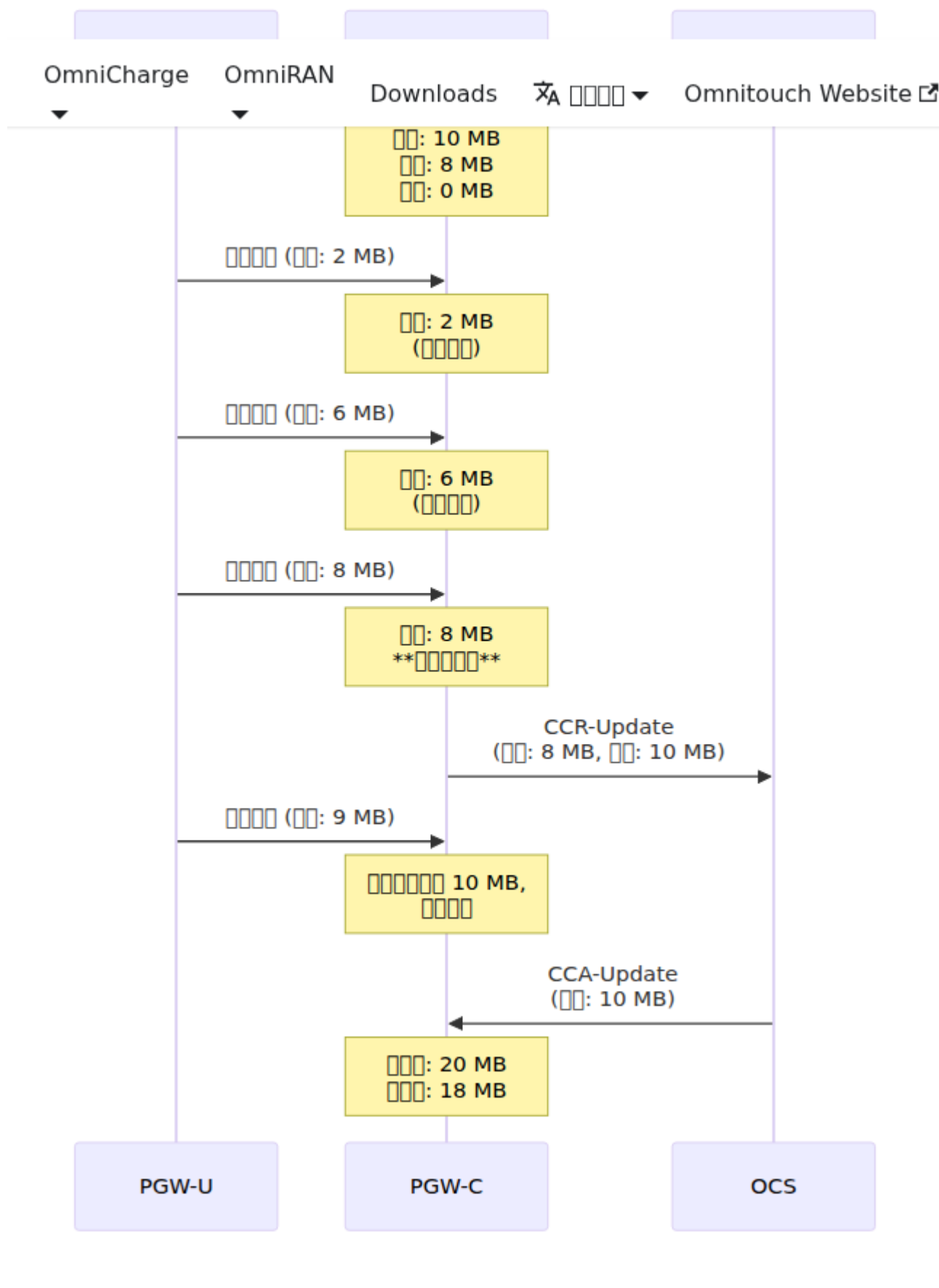
└─ Multiple-Services-Credit-Control
└─ [Rating-Group: 100] → : 10 MB
└─ [Rating-Group: 200] → : 5 MB ()
└─ [Rating-Group: 300] → : 60

PGW-C

PGW-C Rating-Group

```
# [][]
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} # []
}
```

[][][][][][][]



□□□□□□□□

MSCC (□□□□□□□) AVP

□□□ □□□□□/□□□□□□□□□□

□□□

```
Multiple-Services-Credit-Control (□□, AVP 456)
├─ Service-Identifier (Unsigned32, AVP 439)
├─ Rating-Group (Unsigned32, AVP 432)
├─ Requested-Service-Unit (□□, AVP 437)
│   └─ CC-Time (Unsigned32, AVP 420)
│   └─ CC-Total-Octets (Unsigned64, AVP 421)
│   └─ CC-Input-Octets (Unsigned64, AVP 412)
│   └─ CC-Output-Octets (Unsigned64, AVP 414)
├─ Used-Service-Unit (□□, AVP 446)
│   └─ [□ Requested-Service-Unit □□□□]
├─ Granted-Service-Unit (□□, AVP 431)
│   └─ [□ Requested-Service-Unit □□□□]
├─ Validity-Time (Unsigned32, AVP 448)
├─ Result-Code (Unsigned32, AVP 268)
└─ Final-Unit-Indication (□□, AVP 430)
    └─ Final-Unit-Action (Enumerated, AVP 449)
```

Table 1: Example of a Rating Group

Rating Group	Rating Group	Rating Group
Rating Group	Rating Group	Rating Group
Rating Group	Rating Group	Rating Group
Rating Group	Rating Group	Rating Group
Rating Group	Rating Group	Rating Group
Rating Group	Rating Group	Rating Group
Rating Group	Rating Group	Rating Group

Rating Group

Service-Identifier: 1 (Rating Group)

- Rating-Group: 100 (Rating Group - \$0.01/MB)
- Rating-Group: 200 (Rating Group - \$0.05/MB)

Service-Identifier: 2 (Rating Group)

- Rating-Group: 300 (Rating Group - \$0.10/MB)

Rating Group

Rating Group

Rating Group config/runtime.exs

```

config :pgw_c,
  online_charging: %{
    # 在线计费功能
    enabled: true,

    # OCS 鉴权
    timeout_ms: 5000,

    # 默认请求的 PCRF 配额
    default_requested_quota: 10_000_000, # 10 MB

    # 配额阈值
    # (0.8 = 80% 配额 CCR-Update)
    quota_threshold_percentage: 0.8,

    # OCS 鉴权
    # :block, :allow
    timeout_action: :block,

    # OCS 鉴权
    # :terminate, :redirect
    no_credit_action: :terminate,

    # 鉴权 URL no_credit_action: :redirect 鉴权
    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",

    # OCS 鉴权
    peer_list: [
      # PCRF 鉴权 (Gx)
      %{
        host: "pcrf.epc.mnc999.mcc999.3gppnetwork.org",
        realm: "epc.mnc999.mcc999.3gppnetwork.org",
        ip: "10.0.0.30",
        initiate_connection: true
      },
      # OCS 鉴权 (Gy)
      %{
        host: "ocs.epc.mnc999.mcc999.3gppnetwork.org",

```

```

        realm: "epc.mnc999.mcc999.3gppnetwork.org",
        ip: "10.0.0.40",
        initiate_connection: true
    }
]
}

```


enabled

- **true**: CCR OCS
- **false**: Gy

timeout_ms

- OCS CCA
- 3000-5000

default_requested_quota

- PCRF
- 1-100 MB

quota_threshold_percentage

- % CCR-Update
- 0.75-0.85 (75%-85%)
- =
- =

timeout_action

- **:block** - OCS
- **:allow** - OCS

no_credit_action

- **:terminate** -

- **:redirect** - 重定向

配置

配置

```
config :pgw_c,  
  online_charging: %{  
    enabled: true,  
    timeout_action: :block,  
    no_credit_action: :terminate,  
    quota_threshold_percentage: 0.8  
  }
```

配置

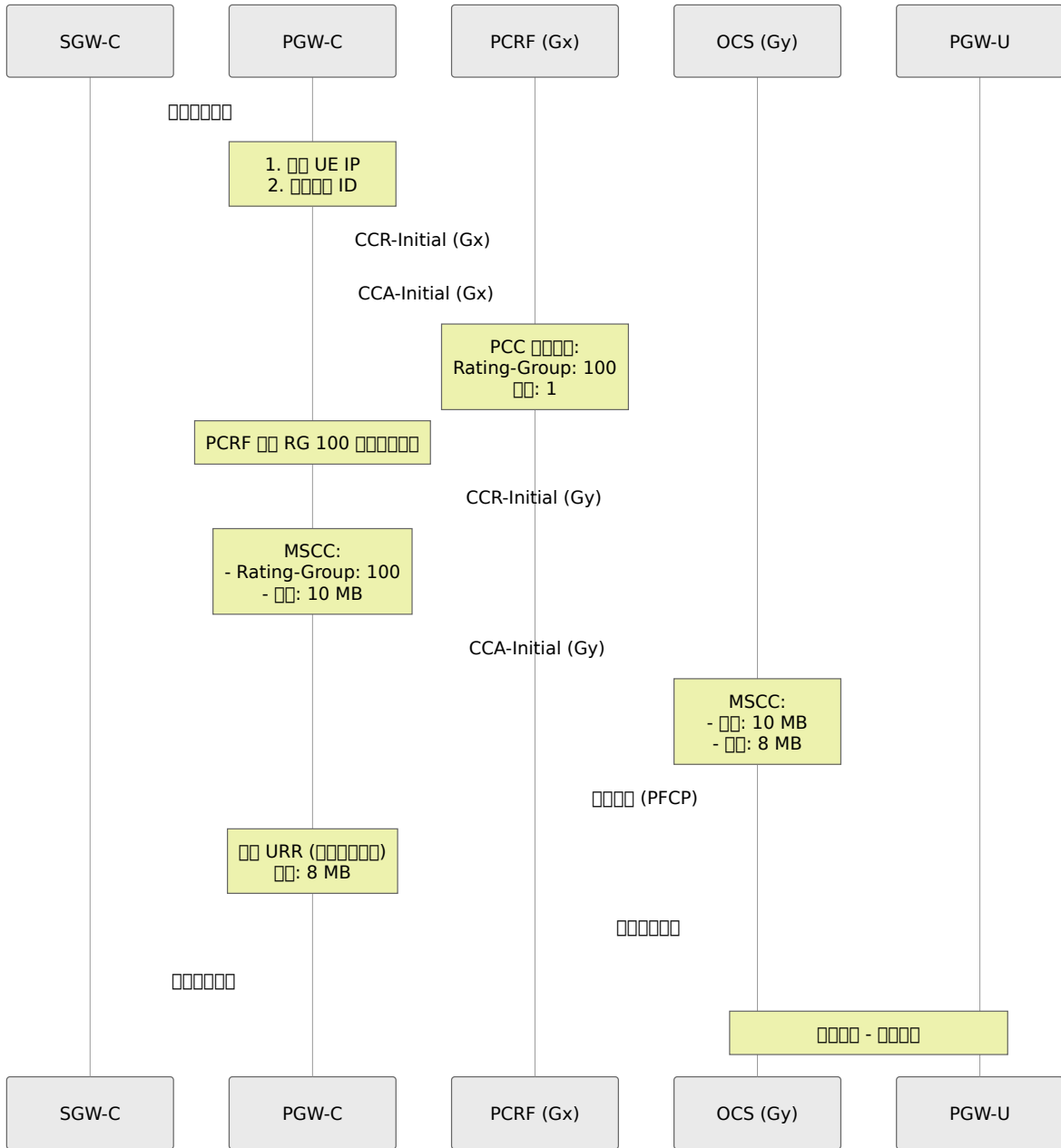
```
config :pgw_c,  
  online_charging: %{  
    enabled: false # 禁用  
  }
```

配置

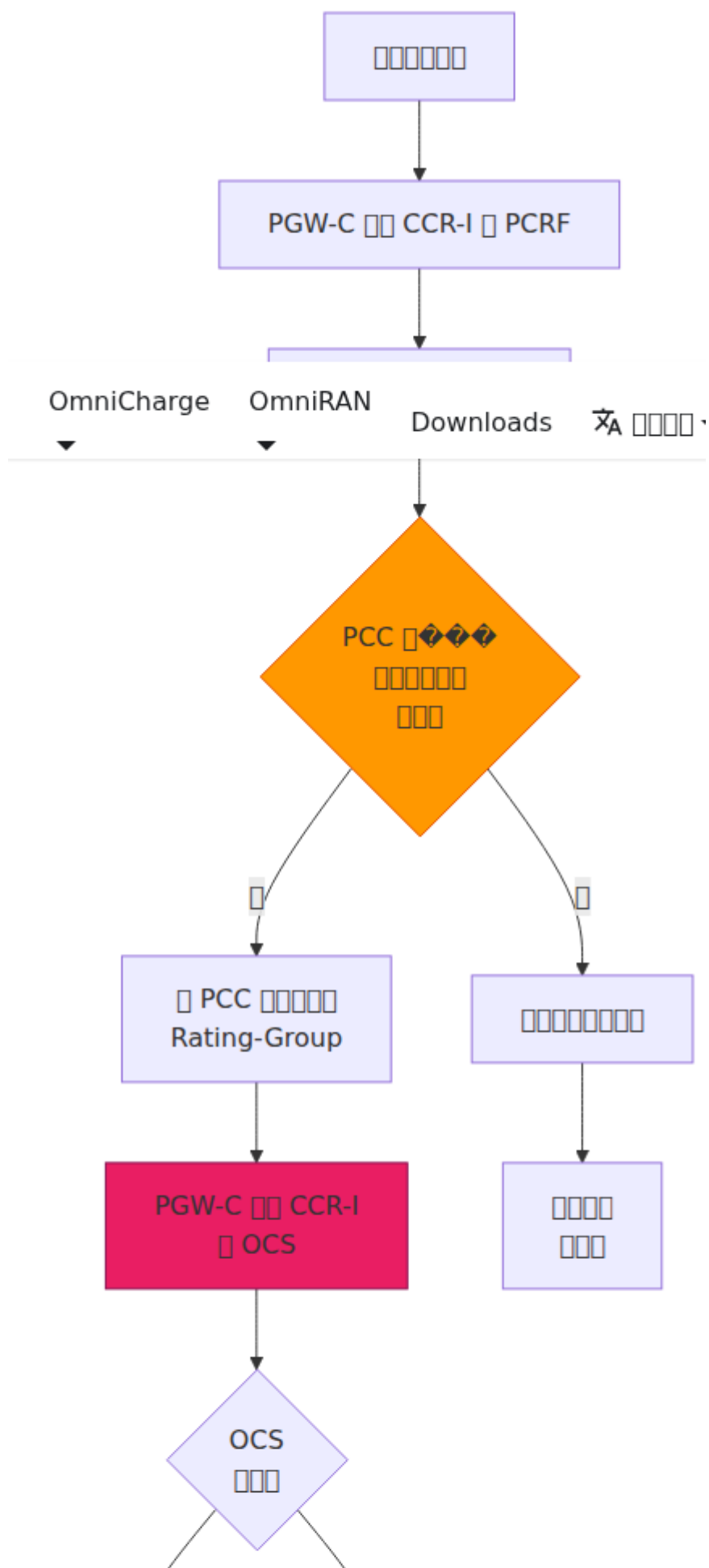
```
config :pgw_c,  
  online_charging: %{  
    enabled: true, # 启用 PCRF 计费  
    timeout_action: :allow, # OCS 计费  
    no_credit_action: :terminate  
  }
```

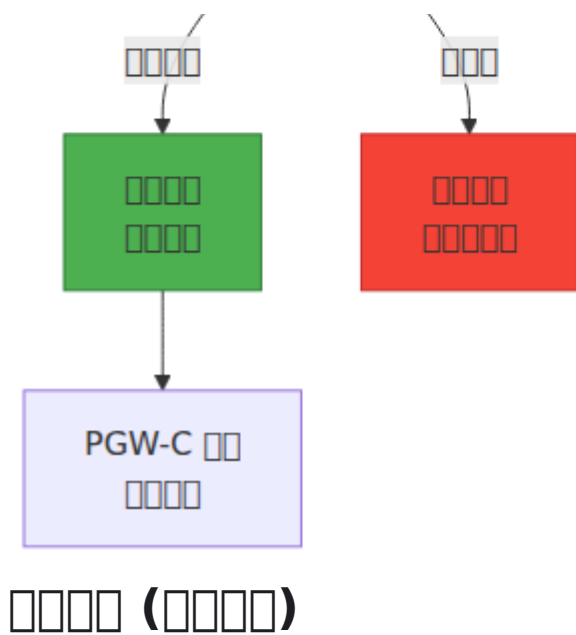
□□□□

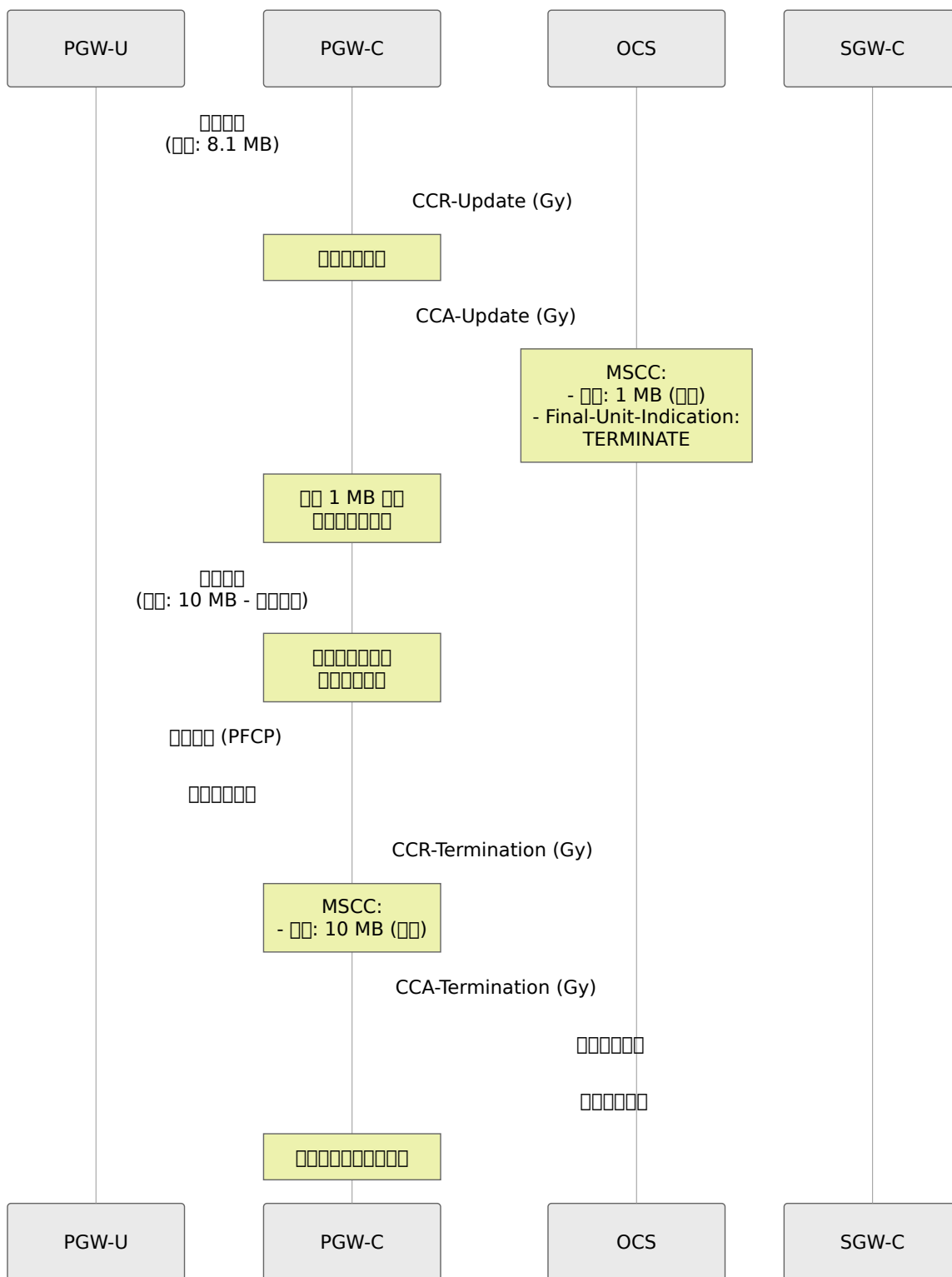
□□□□□□□□



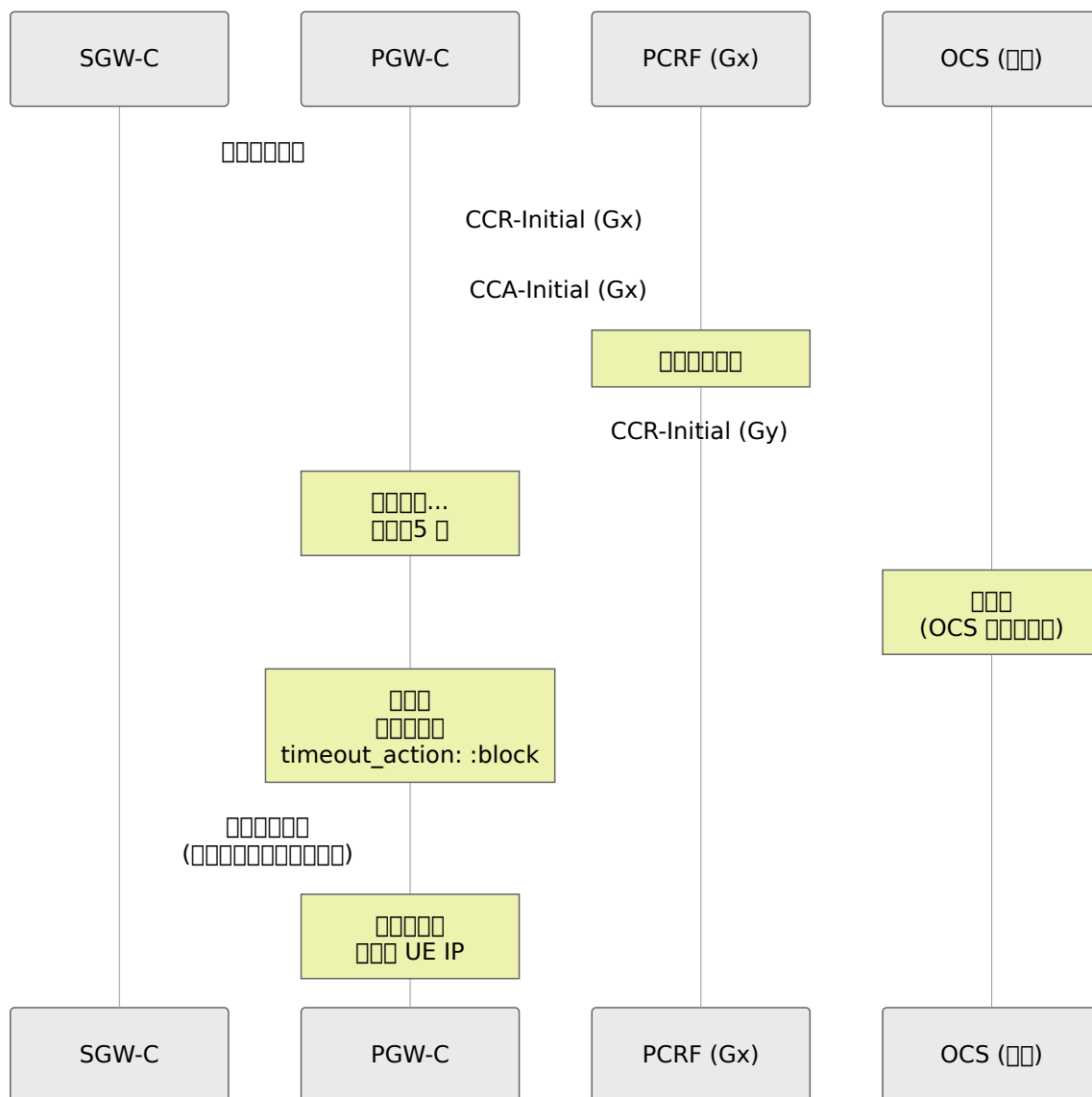
□□□□□□ (CCR-Update)







OCS □□□□



--	--	--	--

□ □ □ □

□□□□□

错误码	错误名称	错误描述
2001	DIAMETER_SUCCESS	成功

4xxx (4xxx)

错误码	错误名称	PGW-C 错误
4010	DIAMETER_TOO_BUSY	太忙
4011	DIAMETER_UNABLE_TO_COMPLY	无法遵守
4012	DIAMETER_CREDIT_LIMIT_REACHED	信用额度已用尽

5xxx (5xxx)

错误码	错误名称	PGW-C 错误
5003	DIAMETER_AUTHORIZATION_REJECTED	授权被拒绝
5031	DIAMETER_USER_UNKNOWN	用户未知

6xxx (6xxx)

Result-Code 错误码 错误描述

1. 错误码 - 错误描述
2. **MSCC** 错误 - 错误描述

错误

CCA-Initial

└─ Result-Code: DIAMETER_SUCCESS (2001) ← OK

└─ Multiple-Services-Credit-Control

└─ [Rating-Group: 100]

└─ Result-Code: DIAMETER_SUCCESS (2001) ← RG 100: OK

└─ [Rating-Group: 200]

└─ Result-Code: DIAMETER_CREDIT_LIMIT_REACHED (4012) ←

RG 200: OK

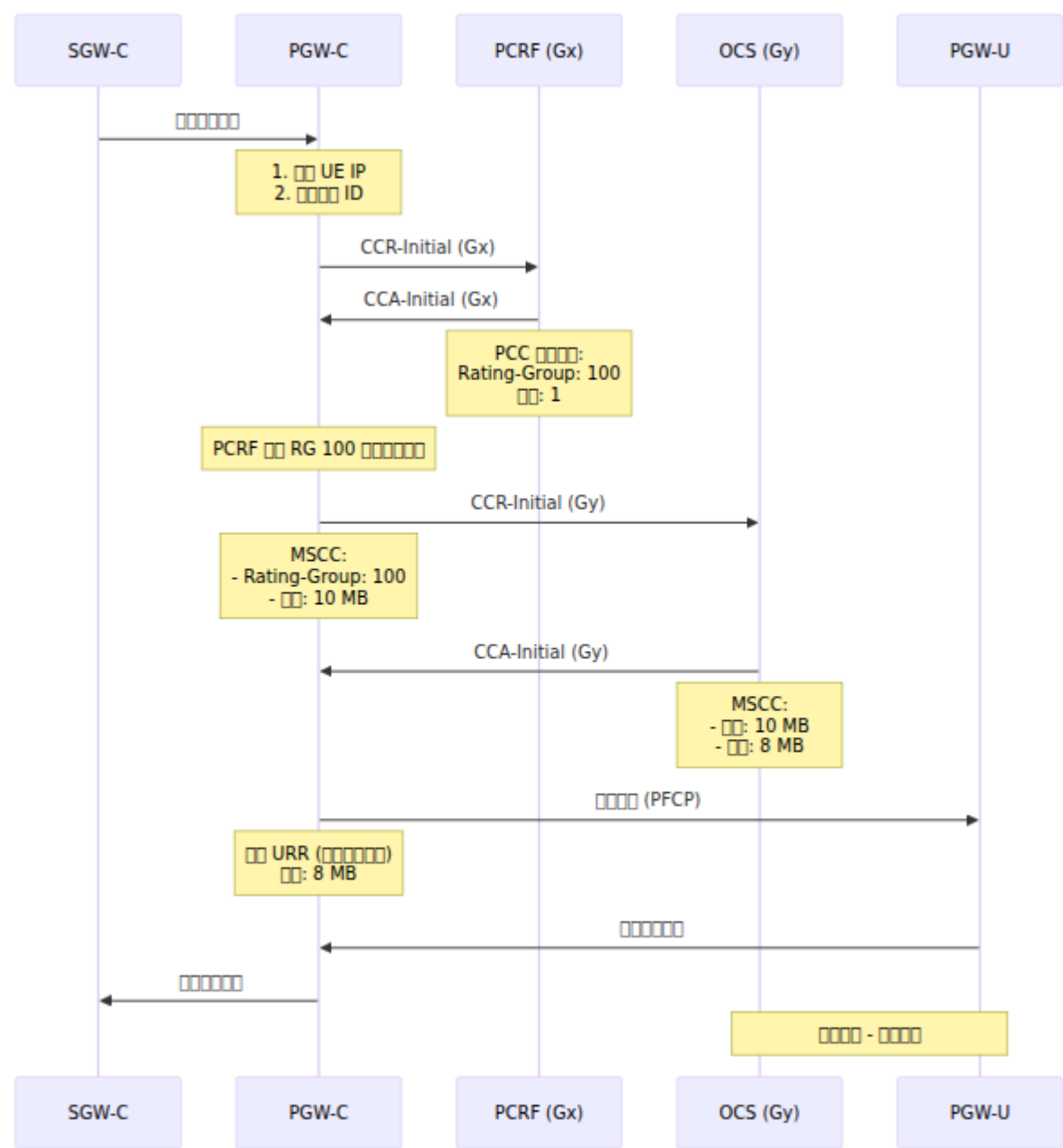
PGW-C OK

- Rating-Group 100 OK
- Rating-Group 200 OK

Gx OK

Gx OK PCRF OK Gy OK Rating-Group OK Diameter Gx OK
OK

Gx & Gy



1.

```

PGW-C   [] [] [] [] [] [] [] []
      ↓
[] CCR-I   [] PCRF   (Gx)
      ↓
[] CCA-I   [] PCC   []
      ↓
[] PCC   [] []
    -   [] [] [] Rating-Group[]
    -   [] = 1   []
      ↓
[] [] []
    [] CCR-I   [] OCS   (Gy)   [] Rating-Group
      ↓
    [] CCA-I   [] []
      ↓
    [] [] [] [] [] [] [] []
    [] [] [] [] [] [] [] []
[] [] []
    [] [] [] [] [] [] [] []

```

2. [] [] [] [] [] [] PCRF RAR [] []

```

PCRF   [] Gx   [] [] RAR [] [] [] [] [] []
      ↓
[] PCC   [] [] [] [] [] [] =1 [] Rating-Group=200
      ↓
PGW-C   [] CCR-U   [] OCS   (Gy)
    -   [] Rating-Group 200   [] MSCC
      ↓
OCS   [] [] [] [] [] [] [] []
      ↓
[] [] [] [] [] [] [] [] [] []

```

□□□□

□□□□

1. CCR-Initial □ OCS □□

□□□

- □□□□□□“OCS □□”
- □□□“CCR-Initial (Gy) □□”

□□□□□

- OCS □□□
- □□□□ OCS IP □□□
- □□□□□ Diameter □□□3868□
- OCS □□

□□□□□

```
# □□□□□□
ping <ocs_ip>

# □□ Diameter □□ (TCP 3868)
telnet <ocs_ip> 3868

# □□□□
# □□□ peer_list □□□□ OCS □□□
```

2. □□□ OCS □□

□□□

- CCA-I □ Result-Code != 2001
- □□□□□□□□

□□□□□□□□

消息	消息	消息
4012	消息	消息
5003	消息	消息
5031	消息	消息 OCS 消息

消息

1. 消息 OCS 消息
2. 消息 OCS 消息
3. 消息 CCR-I 消息 IMSI/MSISDN 消息

3. 消息

消息

- 消息
- 消息 CCR-Update

消息

- PGW-U 消息 URR消息
- 消息
- 消息 PFCP 消息

消息

1. 消息 PFCP 消息 URR

```
消息 URR
├─ URR-ID: 1
├─ Measurement-Method: VOLUME
├─ Volume-Threshold: 8000000 (8 MB)
└─ Reporting-Triggers: VOLUME_THRESHOLD
```

2. 消息 PGW-U 消息

3. 配置参数 `quota_threshold_percentage`

4. 配置 Rating-Group

配置

- OCS 配置“Rating-Group”
- 配置

配置

- CCR-I 配置 Rating-Group 与 OCS 配置
- PCRF 配置 Rating-Group

配置

1. 配置 PCRF 与 PCC 配置 Rating-Group
 2. 配置 OCS 配置 Rating-Group
 3. 配置 PCC 配置 OCS 配置
-

□□

□□□□

```
# Gy □□□□
rate(gy_inbound_messages_total{message_type="cca"}[5m])
rate(gy_outbound_messages_total{message_type="ccr"}[5m])

# Gy □□□□
rate(gy_inbound_errors_total[5m])

# □□□□□□
rate(gy_quota_exhausted_total[5m])

# OCS □□□
rate(gy_timeout_total[5m])

# Gy □□□□□□□□
histogram_quantile(0.95,
rate(gy_inbound_handling_duration_bucket[5m]))
```

11

```
# Gy
- alert: GyErrorRateHigh
  expr: rate(gy_inbound_errors_total[5m]) > 0.1
  for: 5m
  annotations:
    summary: "Gy"

# OCS
- alert: OcsTimeout
  expr: rate(gy_timeout_total[5m]) > 0.05
  for: 2m
  annotations:
    summary: "OCS"

# CreditExhaustionSpike
- alert: CreditExhaustionSpike
  expr: rate(gy_quota_exhausted_total[5m]) > 10
  for: 5m
  annotations:
    summary: "CreditExhaustionSpike"
```

Web UI - Gy

OmniPGW Gy/Ro OCS

```
□□□ http://<omnipgw-ip>:<web-port>/gy_simulator
```

0000 00000000000000000000

0000

1. 0000

- **IMSI** - 0000000000“310170123456789”0
- **MSISDN** - 0000000000“14155551234”0
- 0000 - 0000000000000000
- 00 **ID** - 00000000
- 0000 - 00000

2. CCR-I 00

- 00 CCR-Initial0000000000
- 0000000000000000
- 00 OCS 0000000000

3. 00

- 0000 - 00000000 Gy 00
- **OCS** 00 - 00 OCS 00000000
- 0000 - 0000000000

- 認證 - 認證碼
- 認證 - 認證碼

認證碼

1. 認證碼IMSI/MSISDN
2. 認證碼1000000 1 MB
3. 認證 ID 認證
4. 認證“CCR-I”
5. 認證 OCS 認證碼

認證

- 認證碼 OCS
- 認證碼
- 認證碼
- 認證碼

認證碼

認證碼

- **Diameter Gx** 認證 - PCRF 認證碼 PCC 認證
- **CDR** 認證 - 認證碼
- **認證** - 認證碼

認證碼

- **認證** - PDN 認證碼
- **PFCP** 認證 - 認證 URR 認證 PGW-U 認證碼
- **S5/S8** 認證 - GTP-C 認證碼

□□

- □□□□ - Gy □□□□□□□□OCS □□□□
 - **UE IP** □□ - □□□□□ IP □□□
-

□□□□□□

OmniPGW 部署架构图

Prometheus 部署图

OmniTouch 部署图

部署

1. 部署
2. 部署
3. 部署
4. Prometheus 部署
5. Grafana 部署
6. 部署
7. 部署
8. 部署

部署

OmniPGW 部署架构图

1. Web UI 部署架构图

- 部署
- PFCP 部署
- Diameter 部署
- 部署

2. Prometheus 部署架构图

- 部署
- 部署

- ☐ ☐ ☐ ☐
- ☐ ☐ ☐ ☐

📦 Prometheus 📦 Web UI 📦📦📦📦📦

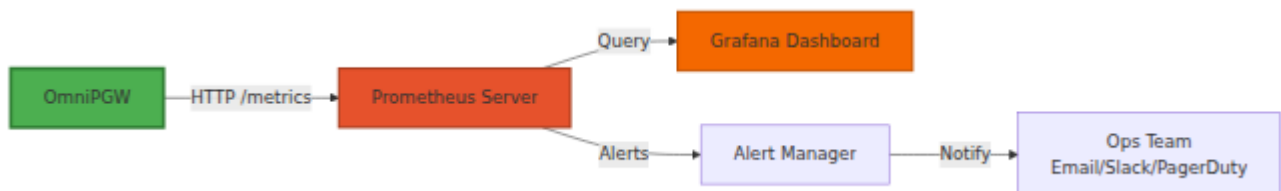
- 3GPP - Web UI
- PFCP 3GPP - Web UI
- Diameter Gx - Web UI

Prometheus ☐☐☐☐

OmniPGW Prometheus

- 1000000 - 1000000000000000
- 10000 - 10000000000
- 100000 - 10000000000
- 100 - 10000000
- 100 - 10000000000000

□ □ □ □



□ □ □ □

11

```
❏ config/runtime.exs ❏❏❏❏❏
```

```
config :pgw_c,  
  metrics: %{  
    enabled: true,  
    ip_address: "0.0.0.0", # 0.0.0.0  
    port: 9090, # HTTP 9090  
    registry_poll_period_ms: 5_000 # 5000ms  
  }
```

0.0.0.0

HTTP 9090

```
http://<omnipgw_ip>:<port>/metrics
```

curl

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

0.0.0.0

0.0.0.0 **Prometheus** 0.0.0.0 0.0.0

```
# 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
```

□□□□

OmniPGW □□□□□□□□

□□□□

□□□□□□

項目名	タイプ	説明
teid_registry_count	Gauge	S5/S8 TEID の数
seid_registry_count	Gauge	PFCP SEID の数
session_id_registry_count	Gauge	Gx Diameter Session-ID の数
session_registry_count	Gauge	IMSI, EBI の数
address_registry_count	Gauge	UE IP の数
charging_id_registry_count	Gauge	ID の数 CDR の数 CDR の数
sxb_sequence_number_registry_count	Gauge	PFCP の数
s5s8_sequence_number_registry_count	Gauge	S5/S8 の数
sxb_peer_registry_count	Gauge	PFCP の数

```
# 項目名
teid_registry_count

# 単位
rate(teid_registry_count[5m])

# 単位
max_over_time(teid_registry_count[1h])
```

S5/S8 (GTP-C)

구분	유형	필드	설명
s5s8_inbound_messages_total	Counter	message_type	구분 S5/S8 인 계
s5s8_outbound_messages_total	Counter	message_type	구분 S5/S8 출 계
s5s8_inbound_errors_total	Counter	message_type	S5/S8 오류

구분

- create_session_request
- create_session_response
- delete_session_request
- delete_session_response
- create_bearer_request
- delete_bearer_request

Sxb (PCFP) 구분

구분	유형	필드	설명
sxb_inbound_messages_total	Counter	message_type	구분 PCFP 인 계
sxb_outbound_messages_total	Counter	message_type	구분 PCFP 출 계
sxb_inbound_errors_total	Counter	message_type	PCFP 오류
sxb_outbound_errors_total	Counter	message_type	PCFP 오류

구분

- association_setup_request
- association_setup_response
- heartbeat_request

- heartbeat_response
- session_establishment_request
- session_establishment_response
- session_modification_request
- session_deletion_request

Gx (Diameter) 消息

消息名称	类型	消息	说明
gx_inbound_messages_total	Counter	message_type	统计 Diameter 消息接收数量
gx_outbound_messages_total	Counter	message_type	统计 Diameter 消息发送数量
gx_inbound_errors_total	Counter	message_type	统计 Diameter 消息接收错误数量
gx_outbound_errors_total	Counter	message_type	统计 Diameter 消息发送错误数量
gx_outbound_responses_total	Counter	message_type, result_code_class, diameter_host	统计 Diameter 消息发送响应数量

消息类型

- gx_CCA (Credit-Control-Answer)
- gx_CCR (Credit-Control-Request)
- gx_RAA (Re-Auth-Answer)
- gx_RAR (Re-Auth-Request)

🔍🔍🔍🔍🔍🔍 `gx_outbound_responses_total` 🔍🔍

- `2xxx` - 📄📄📄📄📄2001 DIAMETER_SUCCESS📄
- `3xxx` - 📄📄📄📄📄3001 DIAMETER_COMMAND_UNSUPPORTED📄
- `4xxx` - 📄📄📄📄📄4001 DIAMETER_AUTHENTICATION_REJECTED📄
- `5xxx` - 📄📄📄📄📄5012 DIAMETER_UNABLE_TO_COMPLY📄

🔍🔍🔍🔍

```
# 📄 Gx 📄📄📄📄
sum(rate(gx_outbound_responses_total{result_code_class="2xxx"}[5m]))
sum(rate(gx_outbound_responses_total[5m])) * 100

# 📄 PCRF 📄📄📄📄
rate(gx_outbound_responses_total{result_code_class!="2xxx"}[5m]) by (

# 📄📄📄 Re-Auth-Answer 📄📄📄
gx_outbound_responses_total{message_type="gx_RAA",result_code_class='

# 📄📄📄 PCRF 📄📄📄📄📄📄
rate(gx_outbound_responses_total{result_code_class=~"4xxx|5xxx",diame
[5m]) > 0.1
```

🔍🔍🔍🔍

📄📄📄	📄📄	📄📄	📄📄
<code>rescues_total</code>	Counter	<code>module</code> , <code>function</code>	📄📄📄📄📄📄📄📄📄📄

🔍🔍🔍🔍

📄📄📄📄📄📄📄📄📄📄

項目名	種類	単位	説明
s5s8_inbound_handling_duration	Histogram	request_message_type	5番台5番台の着信処理時間
sxb_inbound_handling_duration	Histogram	request_message_type	5番台5番台の着信処理時間
gx_inbound_handling_duration	Histogram	request_message_type	5番台5番台の着信処理時間

項目名

項目名	種類	単位	説明
s5s8_outbound_transaction_duration	Histogram	request_message_type	5番台5番台の出発取引時間
sxb_outbound_transaction_duration	Histogram	request_message_type	5番台5番台の出発取引時間
gx_outbound_transaction_duration	Histogram	request_message_type	5番台5番台の出発取引時間

項目名

- 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

例

```
# 95th S5/S8
histogram_quantile(0.95,
  rate(s5s8_inbound_handling_duration_bucket[5m])
)

# PFCP
rate(sxb_inbound_handling_duration_sum[5m]) /
rate(sxb_inbound_handling_duration_count[5m])
```

UPF

UPF

名前	タイプ	単位	説明
upf_peers_total	Gauge	-	UPF のピア数
upf_peers_healthy	Gauge	-	UPF の健康なピア数 + 0
upf_peers_unhealthy	Gauge	-	UPF の不健康なピア数
upf_peers_associated	Gauge	-	PFCP と UPF が関連しているピア数
upf_peers_unassociated	Gauge	-	PFCP と UPF が関連していないピア数
upf_peer_healthy	Gauge	peer_ip	UPF の健康なピア数1=0=0
upf_peer_missed_heartbeats	Gauge	peer_ip	UPF の健康なピア数

000

```
# 00 UPF 000
upf_peers_healthy / upf_peers_total

# 000000 UPF 0000
upf_peers_unhealthy > 0

# 0000 UPF 00
upf_peer_healthy{peer_ip="10.98.0.20"}

# 00000000 UPF
upf_peer_missed_heartbeats > 2
```

00000

```

# UPF
- alert: UPF_Peer_Down
  expr: upf_peer_healthy == 0
  for: 1m
  labels:
    severity: critical
  annotations:
    summary: "UPF {{ $labels.peer_ip }} "
    description: "UPF PFCP "

# UPF
- alert: UPF_Pool_Degraded
  expr: (upf_peers_healthy / upf_peers_total) < 0.5
  for: 2m
  labels:
    severity: critical
  annotations:
    summary: "UPF "
    description: "{{ $value | humanizePercentage }} UPF "

#
- alert: UPF_Heartbeat_Issues
  expr: upf_peer_missed_heartbeats > 2
  for: 30s
  labels:
    severity: warning
  annotations:
    summary: "UPF {{ $labels.peer_ip }} "
    description: "{{ $value }} "

```

P-CSCF

P-CSCF

메트릭	유형	단위	설명
pcscf_fqdns_total	Gauge	-	모든 P-CSCF FQDN 개수
pcscf_fqdns_resolved	Gauge	-	해결된 DNS 개수 P-CSCF FQDN
pcscf_fqdns_failed	Gauge	-	실패한 P-CSCF FQDN
pcscf_servers_total	Gauge	-	P-CSCF 서버 개수
pcscf_servers_healthy	Gauge	fqdn	정상 FQDN P-CSCF 서버
pcscf_servers_unhealthy	Gauge	fqdn	비정상 FQDN P-CSCF 서버

모든 P-CSCF 서버는 IMS 서버

모든 서버

모든 서버

메트릭	유형	단위
license_status	Gauge	0 = 실패, 1 = 성공

모든

```
# 모든 서버
license_status == 1

# 모든 서버
license_status == 0
```

모든

```
- alert: PGW_C_License_Invalid
  expr: license_status == 0
  for: 1m
  labels:
    severity: critical
  annotations:
    summary: "PGW-C 许可证过期"
    description: "许可证过期 - 许可证过期"
```

许可证过期

许可证过期是指PGW-C的许可证过期，GTP-C 许可证 “许可证” (73) 许可证过期
许可证过期

Wireshark 许可证过期许可证过期“许可证”

许可证

- 许可证过期许可证过期 omnipgwc
- 许可证 URL 为 config/runtime.exs 许可证 :license_client 许可证
- 许可证 license_status == 0 许可证过期许可证过期 GTP-C 许可证 73 许可证过期
- 许可证许可证 UI 许可证许可证
- Diameter 许可证 GTP-C 许可证 PFCP 许可证许可证
- 许可证许可证 - 许可证许可证

📄📄📄📄

Erlang VM 📄📄

📄📄📄	📄📄	📄📄
<code>vm_memory_total</code>	Gauge	📄 VM 📄📄📄📄
<code>vm_memory_processes</code>	Gauge	📄📄📄📄📄
<code>vm_memory_system</code>	Gauge	📄📄📄📄📄
<code>vm_system_process_count</code>	Gauge	📄 Erlang 📄📄
<code>vm_system_port_count</code>	Gauge	📄📄📄📄

Prometheus 📄📄

📄📄📄📄

📄 OmniPGW 📄📄 Prometheus `prometheus.yml` 📄

```
# 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'
```

❏❏ OmniPGW ❏❏

```
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'
```

❏❏❏❏

Kubernetes❏

```
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'
```

❏❏

❏❏❏❏❏

```
# ❏❏ Prometheus ❏❏
curl http://prometheus:9090/api/v1/targets

# ❏❏❏❏❏❏
curl 'http://prometheus:9090/api/v1/query?
query=teid_registry_count'
```

Grafana 安装

安装步骤

1. 安装 Prometheus 服务

安装 → 配置 → 启动 Prometheus
URL: <http://prometheus:9090>

2. 配置 Grafana

配置 Grafana 连接 Prometheus JSON 数据源

配置步骤

配置 1. 添加数据源

```
# 名称
teid_registry_count

# 选择Gauge
# 单位
# 范围 < 5000
# 范围 5000-8000
# 范围 > 8000
```

配置 2. 添加仪表盘

```
# 名称
rate(s5s8_inbound_messages_total{message_type="create_session_request"}[5m])

# 选择仪表盘
# 选择仪表盘/模板
```

例 3 IP 割合

```
# 10.0.0.0 /24 の IP 数
(address_registry_count / 254) * 100

# Gauge
# 0-100
#
# < 70%
# 70-85%
# > 85%
```

例 4 95th 分位数

```
#
histogram_quantile(0.95,

rate(s5s8_inbound_handling_duration_bucket{request_message_type="create"}
[5m])
)

#
#
```

例 5

```
#
rate(s5s8_inbound_errors_total[5m])

#
# /
# > 0.1
```

例 6 Gx

```
# 計算Gx の割合
sum(rate(gx_outbound_responses_total{result_code_class="2xxx"}
[5m])) /
sum(rate(gx_outbound_responses_total[5m])) * 100

# Gauge
# 0-100
#
# > 95%
# 90-95%
# < 90%
```

図 - 計算結果

```
# 計算結果
sum(rate(gx_outbound_responses_total[5m])) by (result_code_class)

#
# {{ result_code_class }}
```

図 - **PCRf** の割合

```
# PCRf の割合
sum(rate(gx_outbound_responses_total[5m])) by (diameter_host,
result_code_class)

#
# {{ diameter_host }} - {{ result_code_class }}
```

図 **7UPF** の割合

```
# 仪表盘百分比
(upf_peers_healthy / upf_peers_total) * 100

# 仪表盘Gauge
# 仪表盘范围0-100
# 单位
# 100%
# 50-99%
# < 50%
```

UPF 仪表盘

```
# 仪表盘 UPF 仪表盘
upf_peer_healthy

# 仪表盘Stat
# 单位
# 1 = "UP"仪表盘
# 0 = "DOWN"仪表盘
```

ダッシュボード

```
{
  "dashboard": {
    "title": "OmniPGW - ダッシュボード",
    "panels": [
      {
        "title": "登録数",
        "targets": [
          {
            "expr": "teid_registry_count",
            "legendFormat": "登録数"
          }
        ],
        "type": "graph"
      },
      {
        "title": "セッション数",
        "targets": [
          {
            "expr":
"rate(s5s8_inbound_messages_total{message_type=\"create_session_request\"}[5m])",
            "legendFormat": "セッション数"
          }
        ],
        "type": "graph"
      },
      {
        "title": "IP 利用率",
        "targets": [
          {
            "expr": "(address_registry_count / 254) * 100",
            "legendFormat": "利用率 %"
          }
        ],
        "type": "gauge"
      },
      {
        "title": "遅延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"
}
]
```

□□

□□□□

□□ `omnipgw_alerts.yml` □

```

groups:
- name: omnipgw
  interval: 30s
  rules:
    # 会话数
    - alert: OmniPGW_HighSessionCount
      expr: teid_registry_count > 8000
      for: 5m
      labels:
        severity: warning
      annotations:
        summary: "OmniPGW 会话数高"
        description: "{{ $value }}"
    - alert: OmniPGW_SessionCountCritical
      expr: teid_registry_count > 9500
      for: 2m
      labels:
        severity: critical
      annotations:
        summary: "OmniPGW 会话数临界"
        description: "{{ $value }}"

    # IP 地址
    - alert: OmniPGW_IPPoolUtilizationHigh
      expr: (address_registry_count / 254) * 100 > 80
      for: 10m
      labels:
        severity: warning
      annotations:
        summary: "OmniPGW IP 地址池利用率高"
        description: "IP 地址池 {{ $value }}% 利用率"
    - alert: OmniPGW_IPPoolExhausted
      expr: address_registry_count >= 254
      for: 1m
      labels:
        severity: critical
      annotations:
        summary: "OmniPGW IP 地址池耗尽"
        description: "IP 地址池耗尽"

    # 异常

```

```

- alert: OmniPGW_HighErrorRate
  expr: rate(s5s8_inbound_errors_total[5m]) > 0.1
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "OmniPGW S5/S8"
    description: "{{ $value }}"

- alert: OmniPGW_GxErrorRate
  expr: rate(gx_inbound_errors_total[5m]) > 0.05
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "OmniPGW Gx"
    description: "{{ $value }}" Diameter

# 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: "OmniPGW Gx"
    description: "{{ $value | humanizePercentage }}" Gx
    "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 }}"
    description: "{{ $value }}" PCRF {{
$labels.diameter_host }}"

```

```

# UPF 告警
- alert: OmniPGW_UPF_PeerDown
  expr: upf_peer_healthy == 0
  for: 1m
  labels:
    severity: critical
  annotations:
    summary: "UPF 告警 {{ $labels.peer_ip }} 故障"
    description: "UPF 告警 PFCP 告警"

- alert: OmniPGW_UPF_PoolDegraded
  expr: (upf_peers_healthy / upf_peers_total) < 0.5
  for: 2m
  labels:
    severity: critical
  annotations:
    summary: "UPF 告警"
    description: "{{ $value | humanizePercentage }} % UPF 告警 (< 50%)"

- alert: OmniPGW_UPF_HeartbeatFailures
  expr: upf_peer_missed_heartbeats > 2
  for: 30s
  labels:
    severity: warning
  annotations:
    summary: "UPF {{ $labels.peer_ip }} 心跳失败"
    description: "{{ $value }} 心跳失败"

- alert: OmniPGW_UPF_AllDown
  expr: upf_peers_healthy == 0 and upf_peers_total > 0
  for: 30s
  labels:
    severity: critical
  annotations:
    summary: "告警 UPF 告警"
    description: "告警 UPF 告警"

# 告警
- alert: OmniPGW_HighLatency
  expr: |
    histogram_quantile(0.95,
      rate(s5s8_inbound_handling_duration_bucket[5m])

```

```

    ) > 100000
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "OmniPGW 高延迟"
    description: "p95 延迟 {{ $value }}μs (> 100ms)"

# 内存使用
- alert: OmniPGW_HighMemoryUsage
  expr: vm_memory_total > 2000000000
  for: 10m
  labels:
    severity: warning
  annotations:
    summary: "OmniPGW 高内存使用"
    description: "VM 内存 {{ $value | humanize }}B 使用"

- alert: OmniPGW_HighProcessCount
  expr: vm_system_process_count > 100000
  for: 10m
  labels:
    severity: warning
  annotations:
    summary: "OmniPGW 高进程数"
    description: "{{ $value }} Erlang 进程数"

```

AlertManager ☐☐

```
# 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'

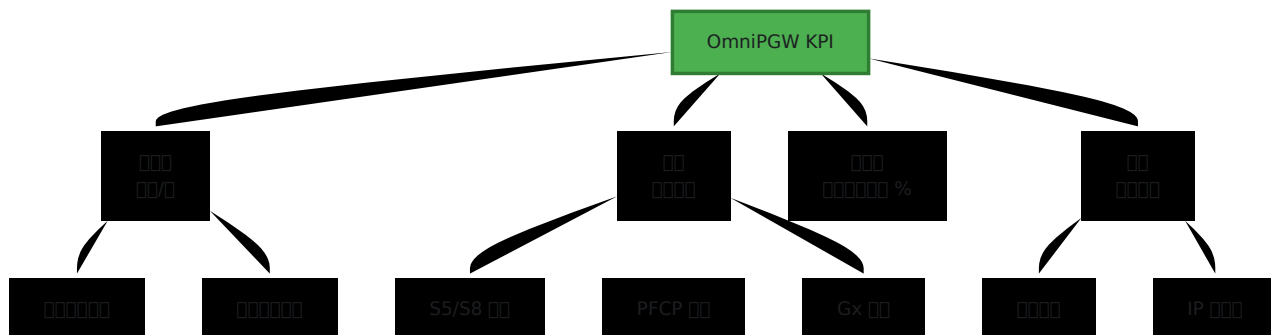
receivers:
  - name: 'ops-team'
    email_configs:
      - to: 'ops@example.com'

  - name: 'slack'
    slack_configs:
      - api_url:
        'https://hooks.slack.com/services/YOUR/SLACK/WEBHOOK'
        channel: '#omnipgw-alerts'
        title: 'OmniPGW ☐☐☐{{ .GroupLabels.alertname }}'
        text: '{{ range .Alerts }}{{ .Annotations.description }}{{
end }}'

  - name: 'pagerduty'
    pagerduty_configs:
      - service_key: 'YOUR_PAGERDUTY_KEY'
```

□□□□

□□□□□□ (KPI)



□□□□□

□□□□□□

```
rate(s5s8_inbound_messages_total{message_type="create_session_request"
[5m] )
```

□□□□□□

```
rate(s5s8_inbound_messages_total{message_type="delete_session_request"
[5m] )
```

□□□□□

```
rate(s5s8_inbound_messages_total{message_type="create_session_request"
[5m] ) -
rate(s5s8_inbound_messages_total{message_type="delete_session_request"
[5m] )
```

□□□□

□□□□□□□□□□□□

```
# p50
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])
)
```

~~~~~

```
histogram_quantile(0.95,
  rate(s5s8_inbound_handling_duration_bucket[5m])
) by (request_message_type)
```

~~~~~

~~~~~**24**~~~~~

```
teid_registry_count -
teid_registry_count offset 24h
```

~~~~~

```
# 10,000
10000 - teid_registry_count
```

~~~~~

```
# 00 1 0000000000000000  
(10000 - teid_registry_count) /  
(rate(teid_registry_count[1h]) * 86400)
```

|  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|

111

```
rate(s5s8_inbound_errors_total[5m]) by (message_type)
```

103

- 5G
- 4G PCRF 4G Gx 4G
- 4G IP 4G

|  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|

111

```
histogram_quantile(0.95,
```

```
rate(s5s8_inbound_handling_duration_bucket{request_message_type="create",
[5m])
)
```

1111

- Gx PCRF
- PFCP PGW-U

- 网络性能指标

## 网络PCRFB指标

网络

```
# 网络 Gx 性能指标
sum(rate(gx_outbound_responses_total{result_code_class!="2xxx"}
[5m])) /
sum(rate(gx_outbound_responses_total[5m])) * 100

# 网络 PCRFB 指标
sum(rate(gx_outbound_responses_total[5m])) by (diameter_host,
result_code_class)

# 网络性能指标
rate(gx_outbound_responses_total{result_code_class="5xxx"}[5m]) by
(diameter_host)
```

网络

- 网络 PCRFB 性能指标
- 网络 PCRFB 性能指标5xxx 性能指标
- 网络 Diameter 性能指标
- 网络 PCRFB 性能指标
- 网络 5012DIAMETER\_UNABLE\_TO\_COMPLY性能指标 Re-Auth-Request 性能指标

网络性能指标

网络

```
# 网络性能指标
rate(vm_memory_total[1h])

# 网络性能指标
rate(vm_memory_processes[1h])

# 网络性能指标
rate(vm_system_process_count[1h])
```

□□□

- □□□□□□
- □□□□□□
- □□□□□□□□□□

□□□□

□□□□□□□□□□

```
max_over_time(teid_registry_count[24h])
```

□□□□□□□□

```
teid_registry_count /  
avg_over_time(teid_registry_count[7d])
```

□□□□□

```
abs(  
    teid_registry_count -  
    avg_over_time(teid_registry_count[1h])  
) > 100
```

---

□□□□

□□□□

1. □□□□□ 15-30 □□□□□□□□□□
2. □□□ 15 □□□□□□□□□□
3. □□□ □□□□□□□□□□□□□□□□□□□

## Network

1. Network - NOC KPI
2. Network - Performance
3. Network - Security

## Cloud

1. Cloud - Infrastructure
2. Cloud - VM → Container
3. Cloud - Kubernetes

## Security

### Network

- Network - Prometheus  Web UI
- Network - Security

### Cloud

- **PFCP** - PFCP between UPF
- **Diameter Gx** - Gx between PCRF
- **Diameter Gy** - Gy between OCS
- **S5/S8** - GTP-C between SGW-C

### Network

- **P-CSCF** - P-CSCF in IMS
- Network - Security
- **UE IP** - IP address

### Network

---

**OmniPGW** □□□□ - □ *Omnitouch* □□□□□□

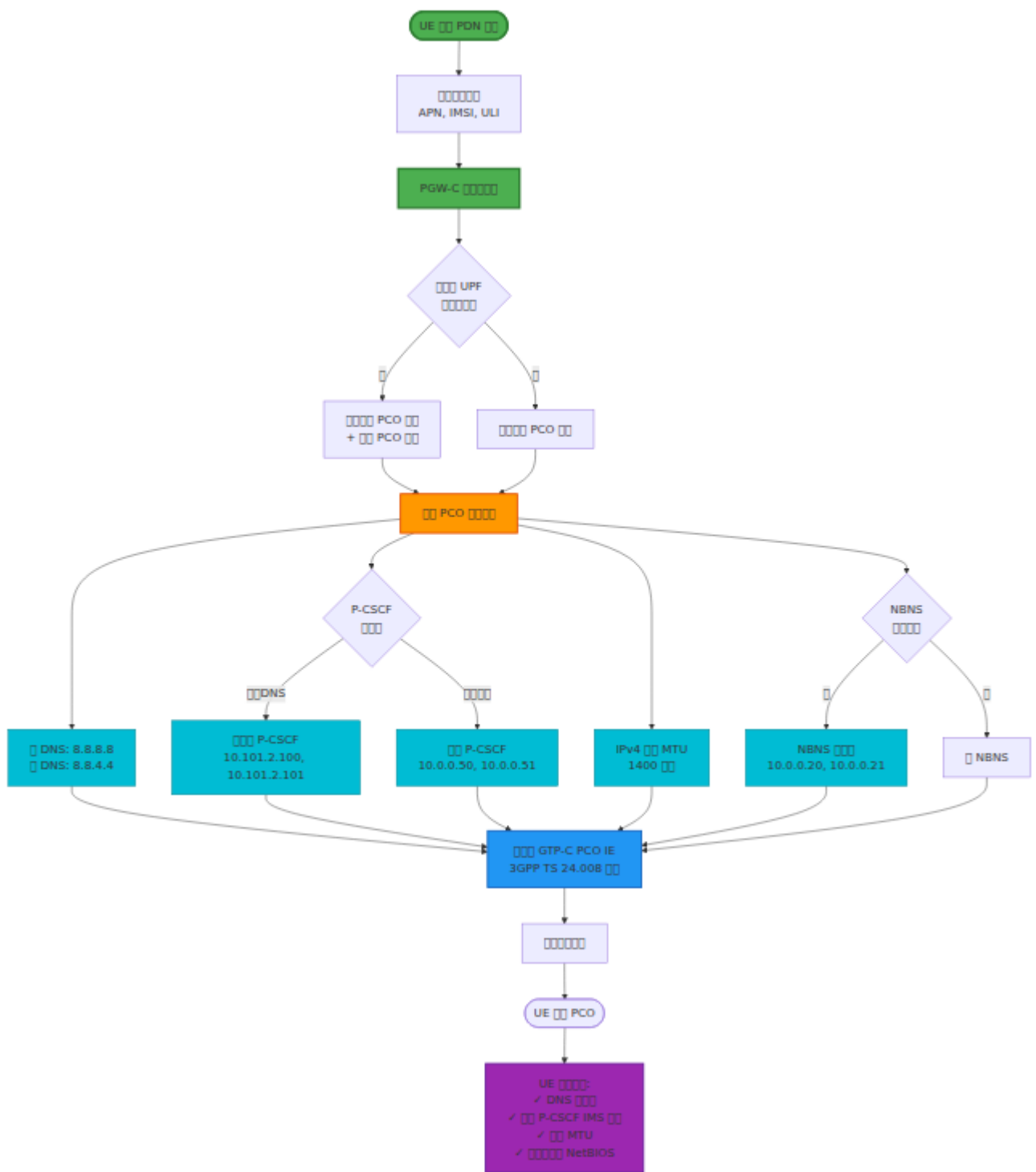
# PCO (PCO)

UE UE

OmniPGW OmniTouch

---

PCO (PCO) PDN UE (UE) UE DNS IMS



**PCO**

| IE 名称           | 选项 ID  | 描述            | 备注       |
|-----------------|--------|---------------|----------|
| DNS 选项 IPv4 地址  | 0x000D | IP DNS        | IP       |
| DNS 选项 IPv4 地址  | 0x000D | IP DNS        | IP       |
| P-CSCF IPv4 地址  | 0x000C | IMS IP P-CSCF | IP (IMS) |
| IPv4 选项 MTU     | 0x0010 | MTU 选项        | IP       |
| NBNS 选项 IPv4 地址 | 0x0011 | NetBIOS 选项    | IP       |

---

□□

□□□□

```
# config/runtime.exs
config :pgw_c,
  pco: %{
    # DNS □□□ (□□)
    primary_dns_server_address: "8.8.8.8",
    secondary_dns_server_address: "8.8.4.4",

    # NBNS □□□ (□□□□□□□ Windows □□)
    primary_nbns_server_address: nil,
    secondary_nbns_server_address: nil,

    # IMS/VoLTE □ P-CSCF □□ (□□)
    p_cscf_ipv4_address_list: [],

    # P-CSCF □□□□ (□□)
    p_cscf_discovery_enabled: false,
    p_cscf_discovery_dns_server: nil,
    p_cscf_discovery_timeout_ms: 5000,

    # IPv4 MTU □□ (□□)
    ipv4_link_mtu_size: 1400
  }
```

---

**PCO** □□

**DNS** □□□□□

□ **DNS** □□ **DNS**□

```
pco: %{\n  primary_dns_server_address: "8.8.8.8",\n  secondary_dns_server_address: "8.8.4.4"\n}
```

## DNS

|            | DNS            | DNS             |
|------------|----------------|-----------------|
| 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

```
pco: %{\n  primary_dns_server_address: "10.0.0.10",\n  secondary_dns_server_address: "10.0.0.11"\n}
```

## P-CSCF (IMS)

### IMS/VoLTE

```
pco: %{\n  p_cscf_ipv4_address_list: [\n    "10.0.0.50", # P-CSCF\n    "10.0.0.51"  # P-CSCF\n  ]\n}
```

### P-CSCF ( )

- IMS 認證
- VoLTE/VoWiFi/RCS 認證
- UE 認證 SIP

## P-CSCF 認證

### PGW-C DNS 與 P-CSCF 認證

OmniPGW 透過 DNS 認證 P-CSCF 認證 3GPP TS 23.003 與 TS 24.229 認證 PGW-C 透過 DNS 與 P-CSCF 認證

```
pco: %{
  # 認證 P-CSCF 認證
  p_cscf_discovery_enabled: true,

  # P-CSCF 認證 DNS 認證 (認證)
  p_cscf_discovery_dns_server: {10, 179, 2, 177},

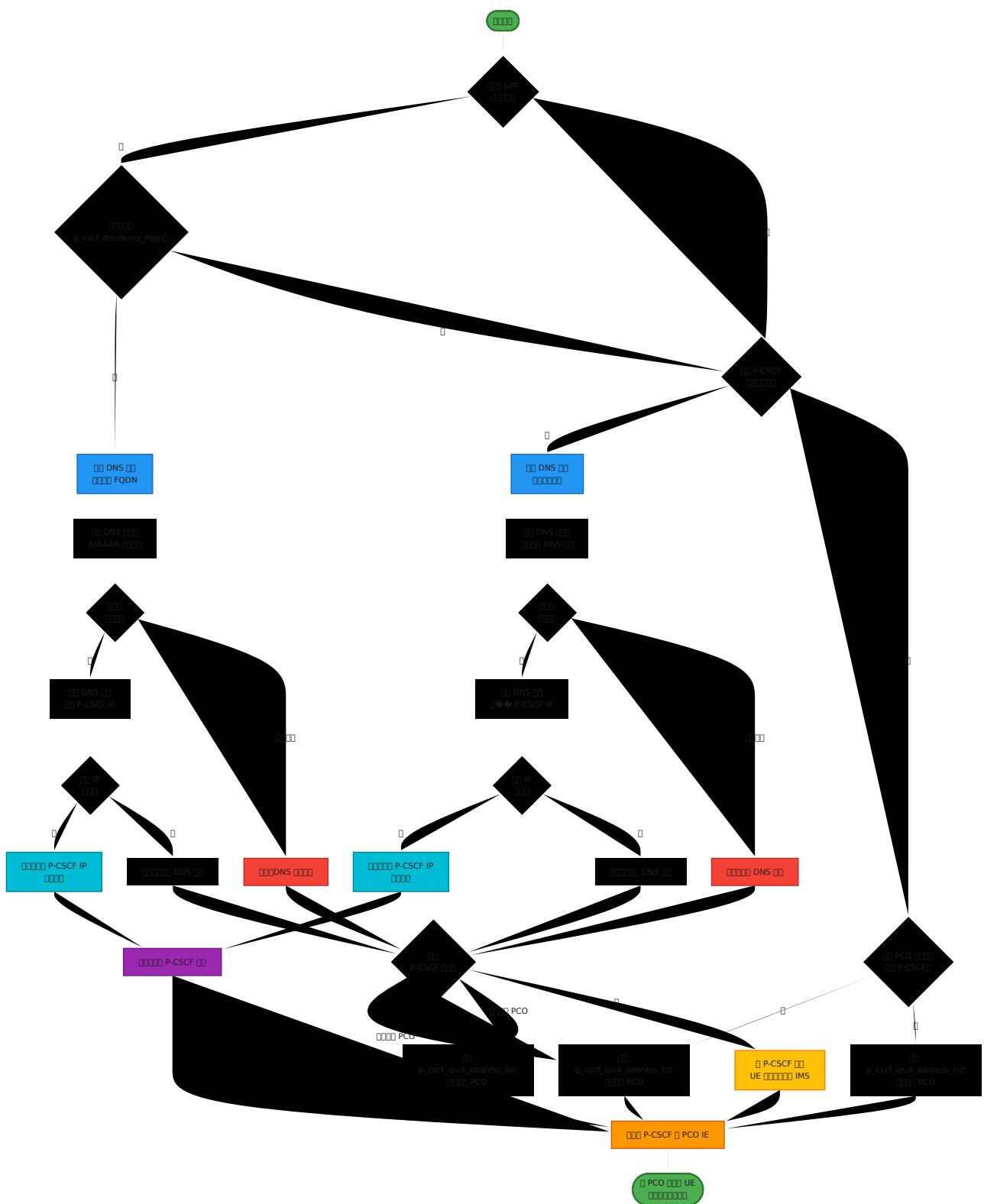
  # DNS 認證 (認證)
  p_cscf_discovery_timeout_ms: 5000,

  # 認證 P-CSCF 認證 (認證 DNS 認證)
  p_cscf_ipv4_address_list: ["10.0.0.50"]
}
```

### 認證

1. `p_cscf_discovery_enabled: true` PGW-C 認證 DNS 認證 P-CSCF 認證
2. DNS 認證 `p_cscf_discovery_dns_server`
3. 認證 DNS 認證 P-CSCF 認證 PCO 認證 UE
4. 認證 DNS 認證 `p_cscf_ipv4_address_list`
5. 認證 P-CSCF 認證

## P-CSCF ☐☐☐☐



□ □ □ □ □ □

1. **FQDN** (FQDN) - `p_cscf_discovery_fqdn` UPF FQDN
2. **DNS** (DNS) - `p_cscf_discovery_enabled: true` PCO DNS

3. 在 **PCO** 配置 - `p_cscf_ipv4_address_list` 配置 PCO 配置
4. 在 **PCO** 配置 (IP) - `p_cscf_ipv4_address_list` 配置 PCO 配置

配置

在 P-CSCF 配置配置配置配置

- DNS 配置/配置
- 配置
- 配置配置
- 配置配置配置

配置配置配置配置配置 **P-CSCF** 配置

配置配置

| 配置                                       | 配置      | 配置                 | 配置                                       |
|------------------------------------------|---------|--------------------|------------------------------------------|
| <code>p_cscf_discovery_enabled</code>    | 配置      | <code>false</code> | 配置 DNS 配置 P-CSCF 配置                      |
| <code>p_cscf_discovery_dns_server</code> | 配置 (IP) | <code>nil</code>   | DNS 配置 IP 配置 4 配置 (配置 {10, 179, 2, 177}) |
| <code>p_cscf_discovery_timeout_ms</code> | 配置      | <code>5000</code>  | DNS 配置配置配置                               |

配置

- 在 **IMS** 配置 - P-CSCF 配置 DNS 配置
- 配置配置 - DNS 配置配置 P-CSCF 配置
- 配置配置 - DNS 配置配置配置 P-CSCF 配置
- 配置配置 - 配置配置配置配置 P-CSCF 配置

配置配置 **DNS** 配置配置 **IMS**

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

  # P-CSCF
  p_cscf_discovery_enabled: true,
  p_cscf_discovery_dns_server: {10, 179, 2, 177}, # IMS DNS
  p_cscf_discovery_timeout_ms: 3000,

  # P-CSCF (DNS)
  p_cscf_ipv4_address_list: [
    "10.0.0.50", #
    "10.0.0.51" #
  ],

  ipv4_link_mtu_size: 1400
}
```

## P-CSCF

P-CSCF UPF APN DNS P-CSCF

```
# upf_selection
rules: [
  %{
    name: "IMS",
    priority: 20,
    match_field: :apn,
    match_regex: "^ims",
    upf_pool: [...],

    # P-CSCF
    p_cscf_discovery_fqdn: "pcscf.mnc380.mcc313.3gppnetwork.org"
  }
]
```

P-CSCF UPF

P-CSCF P-CSCF

# NBNS 配置 (NetBIOS)

## Windows 配置

```
pco: %{\n  primary_nbns_server_address: "10.0.0.20",\n  secondary_nbns_server_address: "10.0.0.21"\n}
```

配置

- Windows 配置
- 
- NetBIOS 配置

## MTU 配置

配置

```
pco: %{\n  ipv4_link_mtu_size: 1400 # \n}
```

## MTU 配置

| MTU  |          |
|------|----------|
| 1500 |          |
| 1400 | GTP      |
| 1420 |          |
| 1280 | IPv6 MTU |
| 1360 | VPN/     |

□□□ □□ LTE □□ **1400** □□□ GTP-U □□□

---

□□□□

## Internet APN

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

## IMS APN

```
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
}
```

□□ **P-CSCF** □□ □□□ IMS □□□□□□ P-CSCF □□□□

□□ **APN**

```
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 □ GTP-C □□□□□□

□ □ □ □ □ □

OmniPGW □ □□□□□ □□□□□ PCO□

```

| | | | |
|   | | : | | | |
|   UE IP | | : 100.64.1.42
|   PC0 (| | | | |)
|       | | DNS | | IPv4 | | : 8.8.8.8
|       | | DNS | | IPv4 | | : 8.8.4.4
|       | | P-CSCF IPv4 | | : 10.0.0.50
|       | | P-CSCF IPv4 | | : 10.0.0.51
|       | | IPv4 | | MTU : 1400

```

UE ☐ ☐

UE □□ PCO □□

1. DNS
2. P-CSCF IMS
3. MTU

## 問題

### UE の DNS

問題

- UE の IP アドレスを確認
- DNS の設定

手順

1. PCO の DNS の設定を確認
2. UE の IP アドレスを確認
3. DNS の設定を確認

確認

```
# DNS の設定を確認
ping 8.8.8.8

# UE の DNS の設定を確認
nslookup google.com 8.8.8.8

# PCO の設定を確認
grep "primary_dns_server_address" config/runtime.exs
```

### IMS の設定

問題

- VoLTE の設定
- UE の IMS の設定

手順

1. P-CSCF の設定
2. P-CSCF IP アドレスを確認
3. P-CSCF の設定を確認

配置

```
# P-CSCF 配置
pco: %{
  p_cscf_ipv4_address_list: ["10.0.0.50"] # 配置
}
```

## 配置MTU

配置

- 配置
- 配置
- 配置

配置

- MTU 配置
- MTU 配置

配置

```
# GTP 配置 1400
pco: %{
  ipv4_link_mtu_size: 1400
}

# 配置
pco: %{
  ipv4_link_mtu_size: 1360
}
```

---

# □□□□

## DNS □□

1. □□□□ **DNS** □□
  - □□□Google (8.8.8.8)□Cloudflare (1.1.1.1)
  - □□□□□□ DNS
2. □□□□□ **DNS**
  - □□□□
  - □□□□□
3. □□ **DNS** □□□
  - □□ DNSSEC □□□□
  - □□ DNS □□□□□□□□

## IMS □□

1. □□□□ **P-CSCF**
  - □□ 2 □□□□□□
  - □□□□□□□□□□
2. □□□□□
  - P-CSCF □□□□ UE IP □□□
  - □□ SIP □□□

## MTU □□

1. □□□□
  - GTP-U: 36 □□ (IPv4)
  - IPsec: □□ (50-100 □□)

## 2. **LTE MTU**

- **1400**
- 

## 3.

- **MTU**
- 

- **runtime.exs** - **UPF** **PCO**
- **UE IP** - IP **APN**
- **P-CSCF** - P-CSCF

- **PDN** - **PDN**
- **S5/S8** - GTP-C **PCO**
- **PFCP** -

## **IMS VoLTE**

- **Diameter Gx** - IMS
- **PCO**

**OmniPGW PCO** - *Omnitouch*

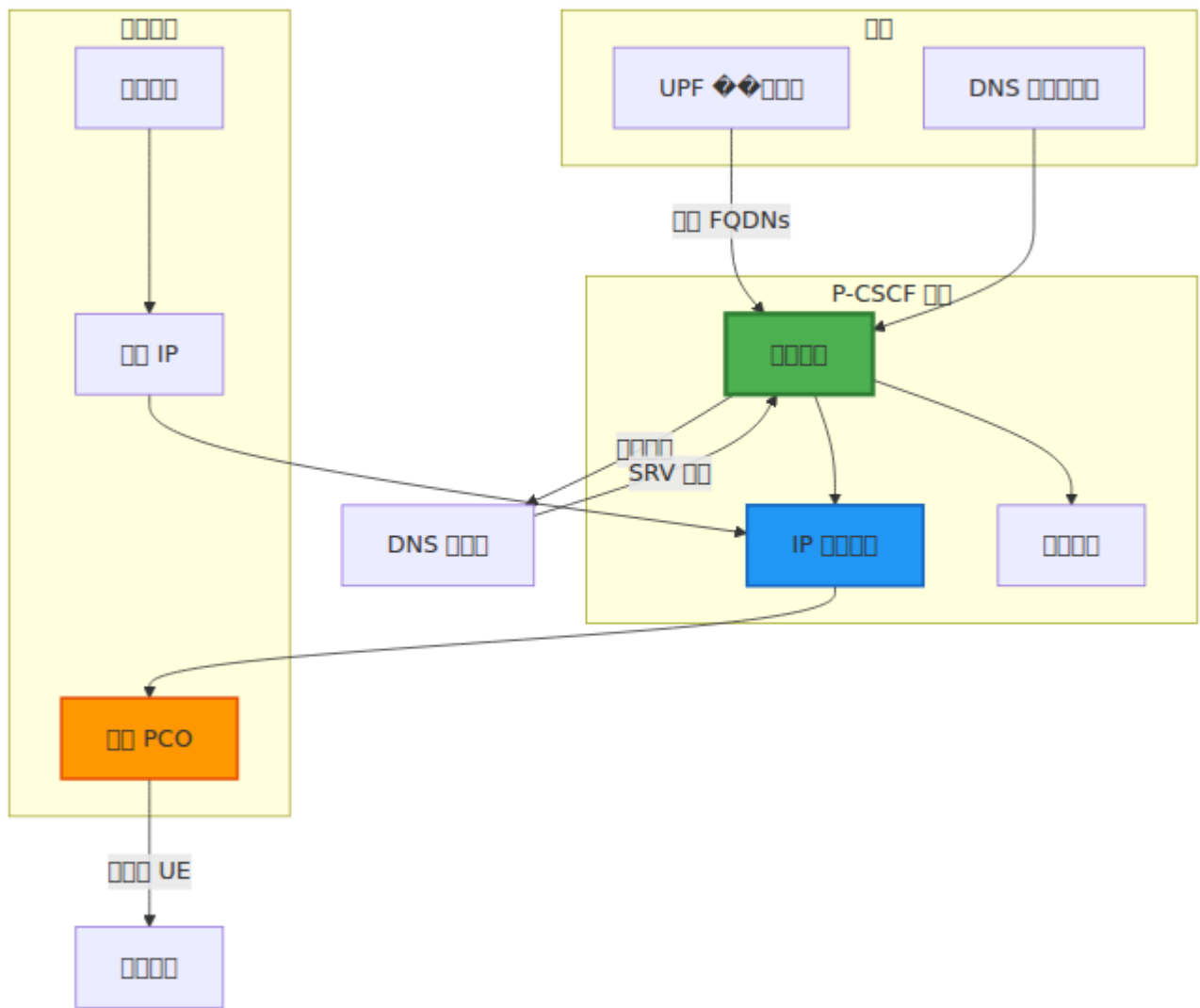
# P-CSCF

P-CSCF

OmniPGW Omnitouch

**P-CSCF** DNS SRV IMS P-CSCF SIP OPTIONS

- **P-CSCF** P-CSCF
- DNS 60
- **SIP OPTIONS** SIP OPTIONS ping P-CSCF
  - **TCP** TCP SIP OPTIONS
  - **UDP** TCP UDP
  - :up :down
- Web UI IP
- 
- **Prometheus** Prometheus



□□

1. □□□□
2. □□
3. □□□□
4. Web UI □□
5. □□□□□□□□
6. □□□□
7. DNS □□
8. □□□□
9. □□□□

□□□□

□□□□

```
# config/runtime.exs
```

```
# □□ PCO □□□□ P-CSCF □□ DNS □□□□
```

```
config :pgw_c,
```

```
  pco: %{
```

```
    p_cscf_discovery_dns_server: "10.179.2.177",
```

```
    p_cscf_discovery_enabled: true,
```

```
    p_cscf_discovery_timeout_ms: 5000
```

```
  },
```

```
  upf_selection: %{
```

```
    rules: [
```

```
      # IMS □□ - □□ P-CSCF □□
```

```
      %{
```

```
        name: "IMS □□",
```

```
        priority: 20,
```

```
        match_field: :apn,
```

```
        match_regex: "^ims",
```

```
        upf_pool: [
```

```
          %{remote_ip_address: "10.100.2.21", remote_port: 8805,
```

```
weight: 80}
```

```
        ],
```

```
      # P-CSCF □□ FQDN□□□□□ UPF □□□□□□□□□□□□
```

```
      p_cscf_discovery_fqdn:
```

```
"pcscf.mnc380.mcc313.3gppnetwork.org",
```

```
      # □□□□□□□□ PCO □□□□□
```

```
      pco: %{
```

```
        p_cscf_ipv4_address_list: ["10.101.2.100",
```

```
"10.101.2.101"]
```

```
      }
```

```
    }
```

```
  ]
```

```
}
```

□□□ □□□□ □□□□□□ UPF □□□□□□□□□□ PCO □□ □□□□□ P-CSCF □□□□□

## 準備

1. 確認 OmniPGW
  2. 確認 **Web UI → P-CSCF** 欄 ([https://localhost:8086/pcscf\\_monitor](https://localhost:8086/pcscf_monitor))
  3. 確認 P-CSCF の IP
- 

## 設定

### 確認 P-CSCF の設定

確認 PCO の P-CSCF の DNS の設定

```
pco: %{  
  # P-CSCF の DNS の設定  
  p_cscf_discovery_dns_server: "10.179.2.177",  
  
  # P-CSCF DNS の設定  
  p_cscf_discovery_enabled: true,  
  
  # DNS SRV の設定  
  p_cscf_discovery_timeout_ms: 5000,  
  
  # P-CSCF の IP の設定  
  p_cscf_ipv4_address_list: ["10.101.2.146"]  
}
```

### 確認 P-CSCF FQDNs

確認 UPF の P-CSCF の FQDN

```

upf_selection: %{
  rules: [
    # IMS - IMS P-CSCF
    %{
      name: "IMS",
      match_field: :apn,
      match_regex: "^ims",
      upf_pool: [...],
      p_cscf_discovery_fqdn:
"pcscf.ims.mnc380.mcc313.3gppnetwork.org",
      pco: %{
        p_cscf_ipv4_address_list: ["10.101.2.100"] #
      }
    },

    # - P-CSCF
    %{
      name: "",
      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"] #
      }
    },

    # - P-CSCF
    %{
      name: "",
      match_field: :apn,
      match_regex: "^internet",
      upf_pool: [...]
      # p_cscf_discovery_fqdn - PCO
    }
  ]
}

```

# 目標

## 前提

### 1. 確認

- P-CSCF 側 GenServer 確認
- 確認する UPF 確認する P-CSCF FQDNs

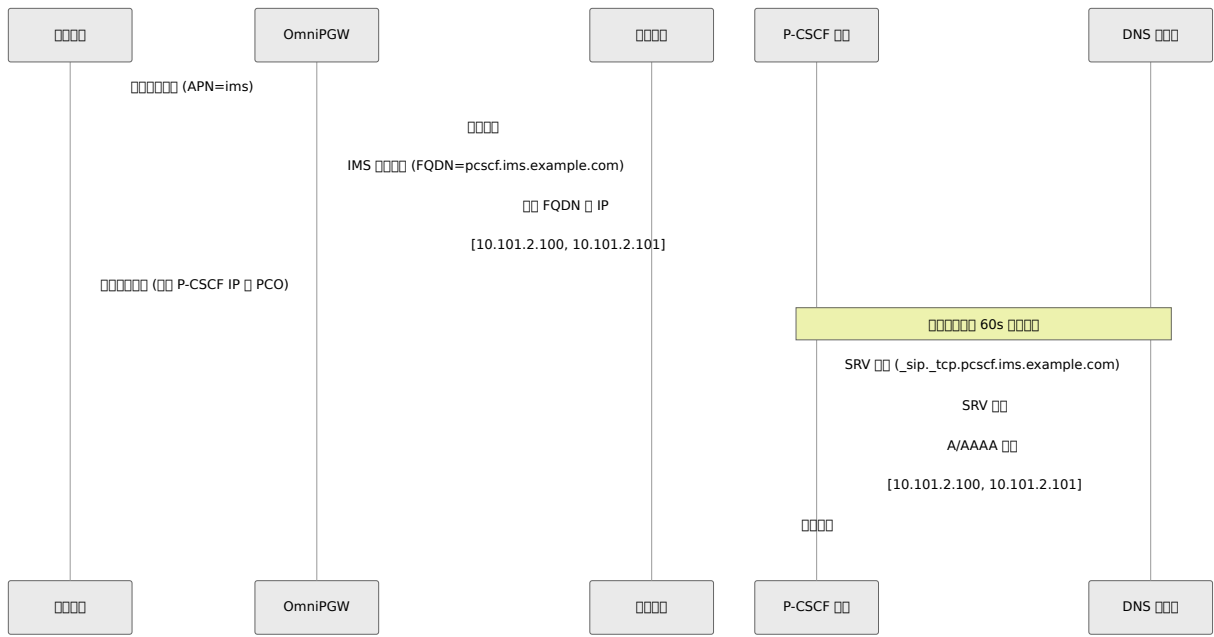
### 2. FQDN 設定

- 確認する FQDN 確認する
- 確認する FQDN 確認 DNS SRV 設定
- **SIP OPTIONS** 確認する
  - 確認 TCP SIP/2.0/TCP 5060 確認
  - 確認 TCP 確認 UDP SIP/2.0/UDP 5060 確認
  - 確認 :up 確認 :down 確認/確認
- 確認 IP 確認

### 3. 確認する 60 確認

- 確認する FQDNs
- DNS 確認する
- 確認する
  - 確認 TCP 確認 SIP OPTIONS 5 確認
  - 確認 TCP 確認 UDP 5 確認
  - 確認
- 確認する DNS 確認

# 网络架构图



## DNS 终端

终端 DNS SRV 终端 终端 P-CSCF 终端

1. **SRV** 终端 `_sip._tcp.{fqdn}` 终端 SRV 终端
2. 终端终端终端终端终端
3. 终端终端 SRV 终端终端终端
4. 终端终端终端终端终端 IP 终端A/AAAA 终端
5. 终端终端终端 IP 终端终端

## P-CSCF 终端终端

终端终端终端 **FQDN** 终端 **PCO** 终端**FQDN** 终端

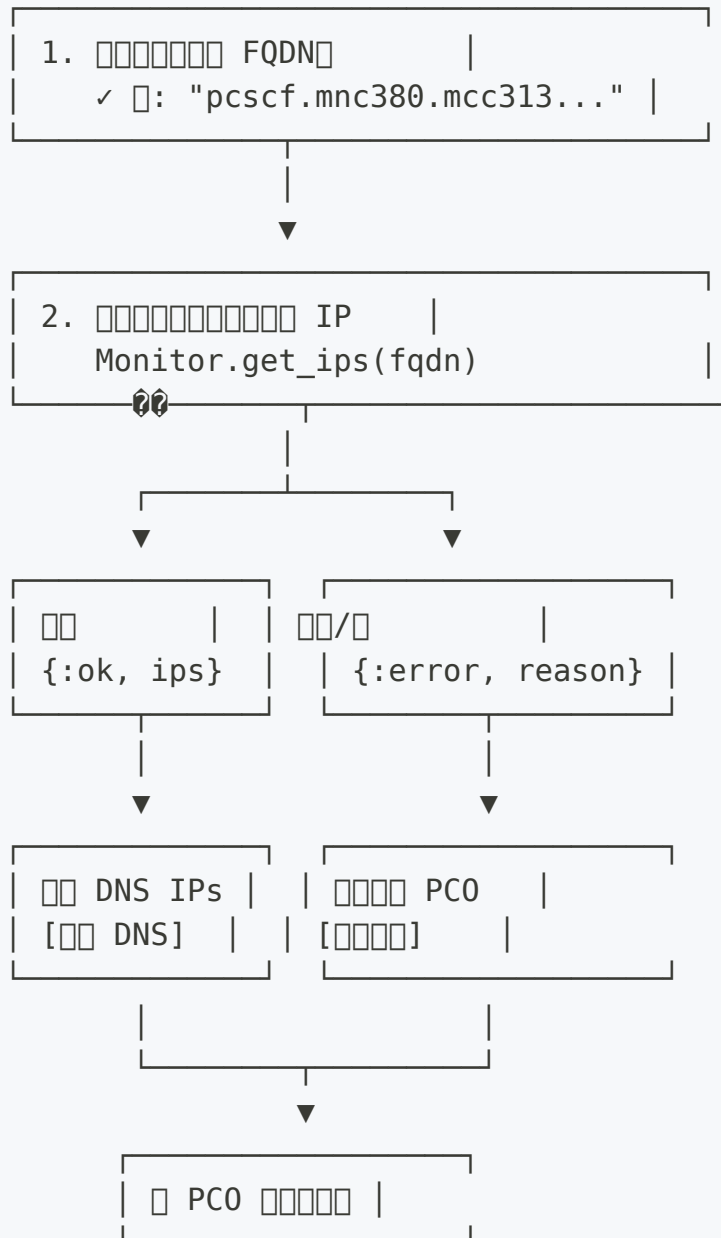
```
%{
  name: "IMS  ",
  p_cscf_discovery_fqdn: "pcscf.mnc380.mcc313.3gppnetwork.org",  #
←  ""
  pco: %{
    p_cscf_ipv4_address_list: ["10.101.2.100", "10.101.2.101"]  #
←  ""
  }
}
```

Table 1

| Field         | P-CSCF<br>Field       | IP                                        | Value                                    |
|---------------|-----------------------|-------------------------------------------|------------------------------------------|
| FQDN<br>Field | DNS<br>Field          | DNS<br>IP                                 | "FQDN<br>pcscf.example.com<br>P-CSCF IP" |
| FQDN<br>Field | PCO<br>Field          | PCO<br>pco.p_cscf_ipv4_address_list<br>IP | "FQDN P-CSCF<br>IPs...,<br>"             |
| FQDN<br>Field | PCO<br>Field          | PCO<br>pco.p_cscf_ipv4_address_list<br>IP |                                          |
| Field         | PCO<br>Field          | PCO<br>pco.p_cscf_ipv4_address_list<br>IP |                                          |
| Field<br>FQDN | PCO<br>Field<br>Field | Field IP                                  |                                          |

Table 2

## IMS 鉴权流程



鉴权

1. DNS 鉴权

```

p_cscf_discovery_fqdn: "pcscf.ims.example.com"
pco.p_cscf_ipv4_address_list: ["10.101.2.100"]

```

```
DNS 00: [10.101.2.150, 10.101.2.151]
UE 00: [10.101.2.150, 10.101.2.151] ← 00 DNS
00: 0 DNS 000000 PC0 000
```

## 2 DNS 警告

```

p_cscf_discovery_fqdn: "pcscf.ims.example.com"
pco.p_cscf_ipv4_address_list: ["10.101.2.100"]

```

```
DNS 00: ERROR :no_naptr_records
UE 00: [10.101.2.100] ← 0000 PC0
00: 00 DNS 0000000000
```

### 3 FQDN

```
##:
# ## p_cscf_discovery_fqdn
pco.p_cscf_ipv4_address_list: ["192.168.1.50"]
```

```
UE IP: [192.168.1.50] ← PC0
IP: 192 DNS IP
```

100

1. ☐ DNS ☐
2. ☐ DNS ☐
3. ☐
4. ☐

□□□□□□□□□□ FQDN □□□ PCO□

```
# ✓ 00: 000000
%{
  p_cscf_discovery_fqdn: "pcscf.ims.example.com", # 00
  pco: %{
    p_cscf_ipv4_address_list: ["10.101.2.100"] # 000
  }
}

# △ 00: 0000000000 PC00
%{
  p_cscf_discovery_fqdn: "pcscf.ims.example.com"
  # 0000000000
}

# ✓ 00: 000000 DNS 000
%{
  pco: %{
    p_cscf_ipv4_address_list: ["192.168.1.50"]
  }
}
```

---

## Web UI 00

### P-CSCF 0000

00000000 [https://localhost:8086/pcscf\\_monitor](https://localhost:8086/pcscf_monitor)

111

- 四四四
- 四四四 FQDNs
  - 四四四四 FQDNs
  - 四四四四四
  - 四四 P-CSCF IP 四
- **FQDN** 四
- 四四四四 FQDN
  - 四四四四 ✓ 四 / ✗ 四 / 四 四四
  - 四四 IP 四
  - 四四 IP 四四四四四四四四四四四四四四四四
  - 四四四四四
  - 四 FQDN 四四四四四
  - 四四四四四四四四四四四四
    - IP 四四四四
    - 四四四四 DNS SRV 四
    - 四四四四四四四 ✓ 四 / ✗ 四四

- 設定

- 設定可能な FQDNs
- **FQDN** 設定可能な FQDN
- 設定可能な 5 設定

- 設定

- **FQDNs** 設定可能な FQDN 設定
- 設定可能な DNS 設定可能な FQDNs
- **DNS** 設定可能な FQDNs
- **P-CSCF** 設定可能な FQDNs 設定可能な
- ✓ **SIP OPTIONS** 設定可能な SIP OPTIONS 設定可能な
- X **SIP OPTIONS** 設定可能な SIP OPTIONS 設定可能な
- **DNS** 設定可能な DNS 設定可能な
- 設定可能な SIP OPTIONS 設定可能な 60s 5s 設定可能な

設定可能な DNS 設定可能な P-CSCF 設定可能な SIP OPTIONS

## UPF 設定

UPF 設定 (/upf\_selection) 設定可能な P-CSCF 設定可能な

```
IMS (20)
  APN: ^ims
  UPF-IMS-Primary (10.100.2.21:8805)

P-CSCF
  FQDN: pcscf.mnc380.mcc313.3gppnetwork.org
  : ✓ (2 IP)
  IPs: 10.101.2.100, 10.101.2.101

PCO
  DNS: 10.103.2.195
  P-CSCF: 10.101.2.100, 10.101.2.101
```

## Prometheus

P-CSCF Prometheus 42069

## Gauge

```
# FQDN
pcscf_fqdns_total          # FQDN
pcscf_fqdns_resolved      # FQDNs DNS
pcscf_fqdns_failed        # FQDNs DNS

# 
pcscf_servers_total       # DNS SRV P-CSCF
pcscf_servers_healthy     # SIP OPTIONS
pcscf_servers_unhealthy   # SIP OPTIONS

# FQDN
pcscf_servers_healthy{fqdn="..."} # FQDN
pcscf_servers_unhealthy{fqdn="..."} # FQDN
```

- **healthy**: 成功した SIP OPTIONS ping (TCP 或 UDP)
- **unhealthy**: 失敗した SIP OPTIONS (5s 未満)

まとめ

## DNS 監視

```
# 成功した FQDNs
pcscf_fqdns_resolved

# DNS 成功率
(pcscf_fqdns_resolved / pcscf_fqdns_total) * 100

# 総サーバー数
pcscf_servers_total
```

## SIP OPTIONS 監視

```
# 成功した FQDNs
pcscf_servers_healthy

# 失敗した FQDNs
pcscf_servers_unhealthy

# SIP OPTIONS 成功率
(pcscf_servers_healthy / pcscf_servers_total) * 100

# 特定の FQDN 監視
pcscf_servers_healthy{fqdn="pcscf.mnc380.mcc313.3gppnetwork.org"}

# 監視対象サーバーが 0 未満の場合
pcscf_servers_healthy == 0 AND pcscf_servers_total > 0
```

## Prometheus 監視

```

# [] P-CSCF []
- alert: AllPCSCFServersDown
  expr: pcscf_servers_healthy == 0 AND pcscf_servers_total > 0
  for: 5m
  labels:
    severity: critical
  annotations:
    summary: "[] P-CSCF []"
    description: "{{ $value }} [] (0) - [] SIP OPTIONS []"

# [] 50% []
- alert: MajorityPCSCFServersDown
  expr: (pcscf_servers_healthy / pcscf_servers_total) < 0.5
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "[] P-CSCF []"
    description: "[] {{ $value }}% [] SIP OPTIONS"

# [] DNS []
- alert: PCSCFDNSResolutionFailed
  expr: pcscf_fqdns_failed > 0
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "P-CSCF DNS []"
    description: "{{ $value }} [] FQDN []"

```

[][][]

[][][][][]

```
[info] P-CSCF [] []
[info] [] 2 [] [] P-CSCF FQDNs [] []: ["pcscf.ims.example.com",
"pcscf.enterprise.example.com"]
[info] P-CSCF []: [] FQDN pcscf.ims.example.com
[debug] P-CSCF []: [] pcscf.ims.example.com [] 2 [] IP
[warning] P-CSCF []: [] pcscf.enterprise.example.com: :nxdomain
[debug] [] FQDN pcscf.ims.example.com [] P-CSCF []: [{10, 101,
2, 100}, {10, 101, 2, 101}]
```

## 配置

配置 DNS 和 PCO 配置

### 1 DNS 配置

```
p_cscf_discovery_fqdn: "pcscf.ims.example.com"
```

- 配置 DNS 配置 IP
- 配置 DNS 配置 IP
- 配置 DNS 配置 IP

### 2 配置 PCO

```
pco: %{
  p_cscf_ipv4_address_list: ["10.101.2.100", "10.101.2.101"]
}
```

- 配置 DNS 配置 IP
- 配置 DNS 配置 IP
- 配置 DNS 配置 IP

### 3. PC0 設定

```
# PC0 設定
pco: %{
  p_cscf_ipv4_address_list: ["10.101.2.146"]
}
```

- DNS 設定
- P-CSCF 設定
- 接続設定

#### DNS 設定

設定 "IMS 設定" の DNS

1. DNS 設定 "pcscf.ims.example.com"
  - └ IP → IP [10.101.2.100, 10.101.2.101] ✓
  - └ IP → IP
2. PC0 設定
  - └ IP → IP [10.101.2.100, 10.101.2.101] ✓
  - └ IP → IP
3. PC0 設定
  - └ IP [10.101.2.146] ✓ (設定)

## DNS 設定

### DNS 設定

DNS 設定 SRV 型 A/AAAA 型 P-CSCF 設定

```

; P-CSCF SRV _sip._tcp
_sip._tcp.pcscf.mnc380.mcc313.3gppnetwork.org. IN SRV 10 50 5060
pcscf1.example.com.
_sip._tcp.pcscf.mnc380.mcc313.3gppnetwork.org. IN SRV 20 50 5060
pcscf2.example.com.

; A
pcscf1.example.com. IN A 10.101.2.100
pcscf2.example.com. IN A 10.101.2.101

```

OmniPGW FQDN `_sip._tcp.` `p_cscf_discovery_fqdn:`  
`"pcscf.mnc380.mcc313.3gppnetwork.org"`  
`_sip._tcp.pcscf.mnc380.mcc313.3gppnetwork.org`

## SRV

SRV

```

_service._proto.domain. IN SRV priority weight port target.

```

- 10 ~ 20
- =
- SIP TCP 5060 UDP 5060
- IP

## ❏❏ DNS ❏❏

```
# ❏❏ SRV ❏❏❏❏❏ _sip._tcp ❏❏❏
dig SRV _sip._tcp.pcscf.mnc380.mcc313.3gppnetwork.org
@10.179.2.177

# ❏❏❏❏❏
# _sip._tcp.pcscf.mnc380.mcc313.3gppnetwork.org. 300 IN SRV 10 50
5060 pcscf1.example.com.

# ❏❏❏ P-CSCF ❏❏❏❏❏ IP
dig A pcscf1.example.com @10.179.2.177

# ❏❏❏❏❏
# pcscf1.example.com. 300 IN A 10.101.2.100
```

## ❏❏❏❏❏

### ❏❏❏FQDN ❏❏“❏❏”❏❏

❏❏❏

- Web UI ❏❏ X ❏❏❏❏
- ❏❏❏ :nxdomain❏:timeout❏ :no\_naptr\_records

❏❏❏❏❏

1. DNS ❏❏❏❏❏❏
2. FQDN ❏ DNS ❏❏❏❏
3. ❏❏❏ NAPTR ❏❏
4. DNS ❏❏❏❏❏

❏❏❏❏❏

```
# 1. Ping DNS Server
ping 10.179.2.177

# 2. Verify NAPTR Record
dig NAPTR pcscf.mnc380.mcc313.3gppnetwork.org @10.179.2.177

# 3. Check OmniPGW Log
grep "P-CSCF" /var/log/pgw_c.log

# 4. Check Configuration
grep "p_cscf_discovery_dns_server" config/runtime.exs

# 5. Check Web UI Configuration
# Verify FQDN is "pcscf1"

```

## Verify IP

Check

- Web UI shows "0 IP"
- Configuration ✓, Log ✗

Steps

1. NAPTR record FQDN is correct
2. Verify IMS/SIP URI
3. Check A/AAAA record

Commands

```
# Verify NAPTR Record
dig NAPTR pcscf.example.com @10.179.2.177

# Check SIP URI
# SIP: "SIP+D2U", "x-3gpp-ims:sip"
# HTTP: "HTTP", "FTP"

# Check A/AAAA Record
dig pcscf1.example.com A @10.179.2.177

```

## 如何配置 P-CSCF

如何

- UE 如何配置 P-CSCF
- 如何配置 P-CSCF IP

如何

1. DNS 如何配置
2. 如何配置
3. FQDN 如何

如何

```
# 1. 如何 P-CSCF 如何
# 如何 FQDN 如何

# 2. 如何
grep "如何 FQDN 如何 P-CSCF 如何" /var/log/pgw_c.log

# 3. 如何 UPF 如何
# 如何 FQDN 如何

# 4. 如何
# 如何 APN 如何
```

## 如何配置 DNS 如何

如何

- 如何
- 如何 `pcscf_discovery_query_duration_seconds`

如何

1. DNS 如何
2. 如何 DNS 如何
3. 如何

配置

```
# 配置
pco: %{
  p_cscf_discovery_timeout_ms: 2000 # 5000ms
}

# 配置 DNS
pco: %{
  p_cscf_discovery_dns_server: "10.0.0.10" # DNS
}
```

配置

## 1. DNS 配置

配置 DNS 配置

```
pco: %{
  # 配置 P-CSCF DNS UE DNS
  p_cscf_discovery_dns_server: "10.179.2.177",

  # UE DNS 配置
  primary_dns_server_address: "8.8.8.8",
  secondary_dns_server_address: "8.8.4.4"
}
```

配置

- 配置 UE DNS 配置 IMS DNS
- 配置 DNS
- 配置 DNS

## 2. 配置 P-CSCF 地址

```
%{
  p_cscf_discovery_fqdn: "pcscf.ims.example.com", # P-CSCF 地址
  pco: %{
    p_cscf_ipv4_address_list: ["10.101.2.100"] # P-CSCF 地址列表
  }
}
```

配置项

- P-CSCF 地址 DNS 配置项
- P-CSCF 地址列表
- P-CSCF SLA 配置

## 3. 配置 P-CSCF 地址 FQDN

```
rules: [
  # IMS
  %{
    name: "IMS",
    match_regex: "^ims",
    p_cscf_discovery_fqdn:
      "pcscf.ims.mnc380.mcc313.3gppnetwork.org"
  },

  # P-CSCF
  %{
    name: "P-CSCF",
    match_regex: "^enterprise",
    p_cscf_discovery_fqdn: "pcscf.enterprise.example.com"
  }
]
```

配置项

- P-CSCF 地址 FQDN 配置项
- P-CSCF 地址列表

- 配置

## 4. 配置 DNS 告警

```
# 配置 P-CSCF 告警规则
alert: HighPCSCFQueryLatency
expr: histogram_quantile(0.95,
pcscf_discovery_query_duration_seconds_bucket) > 2
for: 5m
labels:
  severity: warning
annotations:
  summary: "P-CSCF DNS 异常 (p95 > 2s)"
```

## 5. 配置 DNS 监控

- **Web UI** 配置 P-CSCF 监控
- 配置 `pcscf_monitor_fqdns_failed` 规则
- 配置 DNS 规则
- 配置 DNS 告警规则

## 6. 配置规则

```
# 配置 P-CSCF 规则
pco: %{
  p_cscf_discovery_timeout_ms: 5000 # 5 秒
}

# 配置 DNS 规则
pco: %{
  p_cscf_discovery_timeout_ms: 2000 # 2 秒
}
```

## 7. 配置 DNS 规则

配置 DNS 规则 DNS

```
# 1 P-CSCF DNS
pcscf.mnc380.mcc313.3gppnetwork.org. IN NAPTR 10 50 "s" "SIP+D2U"
"" _sip._udp.pcscf1.example.com.

# 2 P-CSCF DNS
pcscf.mnc380.mcc313.3gppnetwork.org. IN NAPTR 20 50 "s" "SIP+D2U"
"" _sip._udp.pcscf2.example.com.
```

## 配置

- **PCO** 配置 - 配置 DNS 到 P-CSCF 配置
- **配置** - 配置 OmniPGW 配置
- **配置** - 配置
- **配置** - 配置 PCO 配置
- **PFCP** 配置 - 配置

## 配置

**OmniPGW P-CSCF** 配置 - *OmniTouch* 配置

# PFCP/Sxb

PGW-C - PGW-U

## 

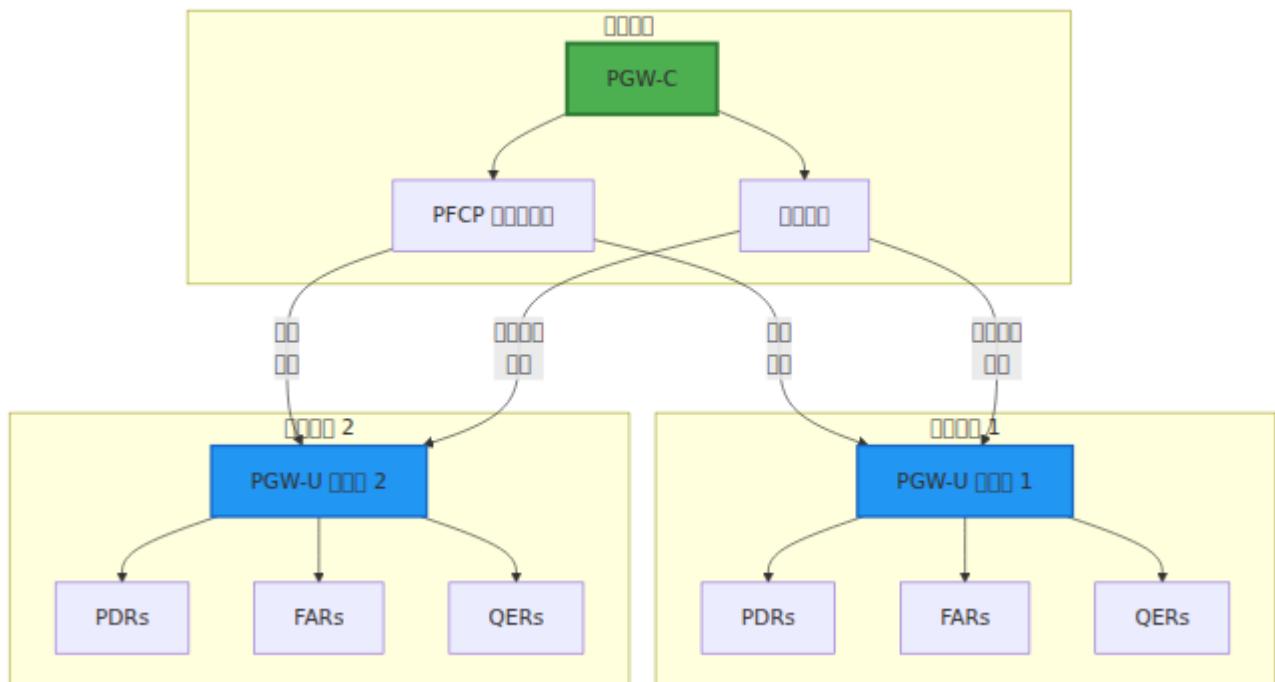
- 
- 
- PFCP
- PFCP
- 
- 
- DNS UPF
- 
- 
- Web UI - PFCP
- 

## 

Sxb PFCP (PGW-C) PGW-U

- (PGW-C)
- (PGW-U)

## PFCP



PFCP

## PFCP

PGW-C PFCP 1 (3GPP TS 29.244)

PFCP

- UDP
- 8805
- PFCP

ID

PFCP ID

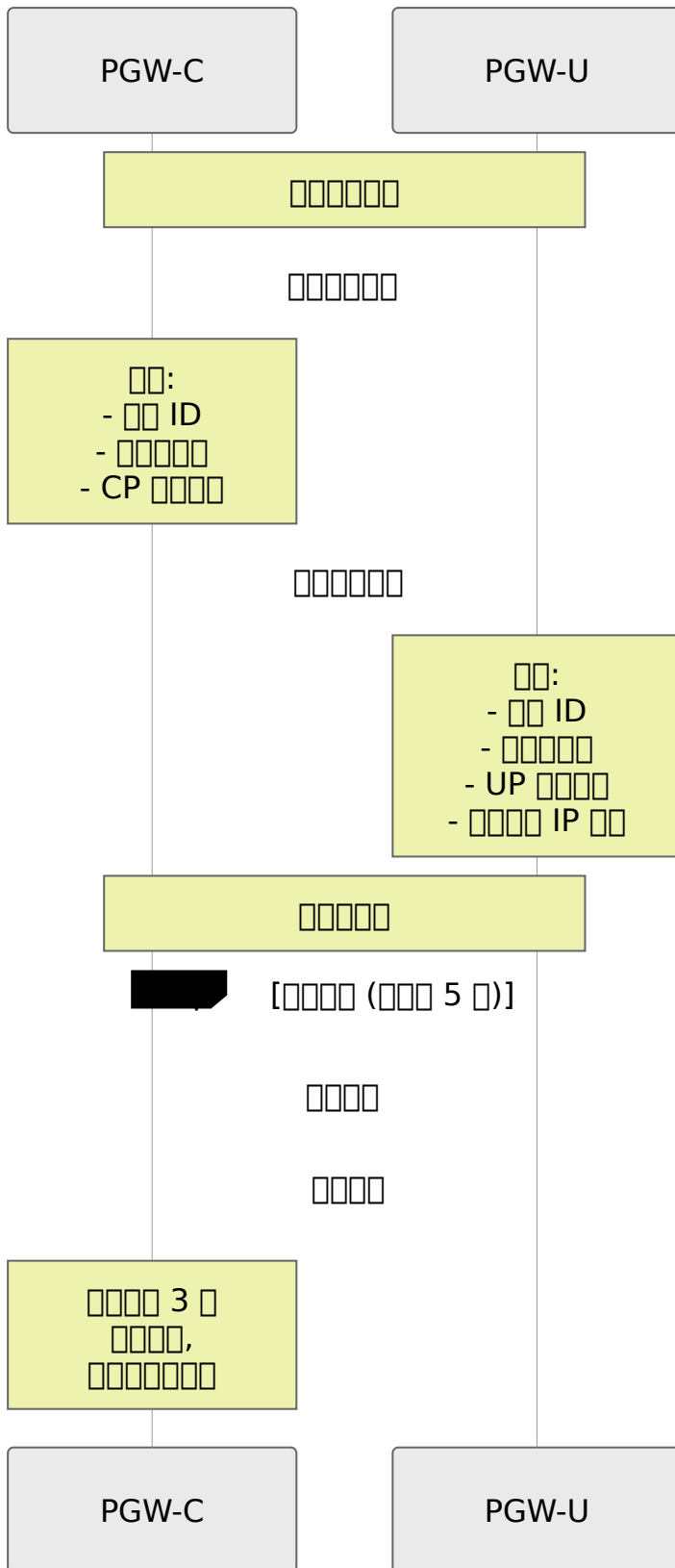
- IPv4 -
- IPv6

- **FQDN** (□□□□□□)
- 

# PFCP □□□□

□□□□□□□□□□ PGW-C □ PGW-U □□□□ PFCP □□□

□□□□□□



节点信息

节点 PFCP 配置信息

| 属性                            | 说明                  |
|-------------------------------|---------------------|
| is_associated                 | 是否关联                |
| remote_node_id                | 远端节点 ID (IP 或 FQDN) |
| remote_ip_address             | 远端节点 IP 地址          |
| remote_port                   | UDP 端口 (默认 8805)    |
| heartbeat_period_ms           | 心跳周期 (ms)           |
| missed_heartbeats_consecutive | 连续丢失心跳次数            |
| up_function_features          | 支持的 UP 功能特性         |
| up_recovery_time_stamp        | UP 恢复时间戳            |

节点 ID

节点 ID 配置信息

节点 ID

```
# 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}
  ]
}
# UPF 5
```

UPF

- missed\_heartbeats\_consecutive
- 3
- 

## PFCP

PFCP UE PDN

```
sequenceDiagram
    participant PGW_C as PGW-C
    participant PGW_U as PGW-U
    PGW_C->>PGW_U: OMNISCAN
    PGW_C->>PGW_U: OMNISCAN-REQ
    PGW_U-->>PGW_C: OMNISCAN-RES
    PGW_C->>PGW_U: OMNISCAN-CONF
    PGW_C-->>PGW_U: loop [OMNISCAN (max 5 times)]
    PGW_C->>PGW_U: OMNISCAN
    PGW_U-->>PGW_C: OMNISCAN
    PGW_C->>PGW_U: OMNISCAN-CONF-3
    PGW_C->>PGW_U: OMNISCAN-CONF-3, OMNISCAN-CONF-3
```

## PGW-C

UE PDN

## PGW-C PGW-U

PGW-C

- **SEID** (ID) -
- **ID** - PGW-C ID
- **F-SEID** - SEID (IP + SEID)
- **PDRs** - (2 + )
- **FARs** - (2 + )
- **QERs** - QoS ( )
- **BAR** - ( )

## PGW-U

PGW-U

- -
- **F-SEID** - PGW-U
- **PDRs** -
- **F-TEID** - S5/S8 TEID

## QoS

QoS

QoS

- PDRsFARsQERs
- 
-

|  |  |  |  |
|--|--|--|--|
|  |  |  |  |
|--|--|--|--|

UE PDN

111

1. PGW-C 0000000000 SEID
2. PGW-U 0000000000
3. PGW-U 00000000

**F-TEID** ☐ ☐

**F-TEID (FQDN)** 000000000000 GTP-U 00000000 PFCP 00000000 F-TEID 00000000  
00 UPF 000000000000000000000000

**□□ F-TEID □□**

□□□□□ F-TEID □□□□□

- **TEID** (識別子) - 32 ビット
- **IP アドレス** - GTP-U 識別子 (UPF の IP アドレス)

□□□ □□□ TEID □□

**1UPF (100%)**

- PGW-C “TEID” CHOOSE
- UPF TEID

## 2 PGW-C ( )

- PGW-C 0000 TEID 0000 UPF“00000000 TEID”
- UPF 00000000 TEID00000000

**UPF**  (  -  )

111

```
sxb: %{
    allocate_uplink_f_teid: false # []
}
```

555

1. PGW-C → F-TEID → CHOOSE → PCF → PCF →
2. UPF → TEID
3. UPF → F-TEID (TEID + IP)
4. PGW-C → F-TEID

|  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|

□ □ □ □ □ □

- UPF 1000000 = UPF 1000000000
- PGW-C 10000 UPF 1000 TEID
- 100000000000000

PGW-C

- PGW-C 通過 UPF 通過
- PGW-C 通過 TEID 通過
- UPF 通過

□ □ **3GPP** □ □

- CHOOSE 3GPP TS 29.244
- UPF
- “”

|  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|

- PGW-C 向 UPF 索取 TEID
- 向 TEID 索取
- UPF 向 TEID

55

- 1 个 UPF 实例
- 1 个 UPF 实例 PGW-C 实例
- 1 个实例
- 1 个实例 3GPP PFCP 实例

实例

- △ 实例 UPF 实例 CHOOSE 实例
- △ 实例“实例 IE”实例 UPF 实例 CHOOSE

## PGW-C 实例 (实例)

实例

```
sxb: %{
  allocate_uplink_f_teid: true
}
```

实例

1. PGW-C 实例 TEID
2. PGW-C 实例 TEID 实例 PFCP 实例
3. UPF 实例 TEID实例
4. PGW-C 实例 UPF 实例 TEID 实例

实例

## 1 UPF 实例 CHOOSE

- 1 UPF 实例/实例
- UPF 实例 PFCP 实例 TEID
- 实例

## 1 实例 TEID 实例

- 实例 PGW-C 实例 TEID
- 实例 PGW-C 实例 TEID 实例
- 实例 TEID 实例

□ □□□□□

- □□□□□□□□ TEID □□□□
- □□□□□□□□□□□□□□ TEID □□

□□□□

△ □ **PGW-C** □□□□

- □□ UPF □□□ PGW-C □□□□□□ TEID □□
- □□□
  - □□ PGW-C □□□ TEID □□□**???**□□□□
  - □□ TEID □□□□□□□□□□□□
  - □□□□□□□□□□□□□□□□

△ □□□□

- PGW-C □□□□□□□□ TEID□□□□□□□□
- PGW-C □□□□ TEID □□□□□□□□□□□□□□□□
- □□□□□□□□□□□□

△ □□□□□□

- □□□□□□ PFCP □□□□□
- □□□□□□□□ CHOOSE □ UPF □□□□□□

□□□□□□

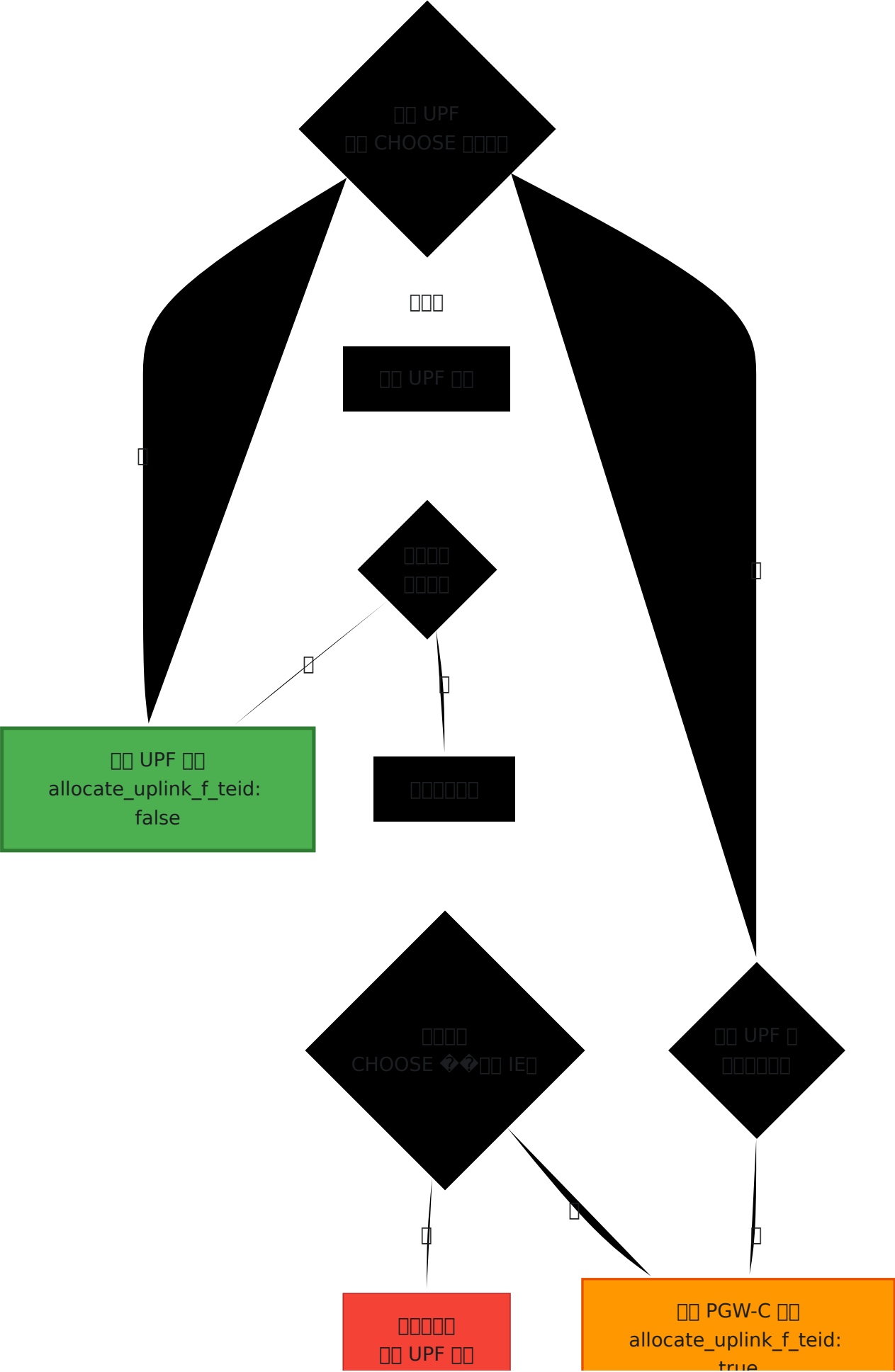
- △ □□ **UPF** □□□ **CHOOSE** □□□
- △ □□ UPF □□□□□□□□□□□□□□□□
- △ □□□□□□□□□□
- △ □□ PGW-C TEID □□□□□□□□□□

**TEID** □□□□□□ PGW-C □□□□□□□□□□□□□□□□

- TEID □□□1 □ 0xFFFFFFFF (42 □□□□)
- □□□□□□□□ 100 □□□□□□□□ 0.023%
- □□□□□□□□□□□□□□□□□□□□□□

- 000000000000 TEID

0000



\_\_\_\_\_

© 2006 The Authors  
Journal compilation © 2006 Blackwell Publishing Ltd

```
# 选择 强制
grep -i "choose\|mandatory.*missing" /var/log/pgw_c.log

# PFCP 会话建立
grep "Session Establishment Response" /var/log/pgw_c.log
```

- 网络侧“网络 IE”和“无线 IE”
- UPF 网络 F-TEID网络 CHOOSE
- 网络侧 网络 `allocate_uplink_f_teid: true`

- 000000 - TEID 00000040 0000
- 00 TEID 00000000000000000000

```
# 確認
grep "registered_teid_count" /var/log/pgw_c.log
```

```
# [[ config/runtime.exs
sxb: %{
  local_ip_address: "10.0.0.20",
  allocate_uplink_f_teid: false # [[ UPF [[ CHOOSE[[[[ true
}
```

PGW-C

```
systemctl restart pgw_c
```

確認する必要があるのは PFCP かどうか

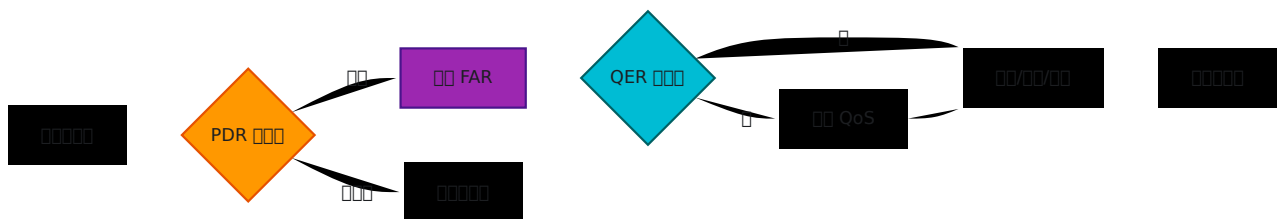
```
# PFCP 確認
tcpdump -i any -n port 8805 -w pfcp.pcap

# Wireshark で確認する
# F-TEID から "CHOOSE" まで UPF まで
# F-TEID から TEID まで PGW-C まで
```

確認する

PFCP 確認する必要がある

確認



## PDR (確認する)

確認する必要がある

PGW-C 確認

**PDR #1** - 確認

PDR ID: 1  
QoS: 100  
PDI (PDR ID):  
- QoS: CORE (PDR ID)  
- UE IP QoS: 100.64.1.42/32  
FAR ID: 1 (PDR ID)

## PDR #2 - QoS

PDR ID: 2  
QoS: 100  
PDI (PDR ID):  
- QoS: ACCESS (SGW ID)  
- F-TEID: <S5/S8 QoS ID>  
FAR ID: 2 (PDR ID)  
QER ID: 1 (QoS ID)

## QoS PDR ID

- **PDR ID** - PDR ID
- QoS - QoS
- **PDI** - PDI IP TEID ID
- QoS ID - QoS ID GTP-U ID
- **FAR ID** - FAR ID
- **QER ID** - QoS QoS ID

## FAR (PDR ID)

QoS PDR ID

## FAR #1 - QoS (QoS → UE)

FAR ID: 1

IP: 10

IP:

- IP: ACCESS (SGW)
- IP: GTP-U/UDP/IPv4
- F-TEID: <SGW S5/S8 IP>

## FAR #2 - IP (UE → IP)

FAR ID: 2

IP: 10

IP:

- IP: CORE (IP)
- (IP - IP IP)

## IP FAR

- **FAR ID** - IP
- IP - IP
- IP
  - IP (ACCESS/CORE)
  - IP (GTP-U IP)
  - IP (VRF/IP)

## QER (QoS)

IP QoS QER IP Diameter Gy IP

## IP QER

QER ID: 1  
 QER: 00  
 QER: 00000000  
 - QER: 100 Mbps  
 - QER: 50 Mbps  
 QER: (00000000 GBR 00)  
 - QER: 10 Mbps  
 - QER: 10 Mbps

QER QER

- QER ID - 00000000
- QER - 0000000000000000
- MBR - 00000000/0000
- GBR - 0000000000000000
- QCI - QoS 000000000000

BAR (00000000)

0000 00 UE 0000000000000000

BAR

BAR ID: 1  
 00000000: 100ms  
 00000000: 10

0000 0000 DRX (000000) 00

00

Sxb

00 config/runtime.exs

```

config :pgw_c,
  sxb: %{
    # PFCP IP
    local_ip_address: "10.0.0.20",

    # (8805)
    local_port: 8805,

    # F-TEID
    # false ( )UPF F-TEID (CH00SE )
    # truePGW-C F-TEID
    # UPF CH00SE
    allocate_uplink_f_teid: false
  },

  # UPF - UPF
  upf_selection: %{
    fallback_pool: [
      %{
        # PGW-U IP
        remote_ip_address: "10.0.0.21",

        # PFCP (8805)
        remote_port: 8805,

        # (100 = , 0 = )
        weight: 100
      }
    ]
  }
}

```

## PGW-U

```

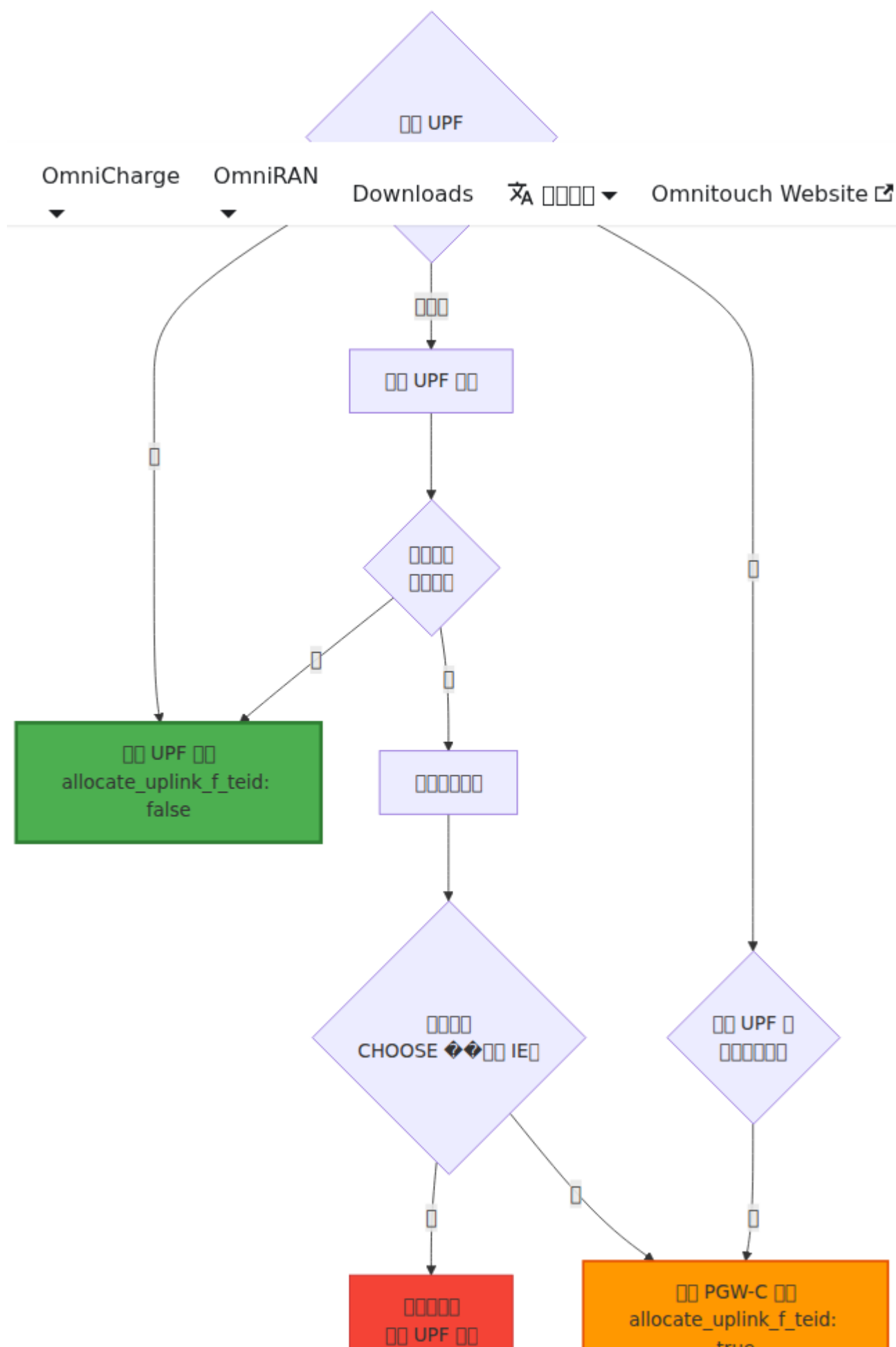
config :pgw_c,
  sxb: %{
    local_ip_address: "10.0.0.20"
  },
  upf_selection: %{
    fallback_pool: [
      %{remote_ip_address: "10.0.1.21", remote_port: 8805, weight:
50}, # 50% []
      %{remote_ip_address: "10.0.2.21", remote_port: 8805, weight:
50} # 50% []
    ]
  }
# [] UPF [] [] [] [] [] 5 []

```

## UPF [] [] [] []

PGW-C [] [] **UPF** [] [] [] [] [] [] [] [] [] []

1. [] [] ([]) - [] [] [] [] [] []
2. [] **DNS** [] [] ([]) - [] DNS NAPTR [] [] [] [] [] [] [] []
3. [] [] ([]) - [] [] [] [] [] [] [] [] UPF []



UPF



```

config :pgw_c,
  # PFCP []
  sxb: %{
    local_ip_address: "10.0.0.20"
  },

  # UPF [] UPF []
  upf_selection: %{
    #
=====

    # [] DNS [] ([])
    #
=====

    # [] (ULI) [] DNS
    # [] UPF []
    dns_enabled: false,
    dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],
    dns_suffix: "epc.3gppnetwork.org",
    dns_timeout_ms: 5000,

    #
=====

    # [] ([])
    #
=====

    # []
    # [] UPF []
    rules: [
      # [] 1[]IMS [] - []
      %{
        name: "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}
        ],
        # [] PCO []
        pco: %{
          p_cscf_ipv4_address_list: ["10.101.2.100", "10.101.2.101"]

```

```

    }
},

# 2G APN - 中国移动
%{
    name: "中国移动",
    priority: 15,
    match_field: :apn,
    match_regex: "^(enterprise|corporate)\.apn",
    upf_pool: [
        weight: 100}
        %{remote_ip_address: "10.100.3.21", remote_port: 8805,
    ],
    pco: %{
        primary_dns_server_address: "192.168.1.10",
        secondary_dns_server_address: "192.168.1.11",
        ipv4_link_mtu_size: 1500
    }
},

# 3G APN - 中国移动
%{
    name: "中国移动",
    priority: 10,
    match_field: :serving_network_plmn_id,
    match_regex: "^(310|311|312|313)", # 中国移动
    upf_pool: [
        weight: 100}
        %{remote_ip_address: "10.100.4.21", remote_port: 8805,
    ],
},

# 4G APN - 中国移动
%{
    name: "中国移动",
    priority: 5,
    match_field: :apn,
    match_regex: "^internet",
    upf_pool: [
        weight: 33}
        %{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}
    ]
  }
],

#
=====

#  ( )
#
=====

#  DNS 
fallback_pool: [
  %{remote_ip_address: "127.0.0.21", remote_port: 8805, weight:
100}
]
}
```

Parameters

| Parameter                | Unit            | Value              |
|--------------------------|-----------------|--------------------|
| :imsi                    | IMSI            | "310260123456789"  |
| :apn                     | APN             | "internet", "ims"  |
| :serving_network_plmn_id | PLMN (MCC+MNC)  | "310260" (MCC+MNC) |
| :sgw_ip_address          | SGW IP (IPv4)   | "10.0.1.50"        |
| :uli_tai_plmn_id         | PLMN ID         | "310260"           |
| :uli_ecgi_plmn_id        | E-UTRAN PLMN ID | "310260"           |

UPF Parameters

UPF Parameters

```
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}
]
```

計算方法

1. 50 + 30 + 20 = 100
2. 0.0 ~ 100.0
3. UPF
  - 0-50: UPF-1 (50%)
  - 50-80: UPF-2 (30%)
  - 80-100: UPF-3 (20%)

例

- (33, 33, 34)
- (80) (20)
- UPF

**PCO**

PCO ( )

```
%{
  name: "IMS []",
  match_field: :apn,
  match_regex: "^ims",
  upf_pool: [...],
  pco: %{
    # []
    p_cscf_ipv4_address_list: ["10.101.2.100", "10.101.2.101"],
    # [] PCO []
  }
}
```

## [] **PCO** []

- 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

## [] **DNS** []

[][]PGW-C [] DNS NAPTR []

```
upf_selection: %{
  dns_enabled: true,
  dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],
  dns_suffix: "epc.3gppnetwork.org",
  dns_timeout_ms: 5000
}
```

## [][]

1. **ECGI** (E-UTRAN []) - []
2. **TAI** ([]) - []
3. **RAI** ([]) - 3G/2G []
4. **SAI** ([]) - 3G []

## 5. CGI (E-UTRAN) - 2G

### DNS

```
# ECGI
eci-1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org

# TAI
tac-lb64.tac-hb00.tac.epc.mnc999.mcc999.epc.3gppnetwork.org
```

### DNS

- ECGI TAI
- DNS (DNS)
- UPF
- DNS DNS

DNS UPF

## DNS UPF

DNS UPF UE (ULI) DNS NAPTR

3GPP TS 23.003 - UPF DNS

- UPF
- 
- 
- UPF

□□□□

Parse error on line 25: ... style PGWC fill:#4CAF50,stroke:#2E7 -----  
--^ Expecting 'SOLID\_OPEN\_ARROW', 'DOTTED\_OPEN\_ARROW',  
'SOLID\_ARROW', 'BIDIRECTIONAL\_SOLID\_ARROW', 'DOTTED\_ARROW',  
'BIDIRECTIONAL\_DOTTED\_ARROW', 'SOLID\_CROSS', 'DOTTED\_CROSS',  
'SOLID\_POINT', 'DOTTED\_POINT', got 'TXT'

□□

□□

```
config :pgw_c,  
  upf_selection: %{  
    # □□□□ DNS □□□  
    dns_enabled: true,  
  
    # □□□□□□□□□□ ECGI□□□□ TAI□□□□ RAI □  
    dns_query_priority: [:ecgi, :tai, :rai, :sai, :cgi],  
  
    # □□□ DNS □□  
    dns_suffix: "epc.3gppnetwork.org",  
  
    # DNS □□□□  
    dns_timeout_ms: 5000,  
  
    # □□□□□□□□□□ DNS  
    rules: [...],  
  
    # □□ DNS □□□□□□  
    fallback_pool: [...]  
  }
```

## DNS □□□□

DNS □□□□□□ GTP-C □□□□□□□□□□ (ULI) □□□

### 1. ECGI (E-UTRAN □□□□□□□□)

ECI - LTE ECI

ECI

```
eci-<HEX-ECI>.ecgi.epc.mnc<MNC>.mcc<MCC>.<dns_suffix>
```

ECI

```
# ECI ID: 0x1A2B3C (1,715,004 ECI)
# PLMN: MCC=999, MNC=999
eci-1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org
```

ECI LTE (4G) ECI

## 2. TAI (TAI)

ECI - TAI

ECI

```
tac-lb<LB>.tac-hb<HB>.tac.epc.mnc<MNC>.mcc<MCC>.<dns_suffix>
```

ECI

```
# TAC: 0x0064 (100 TAC)
# LB: 0x64, HB: 0x00
tac-lb64.tac-hb00.tac.epc.mnc999.mcc999.epc.3gppnetwork.org
```

ECI LTE (4G) ECI

## 3. RAI (RAI)

3G/2G ECI

ECI

```
rac<RAC>.lac-lb<LB>.lac-hb<HB>.lac.raimnc<MNC>.mcc<MCC>.  
<dns_suffix>
```

例

```
# RAC: 0x0A (10 例)  
# LAC: 0x1234 (4660 例)  
rac0a.lac-lb34.lac-hb12.lac.raimnc999.mcc999.epc.3gppnetwork.org
```

3G/2G UMTS/GPRS

#### 4. SAI (例)

3G

例

```
sac<SAC>.lac-lb<LB>.lac-hb<HB>.lac.saimnc<MNC>.mcc<MCC>.  
<dns_suffix>
```

例

```
# SAC: 0x0001  
# LAC: 0x1234  
sac0001.lac-lb34.lac-hb12.lac.saimnc999.mcc999.epc.3gppnetwork.org
```

3G UMTS

#### 5. CGI (例)

2G

例

```
ci<CI>.lac-lb<LB>.lac-hb<HB>.lac.cgi.mnc<MNC>.mcc<MCC>.  
<dns_suffix>
```

□□□

```
# CI: 0x5678  
# LAC: 0x1234  
ci5678.lac-lb34.lac-hb12.lac.cgi.mnc999.mcc999.epc.3gppnetwork.org
```

□□□□□ 2G GSM □□

## DNS □□□□

### NAPTR □□□□□

DNS □□ NAPTR □□□□□ UPF IP □□□

```
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
```

### PGW-C □□□

1. □□ NAPTR □□□□□ UPF IP □□
2. □ DNS □□□□□□□□□□
3. □□□□□□□□□□ (□□□□□□□□□□)

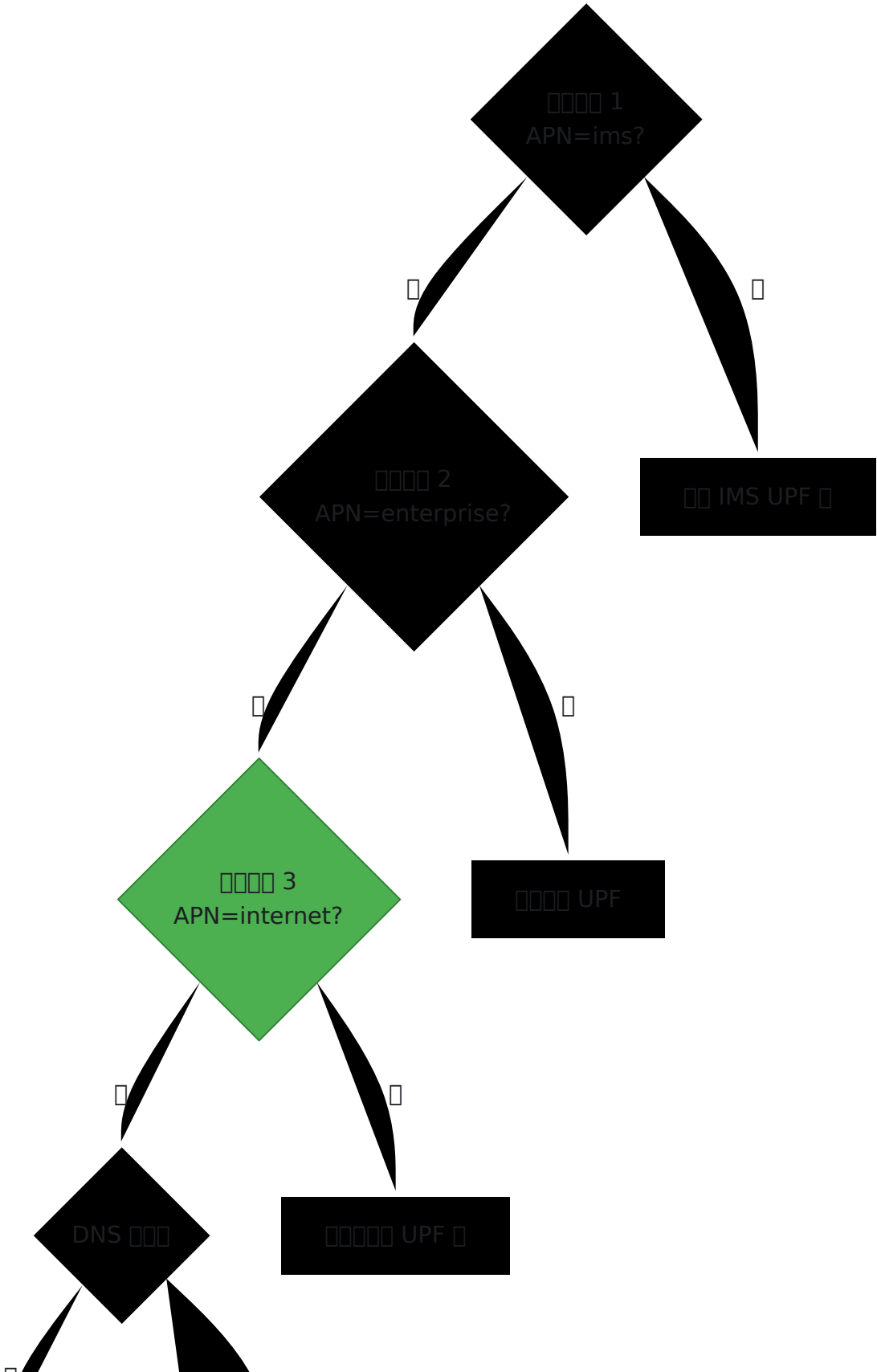
□□□

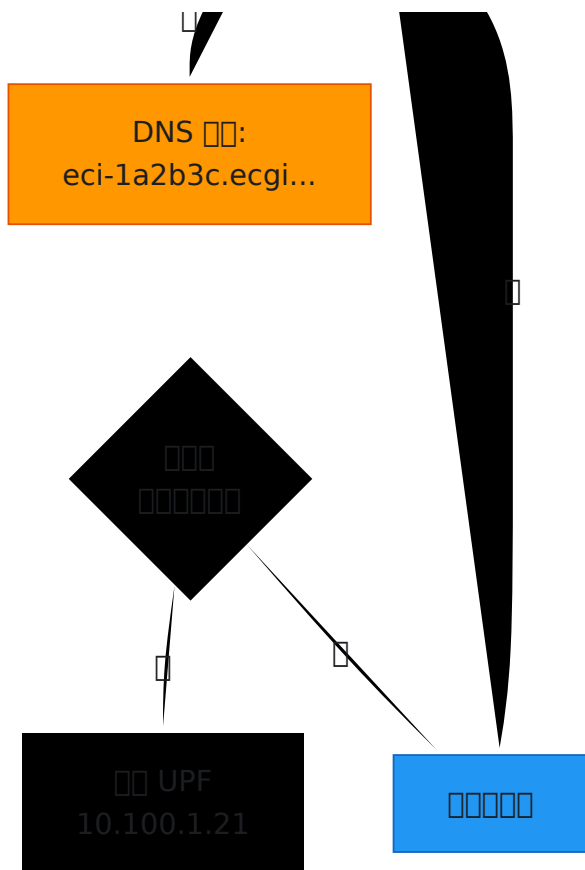
DNS □□: [10.100.1.21, 10.100.5.99]

□□: 10.100.1.21 (□□□□□□)  
□□: □□ upf\_selection □□□□□□□□

□□□□□□

例: APN=internet  
ECGI=0x1A2B3C





核心网

## 1. 核心网

核心网 核心网核心网 UPF

## DNS 解析

```
# 核心网
eci-aaa.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org → UPF-核心网
(10.1.1.21)

# 核心网
eci-bbb.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org → UPF-核心网
(10.2.1.21)

# 核心网
eci-ccc.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org → UPF-核心网
(10.3.1.21)
```

核心网 核心网核心网 UPF核心网核心网

## 2. DNS

MEC (UPF) UPF

### DNS

```
# UPF  
eci-*.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org → UPF
```

## 3. DNS

UPF

DNS PGW-C

### DNS

### DNS

- “DNS UPF :nxdomain”
- 🔍

1. DNS
2. DNS ID
3. GTP-C ULI

```
# 检查 DNS 配置
dig eci-1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org NAPTR

# 检查 PGW-C 是否 DNS 配置
grep "DNS UPF selection: querying" /var/log/pgw_c.log

# 检查 ULI 配置
# 检查 "uli" 配置
```

## DNS 配置 UPF

检查

- DNS 配置 `upf_selection` 配置 UPF
- 检查配置
- 检查 PFCP 配置
- 检查 PFCP 配置

检查

```
DNS 配置: [10.99.1.50]
upf_selection: [10.100.1.21, 10.100.1.22]

配置: 检查 10.99.1.50
- 检查 PFCP 配置
- 检查配置
- 检查配置
```

检查配置

1. 检查 `upf_selection` 配置

```
upf_selection: %{
  fallback_pool: [
    %{remote_ip_address: "10.99.1.50", remote_port: 8805, weight:
100}
  ]
}
```

2. DNS UPF IP

3. MEC/

- “DNS UPF ”
- 

```
upf_selection: %{  
  dns_timeout_ms: 10000 # 10  
}
```

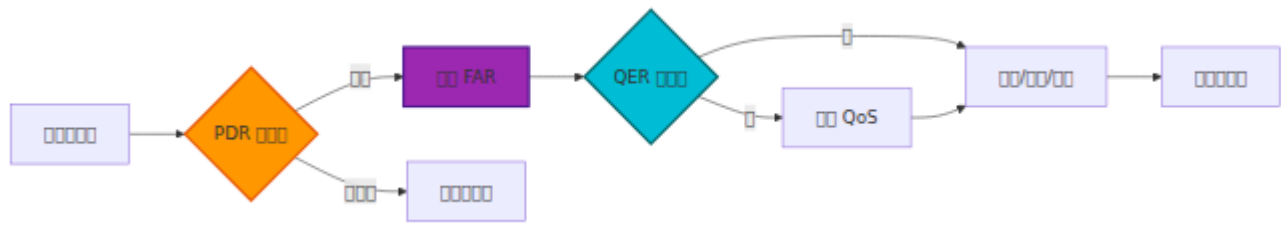
## DNS

```
# DNS  
rate(upf_selection_dns_success_total[5m]) /  
rate(upf_selection_dns_attempts_total[5m])  
  
# DNS  
histogram_quantile(0.95,  
rate(upf_selection_dns_duration_seconds_bucket[5m]))  
  
# DNS  
rate(upf_selection_fallback_used_total[5m])
```

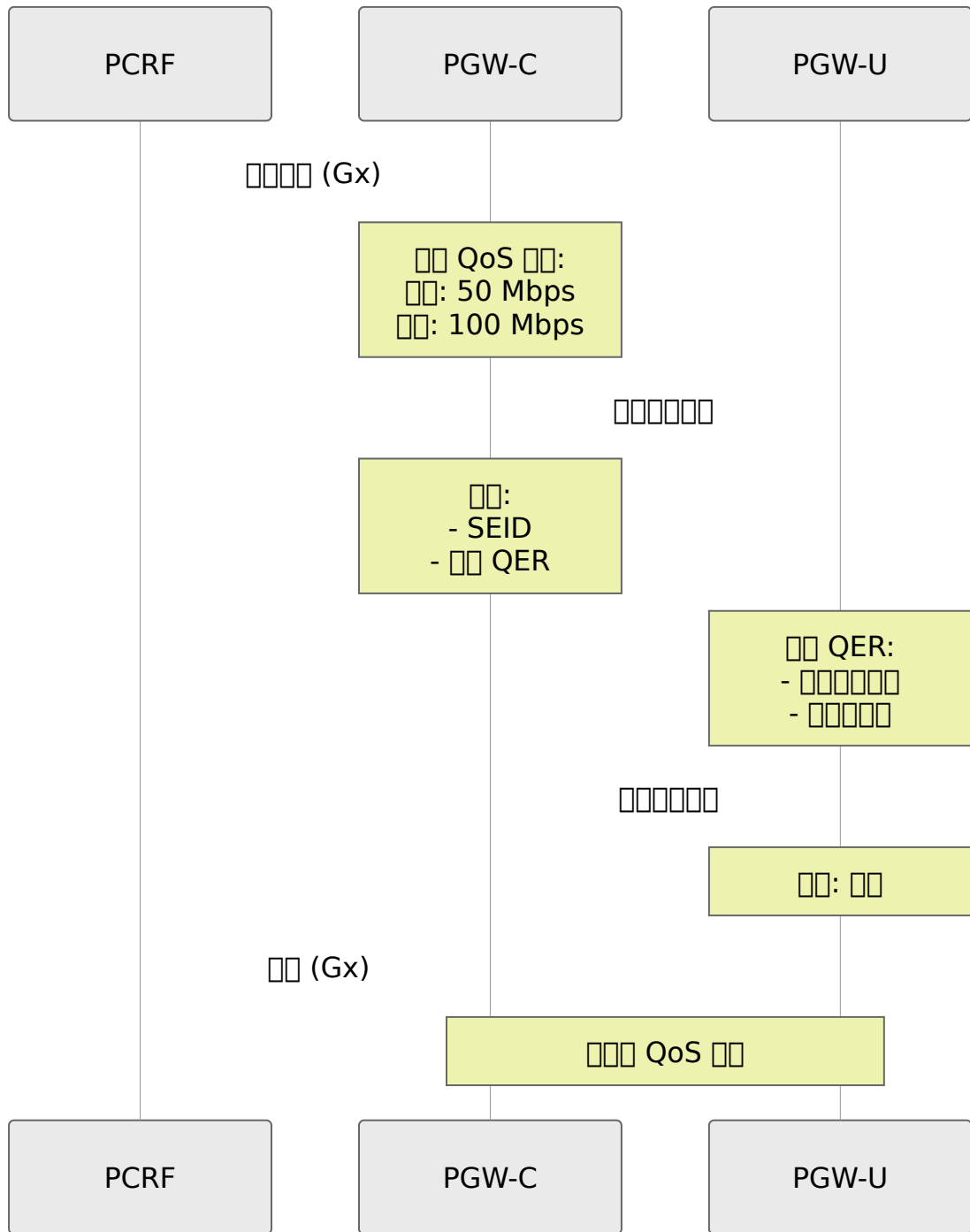
```
[debug] DNS UPF selection: querying eci-  
1a2b3c.ecgi.epc.mnc999.mcc999.epc.3gppnetwork.org  
[debug] DNS UPF selection: got 2 candidates from DNS  
[info] DNS UPF selection: selected 10.100.1.21
```

□□□

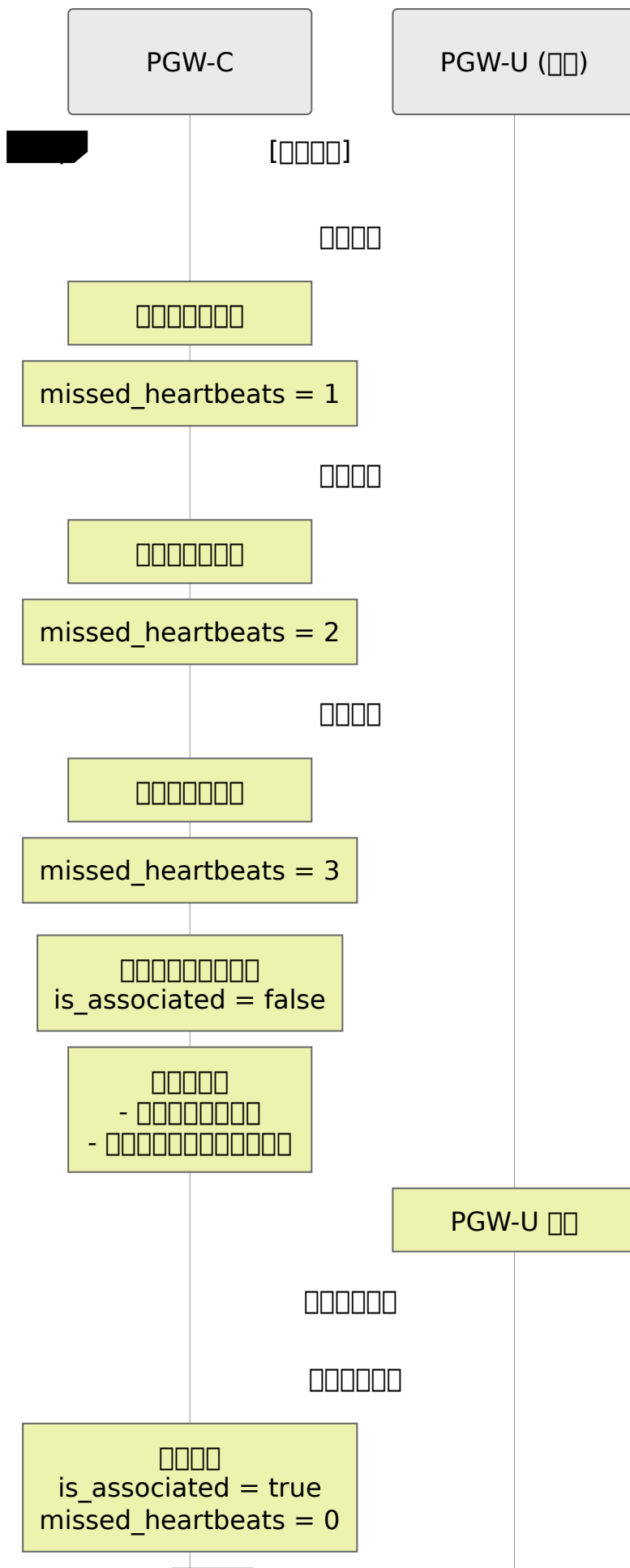
□□□□□□□

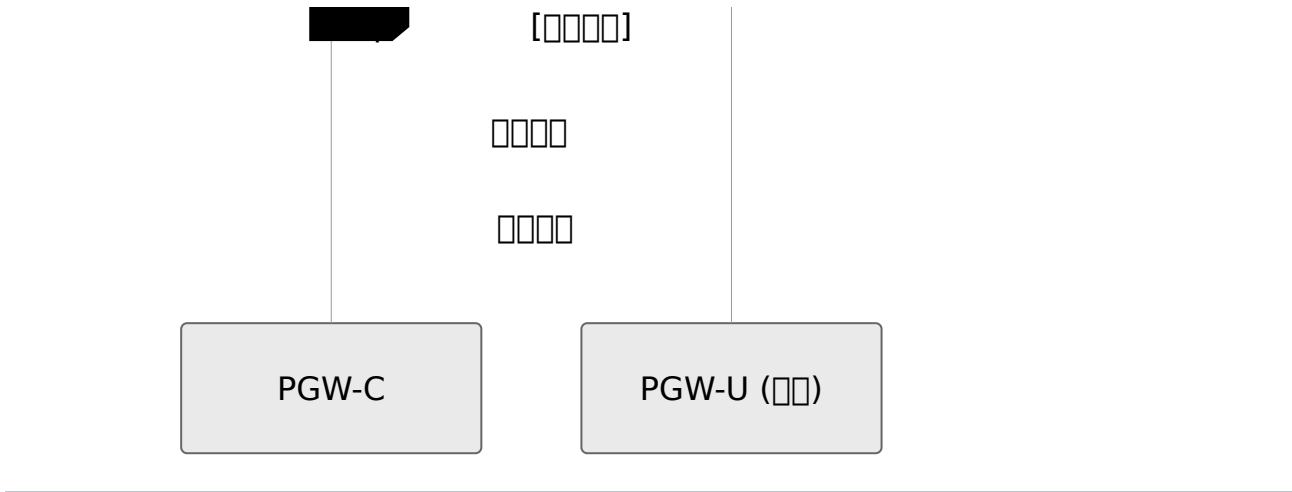


□□□□□



□□□□□□





[ ]

[ ]

## 1. [ ]

[ ]

- [ ]“PFCP [ ]”
- [ ]

[ ]

- PGW-U [ ]
- PGW-U [ ]
- [ ] UDP [ ] 8805
- [ ] `remote_ip_address` [ ]

[ ]

```
# ping
ping <pgw_u_ip_address>

# nc UDP
nc -u -v <pgw_u_ip_address> 8805

# iptables
iptables -L -n | grep 8805
```

## 2. 確認

確認

- 確認“確認確認3”
- 確認確認

確認

- 確認確認
- PGW-U 確認
- 確認確認

❓❓❓確認

確認確認 5 確認確認確認 3 確認

## 3. 確認

確認

- 確認確認確認
- 確認“PFCP 確認確認”

確認

- 確認 PGW-U 確認
- PGW-U 確認
- 確認確認

□□□

1. □□□□□□□□□□ `is_associated = true`
2. □□ PGW-U □□□□□□
3. □□ SEID □□□□

#### 4. □□ **SEID** □□

□□□

- □□□□□□□□□□“□□□□□□□□□□”

□□□

- SEID □□□□□□□□
- PGW-U □□□ PGW-C □□

□□□□□

- □□□□ PFCP □□□□□□□□□□□□□□
- PGW-C □□□□ PGW-U □□□□□□□□

### □□ **PFCP** □□

□□□□□□□

```
# PFCP 关联
pfcpeer_associated{peer="PGW-U Primary"} 1

# PFCP 注册
seid_registry_count 150

# PFCP 速率
rate(sxb_inbound_messages_total[5m])

# PFCP 错误
rate(sxb_inbound_errors_total[5m])

# 心跳失败
pfcpeer_consecutive_heartbeat_failures{peer="PGW-U Primary"} 0
```

告警规则

```
# PFCP 告警规则
- alert: PFCPAssociationDown
  expr: pfcpeer_associated == 0
  for: 1m
  annotations:
    summary: "PFCP 关联失败 {{ $labels.peer }}"

# PFCP 会话建立失败告警
- alert: PFCPSessionEstablishmentFailureHigh
  expr:
rate(sxb_inbound_errors_total{message_type="session_establishment_res
[5m]) > 0.1
  for: 5m
  annotations:
    summary: "PFCP 会话建立失败"
```

## Web UI - PFCP 告警

OmniPGW 告警 Web UI 告警规则 PFCP/Sxb 告警

# UPF/PFCP 配置

URL: `http://<omnipgw-ip>:<web-port>/upf_status`

PGW-U 配置 PFCP 配置

配置

## 1. 配置

- 配置 - PFCP 配置
- 配置 - 配置
- 2 配置

## 2. 配置 PGW-U 配置

- 配置 - 配置
- **IP** 配置 - PGW-U IP
- 配置 - 配置
- **ID** 配置 - PFCP 配置
- 配置 - 配置
- 配置 - 配置

- **PGW-U** - **PGW-U**
- **UP** - PGW-U

### 3. PGW-U

- **PGW-U**
- UP
- PGW-U
- PGW-U

## PFCP

PGW-U `http://<omnipgw-ip>:<web-port>/pfcpsessions`

PGW-U OmniPGW PGW-U PFCP

PGW-U

### 1. PGW-U

- PGW-U
- PGW-U

## 2. 網路網路 PFCP 網路

- 網路 - 網路網路
- 網路 ID - 網路網路
- IMSI - 網路網路網路網路
- 網路 - 網路

## 3. 網路網路網路 網路網路網路

- 網路 PFCP 網路網路
- PDRs/FARs/QERs/BARs網路網路
- F-SEIDs網路網路網路
- PGW-U 網路網路

網路網路

網路 PFCP 網路網路

1. 網路 UPF 網路
2. 網路網路網路“網路”
3. 網路網路網路 = 0
4. 網路網路“網路”
  - 網路 IP 網路
  - 網路網路網路
  - 網路 (UDP 8805)

網路網路網路網路

1. 網路網路網路
2. 網路 PGW 網路 - 網路網路
3. 網路 PFCP 網路 - PFCP 網路網路
4. 網路 PFCP 網路
  - 網路 UPF 網路 - 網路網路網路網路
  - 網路網路 PFCP 網路
5. 網路 PFCP 網路網路
  - 網路 PDRs/FARs 網路網路網路網路
  - 網路網路網路PGW-U 網路

## 功能需求

1. 支持 PGW-U 功能
2. 支持 PCF 功能
3. 支持策略控制功能
4. 支持计费功能

## 性能需求

- 支持 UPF 功能
- 支持“切片”功能
- 支持策略控制功能
- 支持 Web UI 功能

## 安全需求

- 支持 - 支持 SSH
- 支持 - 支持/支持
- 支持 - 支持
- 支持 - 支持 PDRs/FARs/QERs
- 支持 - 支持

## 接口需求

### 接口

- 接口 - UPF 支持 PCF 接口
- 接口 - PDN 支持

### 接口

- **Diameter Gx** 接口 - 支持 PCF QoS 支持 PCC 接口
- **Diameter Gy** 接口 - 支持 URRs 支持
- **CDR** 接口 - 支持 PCF 支持 CDR

- **QoS** - PFCP 与 UPF 交互

|  |  |  |  |
|--|--|--|--|
|  |  |  |  |
|--|--|--|--|

- **S5/S8** 接口 - 承载用户数据
- **UE IP** 地址 - 通过 PFCP 向 UE 分配

000000

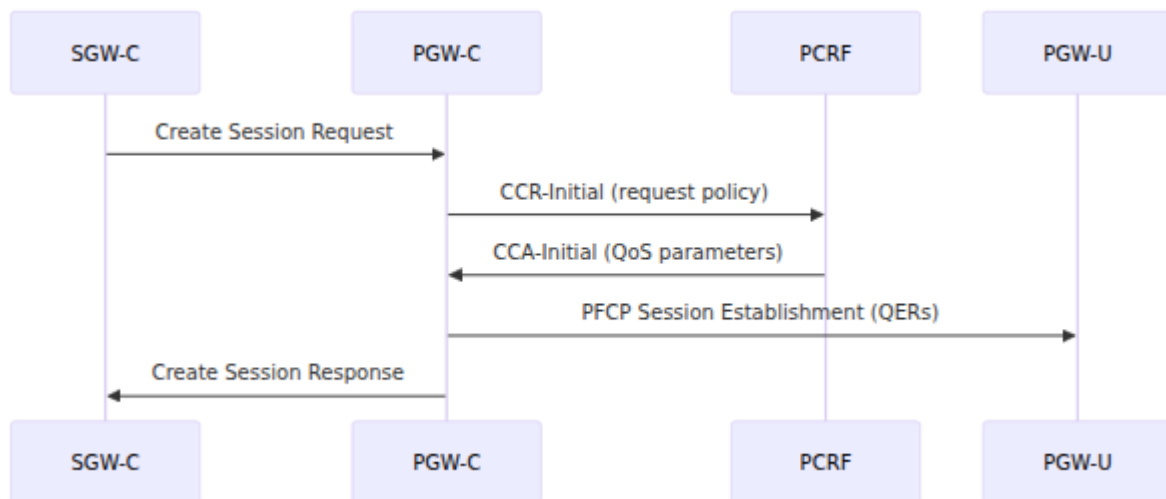
# QoS

## QoS

PGW-C QoS

- **Gx (Diameter)** - PCRF QoS
- **S5/S8 (GTP-C)** - SGW-C
- **Sxb (PFCP)** - QoS PGW-U

## QoS



## QoS

- UE: UE PDR/FAR/QER/BAR AMBR
- EBI: EBI (EPS ID) PDR/FAR QER
- **QER (QoS)**: MBR/GBR
- PDN: PDN
- PCRF: PCRF QoS

□□

## □□□□: □□ **QoS** □□

□□ QoS □□□□ Diameter Gx □□□□□□□□ PCRF□□□□ PCRF □□□□□□□□□□□□□□□□ OmniHSS□□

□□□□ `config/runtime.exs` □□□□ **PCRF** □□□□

```
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
      }
    ]
  }
}
```

**QoS** □□□□□□□□□□□□□□ **PCRF** □□□□□□□□□□ **PGW-C** □□□□□□□□

□□□□□□□□

□□□□□□□□

□□□□□□ **PDN** □□□□□□□□□□



Create Session Request

AllocateIP

UE IP assigned

RequestPolicy

CCR-Initial sent to PCRF

CreateBearer

CCA-Initial received  
with QoS

ProgramUPF

PFCP Session  
Establishment

Active

Delete Session Request



□□□□:

1. SGW-C □□ Create Session Request
2. PGW-C □□□□□□□□ UE IP □□
3. PGW-C □□ CCR-Initial □ PCRF□□□ IMSI□APN□IP □□
4. PCRF □□ CCA-Initial□□□ QoS □□□
  - Default-EPS-Bearer-QoS (QCI, ARP)
  - QoS-Information (AMBR □□)

## 5. PGW-C 消息交互

- 消息 ID: 消息 PDR=1 消息 PDR=2 消息 FAR=1 消息 FAR=2 消息 QER=1 消息 BAR=1
- QER 消息 QoS 消息 MBR

## 6. PGW-C 向 PGW-U 发送 PFCP Session Establishment Request

## 7. PGW-C 向 SGW-C 发送 Create Session Response

消息内容:

- 本地 PDN 地址
- 本地 QCI 5 本地 QCI 9 本地 GBR
- EBI 地址
- 本地地址

消息内容

消息内容 PCRF 消息内容

消息: 消息 PCRF 向 Re-Auth Request (RAR) 消息 Charging-Rule-Install

消息内容:

### 1. PCRF 向 RAR 消息 Charging-Rule-Definition 消息内容

- Charging-Rule-Name (消息内容)
- Flow-Information (消息内容)
- QoS-Information (QCI, MBR, GBR, ARP)
- Precedence (消息内容)

### 2. PGW-C 向 PCRF 发送 PFCP 消息

- 本地 Flow-Information 消息 → 本地 PDR 消息 SDF 消息
- QoS-Information → 本地 QER 消息 MBR/GBR 消息
- Flow-Description → IP 5-消息内容

### 3. PGW-C 向 PCRF 发送 PFCP Session Modification Request 消息 PDRs/FARs/QERs

### 4. PGW-C 向 SGW-C 发送 Create Bearer Request

### 5. SGW-C 向 PGW-C 发送 Create Bearer Response 消息内容

消息 **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)
```

□□□□

□□ QoS □□□□□□□□□□

- **Gx RAR**□□□□□ Charging-Rule-Definition
- **PFCP Session Modification**□□□□□□ QERs□□□□□□□□ FARs□□□□□□□ PDRs□□□□□□□□□□

□□□□

□□:

- **Delete Session Request**□SGW □□□ - □□□□□□□□□□□□
- **Re-Auth Request with Charging-Rule-Remove**□PCRF □□□ - □□□□□□□

□□□□:

1. □□□□□□□□□□□□
2. □□□□□ PDRs/FARs/QERs
3. □ SGW-C □□ Delete Bearer Request□□□□□ PCRF □□□
4. □□ PFCP Session Modification□□□□□□□□ Session Deletion□□□□□□□□□□

# QoS

## QCI (QoS Class Identifier)

PCRF → Gx → **QoS-Class-Identifier** → AVP

QCI:

- **QCI 1:** GBR, 100ms delay budget
- **QCI 2:** GBR, 150ms delay budget
- **QCI 3:** GBR, 50ms delay budget
- **QCI 4:** GBR, 300ms delay budget
- **QCI 5:** IMS Signaling (GBR, 100ms delay budget) - highest priority
- **QCI 6:** TCP-based traffic (GBR, 300ms delay budget)
- **QCI 7:** RTT sensitive traffic (GBR, 100ms delay budget)
- **QCI 8:** TCP-based traffic like YouTube (GBR, 300ms delay budget)
- **QCI 9:** GBR, 300ms delay budget

QoS:

- QCI → PCRF → QoS IE → SGW-C → PGW-C
- PGW-C → QCI → QERs → MBR/GBR
- QCI → QoS
- QCI → QoS

## ARP (Allocation-Retention-Priority)

PCRF → **Allocation-Retention-Priority** → AVP

ARP:

- ARP: 1 to 15
- ARP: 0 = lowest priority, 1 = highest priority
- **???** ARP: 0 to 15

- 0 = 00000000
- 1 = 11111111

000:

- 00000: 1
- 0000: 00 (0)
- 00000: 00 (1)

00000:

- ARP → SGW-C → 00000000 eNodeB
- **PGW-C** 00000 - 00000 eNodeB 0000000000
- 000000000000000000000000
- 00000000000000 10000000000000

## MBR (000000)

00: PCRF 00 Max-Requested-Bandwidth-UL 0 Max-Requested-Bandwidth-DL  
AVPs

00: 000000000000 kbps: bytes / 1000

0000: 000000000000

0000:

- PGW-C 00 QER000 mbr: %Bitrate{ul: kbps\_ul, dl: kbps\_dl}
- QER 00 PFCP 000 PGW-U
- **PGW-U** 0000000000000000
- 00 MBR 00000000

00:

Max-Requested-Bandwidth-UL: 5000000 (5 Mbps)  
Max-Requested-Bandwidth-DL: 10000000 (10 Mbps)

→ QER `qer` mbr: {ul: 5000, dl: 10000} kbps  
→ PGW-U `pgw-u` 5 Mbps `pgw-u`  
→ PGW-U `pgw-u` 10 Mbps `pgw-u`

## GBR (Guaranteed Bit Rate)

`qer`: PCRF `qer` `Guaranteed-Bitrate-UL` `Guaranteed-Bitrate-DL` AVPs

`qer`: `qer` kbps

`qer`: `qer` GBR `qer`

`qer`:

- `qer` Charging-Rule-Definition `qer` GBR `qer` **GBR** `qer`
- PGW-U `qer` QER `qer`
- `qer` eNodeB `qer`
- GBR `qer` - `qer`

`qer`:

Guaranteed-Bitrate-UL: 1000000 (1 Mbps)  
Guaranteed-Bitrate-DL: 2000000 (2 Mbps)

→ QER `qer` gbr: {ul: 1000, dl: 2000} kbps  
→ `qer` 1 Mbps `qer` 2 Mbps `qer`  
→ `qer` VoIP `qer`

`qer`:

- GBR `qer`
- `qer` GBR `qer`
- `qer` GBR `qer`

## AMBR (Aggregate-Max-Bitrate)

PCRF → APN-Aggregate-Max-Bitrate-UL / APN-Aggregate-Max-Bitrate-DL AVPs

GBR APN

Flow-Status:

- AMBR GBR
- Create Session Response SGW-C
- eNodeB/SGW
- PGW-C AMBR SGW-C

Flow-Status:

APN-Aggregate-Max-Bitrate-UL: 50000000 (50 Mbps)  
APN-Aggregate-Max-Bitrate-DL: 100000000 (100 Mbps)

- GBR 50 Mbps / 100 Mbps
- MBR
- AMBR UE/APN

Flow-Status:

- HSS/PCRF
- 10 Mbps 100 Mbps
- GBR

Flow-Status

(Gx) (PFCP)

PCRF → Flow-Status AVP → Charging-Rule-Definition

| Flow-Status (Gx)     | Gate-Status (PFCP QER) | 備考                 |
|----------------------|------------------------|--------------------|
| 0 = ENABLED-UPLINK   | ul: OPEN, dl: CLOSED   | アップリンクのみ許可         |
| 1 = ENABLED-DOWNLINK | ul: CLOSED, dl: OPEN   | ダウンリンクのみ許可         |
| 2 = ENABLED          | ul: OPEN, dl: OPEN     | アップリンクとダウンリンクの両方許可 |
| 3 = DISABLED         | ul: CLOSED, dl: CLOSED | アップリンクとダウンリンクの両方禁止 |
| 4 = REMOVED          | ul: CLOSED, dl: CLOSED | アップリンクとダウンリンクの両方禁止 |

備考:

- **DISABLED:** アップリンクとダウンリンクの両方禁止
- **ENABLED-UPLINK:** アップリンクのみ許可
- **ENABLED-DOWNLINK:** ダウンリンクのみ許可
- **ENABLED:** アップリンクとダウンリンクの両方許可

セッション登録

## Prometheus 監視

◆◆◆監視:

```
session_registry_count      # セッション (IMSI, EBI 別)
address_registry_count      # ユーザ IP
charging_id_registry_count  # 課金ID
```

Gx 監視:

```

gx_inbound_messages_total{message_type="gx_RAR"}      # [] PCRF []
[]
gx_outbound_messages_total{message_type="gx_CCR"}     # [] PCRF []
[]
gx_outbound_transaction_duration_bucket              # [] PCRF []

```

## PCFP :

```

sxb_outbound_messages_total{message_type="pfcg_session_establishment_
sxb_outbound_messages_total{message_type="pfcg_session_modification_r
sxb_outbound_transaction_duration_bucket

```

## :

```

s5s8_inbound_messages_total{message_type="create_session_request"}
# []
s5s8_outbound_messages_total{message_type="create_bearer_request"}
# []

```

## Web UI

### PGW (/pgw\_sessions):

- IMSI/IP []MSISDN [] APN []
- []
- [] QoS [] (QCI, MBR, GBR, AMBR)
- [] (2 )

### Diameter (/diameter):

- PCRF []
- Gx []
- []/[]

### (/logs):

- []

- "Credit Control" CCR/CCA
- "Re-Auth" RAR
- "PFCP"

```
[debug] Sending Credit Control Request: ... # CCR  PCRF
[debug] Handling Credit Control Answer: ... # CCA  PCRF
      QoS
[debug] Handling Re-Auth Request # RAR  PCRF
      QoS
[debug] Sending Session Establishment Request # PFCP  PGW-
      QERs
[debug] Sending Session Modification Request # PFCP  PGW-
      QERs
```

## QoS

1. Web UI → **PGW**
2. IMSI 999000123456789
- 3.
4. **qer\_map**

```
qer_id: 1
gate_status: {ul: OPEN, dl: OPEN}
mbr: {ul: 50000, dl: 100000} # kbps
gbr: {ul: 10000, dl: 20000} # kbps nil GBR
```

5. PCRF

## QoS

: QoS

11

### 1. **PCRf** **:**

- Web UI → **Diameter**
- PCRF state = "connected"
- Diameter

## 2. **CCR/CCA** **□□**:

- Web UI → **Logs**
- "Credit Control Answer"
- CCA contains QoS-Information AVP
- CCA contains Result-Code 2001 = SUCCESS

### 3. **PFCP** **□□**:

- "PFCP Session Establishment Request"
- QER
- PGW-U PFCP

#### 4. ☐ **PCRf** ☐

- 3GPP PCRF 3GPP 3GPP
- 3GPP APN 3GPP
- 3GPP PCRF 3GPP 3GPP

|  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|

## Prometheus ☐☐:

```
# 5S8 inbound messages / min
rate(s5s8_inbound_messages_total{message_type="create_session_request"}[5m])

# 5S8 outbound messages / min
rate(s5s8_outbound_messages_total{message_type="create_bearer_request"}[5m])

# PCRF Gx inbound messages / min
rate(gx_inbound_messages_total{message_type="gx_RAR"}[5m])
```

Queries

Queries:

```
# UE IP address registry count
(address_registry_count / <configured_pool_size>) * 100

# Session registry count
session_registry_count

# PCRF P95 transaction duration
histogram_quantile(0.95, gx_outbound_transaction_duration_bucket)
```

Queries:

- `config/runtime.exs` `ue.subnet_map`
- TEID 32 40
- 

Queries:

- IP - 80%
- PCRF -
-

## □□□□

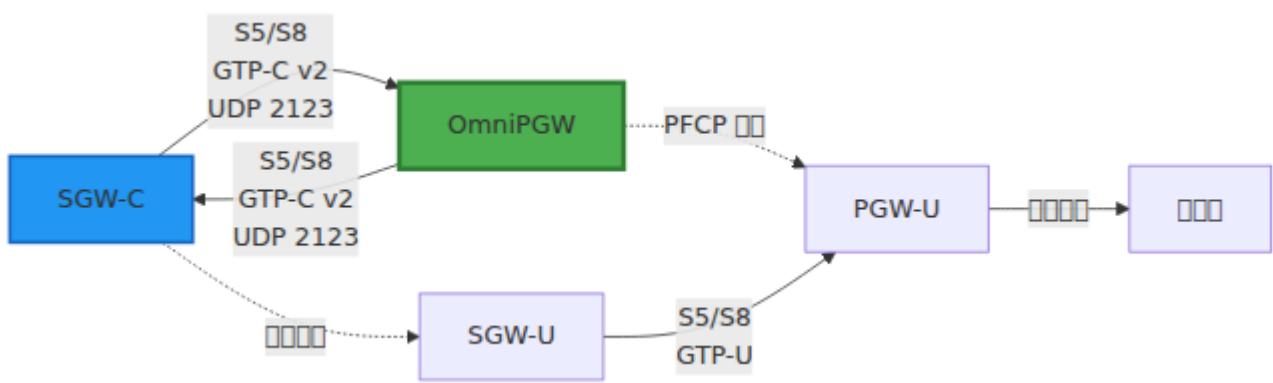
- □□□□ - PDN □□□□□□
- Diameter Gx □□ - PCRF □□□□□□□□
- PFCP □□ - □□□□□□
- □□□□ - □□□□
- □□□□ - □□□□□□□□

# S5/S8

SGW-C GTP-C

OmniPGW Omnitouch

S5/S8 GTP-C v2 GPRS - OmniPGW SGW-C



## GTP-C 2

- GTP-C v2 (3GPP TS 29.274)
- UDP
- 2123
- 

## TEID

TEID

- **TEID** - OmniPGW
- **TEID** - SGW-C

SGW-C → OmniPGW: TEID = OmniPGW TEID

OmniPGW → SGW-C: TEID = SGW-C TEID

```
# config/runtime.exs
config :pgw_c,
  s5s8: %{
    # S5/S8 IPv4
    local_ipv4_address: "10.0.0.20",

    # IPv6
    local_ipv6_address: nil,

    #
    local_port: 2123,

    # GTP-C 500ms
    # GTP-C
    request_timeout_ms: 500,

    # GTP-C 3
    # = request_timeout_ms * request_attempts
    # 500ms * 3 = 1500ms 1.5
    request_attempts: 3
  }
```

S5/S8 GTP-C

□□□

- `request_timeout_ms` - □□□□□□□□□□□□□□□□500ms□
- `request_attempts` - □□□□□□□□□□□□3□

□□□□□□ `request_timeout_ms × request_attempts`

□□□□□ 500ms × 3 □□□ = □□□□□ **1.5** □

□□□□□

| □□□□□        | □□□ <code>request_timeout_ms</code> | □□□□□             |
|--------------|-------------------------------------|-------------------|
| □□□□□<50ms□  | 200-300ms                           | 600-900ms□3 □□□□□ |
| □□□50-150ms□ | 500ms□□□□□                          | 1.5s□3 □□□□□      |
| □□□□□>150ms□ | 1000-2000ms                         | 3-6s□3 □□□□□      |
| □□□□/□□      | 2000-3000ms                         | 6-9s□3 □□□□□      |

□□ - □□□□□□□

```
s5s8: %{  
  local_ipv4_address: "10.0.0.20",  
  request_timeout_ms: 1500, # □□□□□ 1.5 □  
  request_attempts: 3      # □□□□□□ 4.5 □  
}
```

□□□□□□□□

- OmniPGW □□□□□□ "□□□□□□□□□□"
- □□□□□□ PCRF□□□□□□□□□□5012 UNABLE\_TO\_COMPLY□
- □□□□□□□□□□□□□□□□□□□□ Charging-Rule-Remove □□

□□□□□

□□□□□□□

```
# SGW-C GTP-C
iptables -A INPUT -p udp --dport 2123 -s <sgw_network>/24 -j
ACCEPT

# SGW-C GTP-C
iptables -A OUTPUT -p udp --dport 2123 -d <sgw_network>/24 -j
ACCEPT
```

```
# SGW-C
ip route add <sgw_network>/24 via <gateway_ip> dev eth0
```

---

S5/S8 PDN GTP-C

SGW-C → OmniPGW

PDN

**IEs**

| IE 名称  | 类型  | 值                                     |
|--------|-----|---------------------------------------|
| IMSI   | 字符串 | 310260123456789                       |
| MSISDN | 字符串 | 14155551234                           |
| APN    | 字符串 | internet                              |
| RAT 类型 | 枚举  | EUTRAN                                |
| QoS 参数 | 字符串 | QCI 9, ARP, S5/S8 F-TEID (SGW-U FQDN) |
| UE 名称  | 字符串 | UE 名称                                 |
| ULI    | 字符串 | TAI, ECGI                             |
| PLMN   | 字符串 | MCC/MNC                               |

数据

```
数据
├─ IMSI: 310260123456789
├─ MSISDN: 14155551234
├─ APN: internet
├─ RAT 类型: EUTRAN (6)
├─ QoS 参数
│   ├─ EBI: 5
│   ├─ QoS (QCI 9, ARP, S5/S8 F-TEID (SGW-U FQDN))
├─ ULI
│   ├─ TAI: MCC 310, MNC 260, TAC 12345
│   └─ ECGI: MCC 310, MNC 260, ECI 67890
```

数据

OmniPGW → SGW-C

数据 数据

IEs

| IE  |    |             |
|-----|----|-------------|
|     |    |             |
|     |    |             |
|     |    |             |
| PDN | IP | UE IP UE IP |
| APN |    | APN         |
| PCO |    | PCO         |

```

{
  |  : (16)
  | PDN
  |   IPv4: 100.64.1.42
  |
  |
  |   EBI: 5
  |   :
  |   S5/S8 F-TEID ( PFCP PGW-U )
  | APN : Public-1 (1)
  | PCO
  |   DNS : 8.8.8.8
  |   DNS : 8.8.4.4
  |   MTU: 1400
}
```

SGW-C → OmniPGW

PDN

IEs

| IE 項目  | 項目             |
|--------|----------------|
| EBI    | 項目項目 EPS 項目 ID |
| 項目 EBI | 項目項目項目項目       |

項目項目項目

項目項目 OmniPGW → SGW-C

項目項目 項目項目項目

項目項目 **IEs**

| IE 項目 | 項目       |
|-------|----------|
| 項目    | 項目項目項目項目 |

項目項目項目

項目項目項目

項目項目 OmniPGW → SGW-C

項目項目 項目項目項目項目 PCRF 項目項目項目

項目項目項目

- PCRF 項目項目 PCC 項目項目項目項目項目
- OmniPGW 項目項目 SGW-C 項目項目項目

項目項目項目

項目項目 OmniPGW → SGW-C 項目 SGW-C → OmniPGW

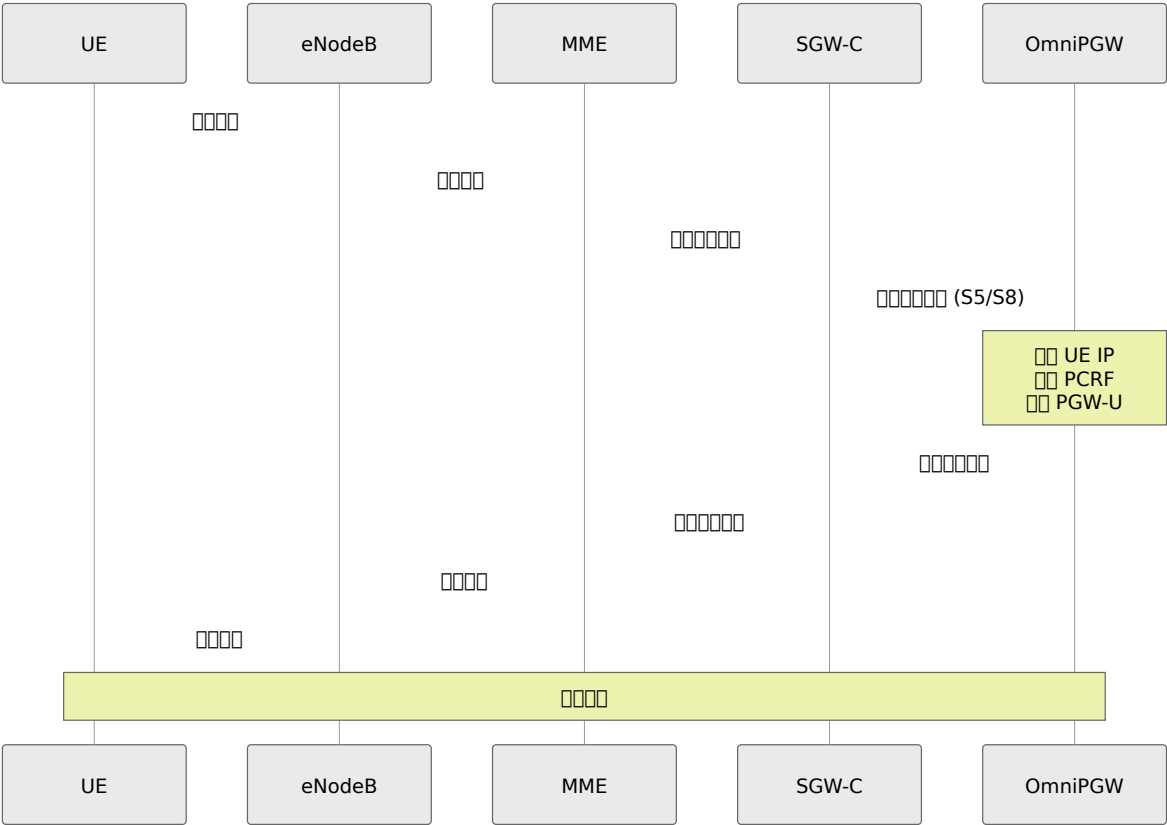
項目項目 項目項目項目

項目項目

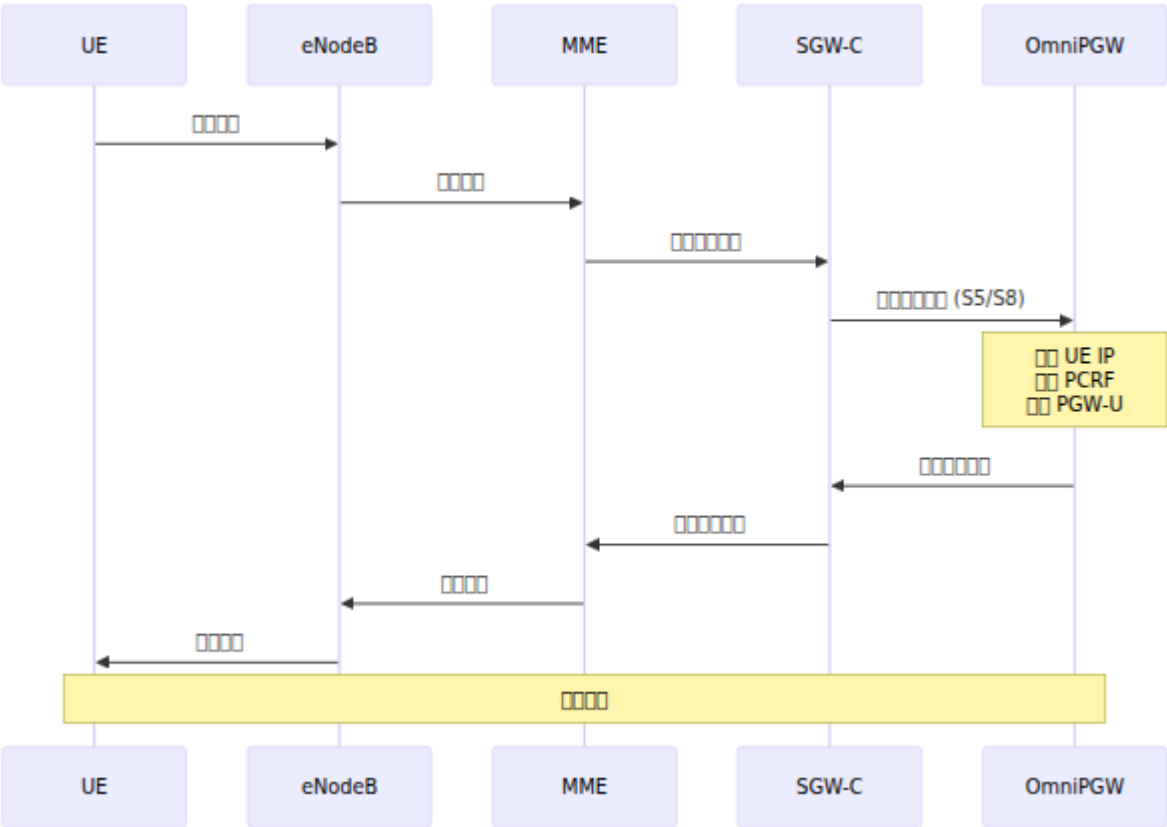
- **PGW** 通过 PCRF 实现策略控制
- **SGW** 实现路由转发

网络

网络



□□□□



□□□□

□□

|    |       |      |
|----|-------|------|
| □□ | □□    | □□   |
| 16 | □□□□□ | □□□□ |

□□□□□□□□

| □□ | □□          | □□□□              |
|----|-------------|-------------------|
| 65 | □□□□        | PCRF □□□□□□ IMSI□ |
| 66 | □□□□□□      | IP □□□            |
| 93 | □□□□□□□     | □□□ APN           |
| 94 | TFT □□□□□□□ | □□□□□□□□□         |

□□□□□□□□

| □□ | □□         | □□□□          |
|----|------------|---------------|
| 72 | □□□□□□□□□  | PCRF/PGW-U □□ |
| 73 | □□□□□□□□□□ | □□□□          |

---

□□

## S5/S8 □□

```
# □□□□□
s5s8_inbound_messages_total{message_type="create_session_request"}
s5s8_inbound_messages_total{message_type="delete_session_request"}

# □□□□□
s5s8_inbound_errors_total

# □□□□□□
s5s8_inbound_handling_duration_bucket

# □□ TEID
teid_registry_count
```

□□□□□

□□□□□

```
rate(s5s8_inbound_messages_total{message_type="create_session_request"}[5m])
```

□□□□

```
rate(s5s8_inbound_errors_total[5m])
```

□□□**p95**□□

```
histogram_quantile(0.95,

rate(s5s8_inbound_handling_duration_bucket{request_message_type="create_session_request"}[5m])
)
```

---

## 環境構築

### 環境構築 **OmniPGW** 環境

環境

- SGW-C 環境構築
- 環境構築
- SGW-C 環境

環境

1. 環境構築
2. OmniPGW 環境構築 IP 環境
3. 環境構築 UDP 2123
4. 環境構築 TEID 環境

環境

```
# 環境 OmniPGW 環境構築
netstat -ulnp | grep 2123

# 環境構築
tcpdump -i any -n port 2123

# 環境構築
grep "local_ipv4_address" config/runtime.exs

# 環境構築
iptables -L -n | grep 2123
```

## 環境構築

環境

- 環境構築
- 環境構築

□□□□□

□□ 65□□□□□□□□

→ PCRF □□□□

→ □□ HSS/SPR □□ IMSI

□□ 66□□□□□□

→ IP □□□

→ □□□ curl http://pgw:9090/metrics | grep address\_registry\_count

→ □□ IP □

□□ 72□□□□□□□□□□□□

→ PCRF □□□ PGW-U □□□□

→ □□ Gx □□

→ □□ PFCP □□

## □□□TEID □□

□□□

- □□□□□□□□□□
- □□□□

□□□

- TEID □□□□□□□□
- TEID □□□□□□

□□□□□

- □□□□□ TEID □
  - □□ TEID □□□□□□□□
-

## □□□□

## □□□□

### 1. □□□□□□

- □ S5/S8 □□□□ VLAN
- □□□□□□

### 2. **MTU** □□

- □□ MTU □□ GTP □
- □□ MTU□1500 □□□1464 □□□□ + 36 GTP□

### 3. □□

- □□ OmniPGW □□
- □ SGW-C □□ DNS □□□□

## □□

### 1. **UDP** □□□□□

- □□□□□□□□□□□□
- □□□□□□□□ 4-8 MB

### 2. □□□□

- □□□□□□□□□□
- □□ TEID □□□□□

## □□

### 1. **IP** □□

- □□□□□□□ SGW-C IP □ GTP-C
- □□ iptables □□□ ACL

## 2. 网络

- OmniPGW 网络
  - 网络 GTP-C 网络
- 

### 网络

#### 网络

- 网络 - S5/S8 网络 IP 网络
- 网络 - PDN 网络
- **UE IP** 网络 - 网络 IP 网络
- **PCO** 网络 - GTP-C 网络 PCO 网络

#### 网络

- **PFCP** 网络 - 网络 S5/S8 网络
- 网络 **Gx** 网络 - 网络网络网络
- 网络 **Gy** 网络 - 网络网络

#### 网络

- 网络 - S5/S8 GTP-C 网络
  - 网络 **CDR** 网络 - 网络 GTP-C 网络 CDR
- 

### 网络

---

**OmniPGW S5/S8 网络 - 网络 Omnitouch 网络**



11

```
graph TD; Start(( )) -- "Red Green Blue Yellow" --> Creating[Creating]; Creating -- "Red Green Blue Yellow" --> Active[Active]; Active -- "Red Green Blue Yellow" --> Modifying[Modifying]; Active -- "Red Green Blue Yellow" --> Terminating[Terminating]; Modifying -- "Red Green Blue Yellow" --> Active; Terminating -- "Red Green Blue Yellow" --> Active; Terminating -- "Red Green Blue Yellow" --> End((( ))); Creating -- "Red Green Blue Yellow" --> End;
```

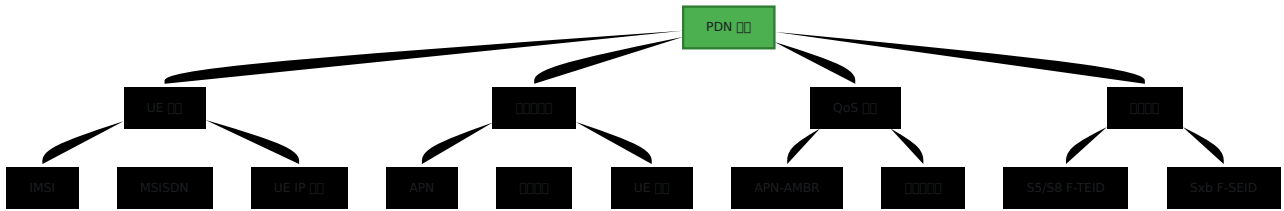
□□□□

□□□□□

□□□□□□□□□□□□□□□□

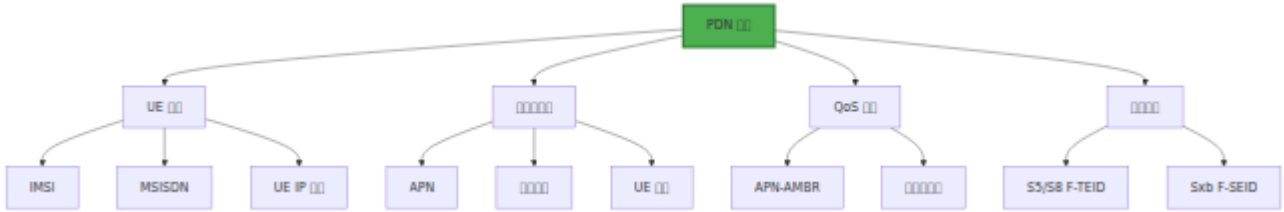
| □□□         | □□            | □□                   |
|-------------|---------------|----------------------|
| TEID        | S5/S8 (GTP-C) | SGW-C □□□□□□ ID      |
| SEID        | Sxb (PFCP)    | PGW-U □□□□□□ ID      |
| Session-ID  | Gx (Diameter) | PCRF □□□ Diameter □□ |
| Charging-ID | □□            | □□□□□/□□ ID          |

□□□□



□□□□

□□□□



□□

## 1. □□□□□□ (S5/S8)

□□□□□ S5/S8 □□□ GTP-C □□□□□□□□ GTP-C □□□□□□□□□□ S5/S8 □□□

□□□

- IMSI, MSISDN, IMEI
- APN (□□□“internet”)
- RAT □□ (EUTRAN)
- UE □□ (TAI, ECGI)
- □□□□□ (QoS, F-TEID)

## 2. □□□□

- □ APN □□□□ UE IP
- □□□□ ID
- □□ Gx □□-ID
- □□ S5/S8 TEID
- □□ PGW-U □□□

## 3. □□□□ (Gx)

□ PCRF □□□□□

- □□ CCR-□□
- □□□□ QoS □ PCC □□□ CCA-□□

## 4. □□□□□□ (PFCP)

□□□□□□□□ PGW-U□

- □□□□□□□□
- □□ PDRs, FARs, QERs, BAR
- □□ S5/S8 □□□ F-TEID

## 5. □□ SGW-C

## UE Parameters

- UE IP
  - S5/S8 F-TEID (PGW-U)
  - PCO (DNS, P-CSCF, MTU)
  - QoS
- 

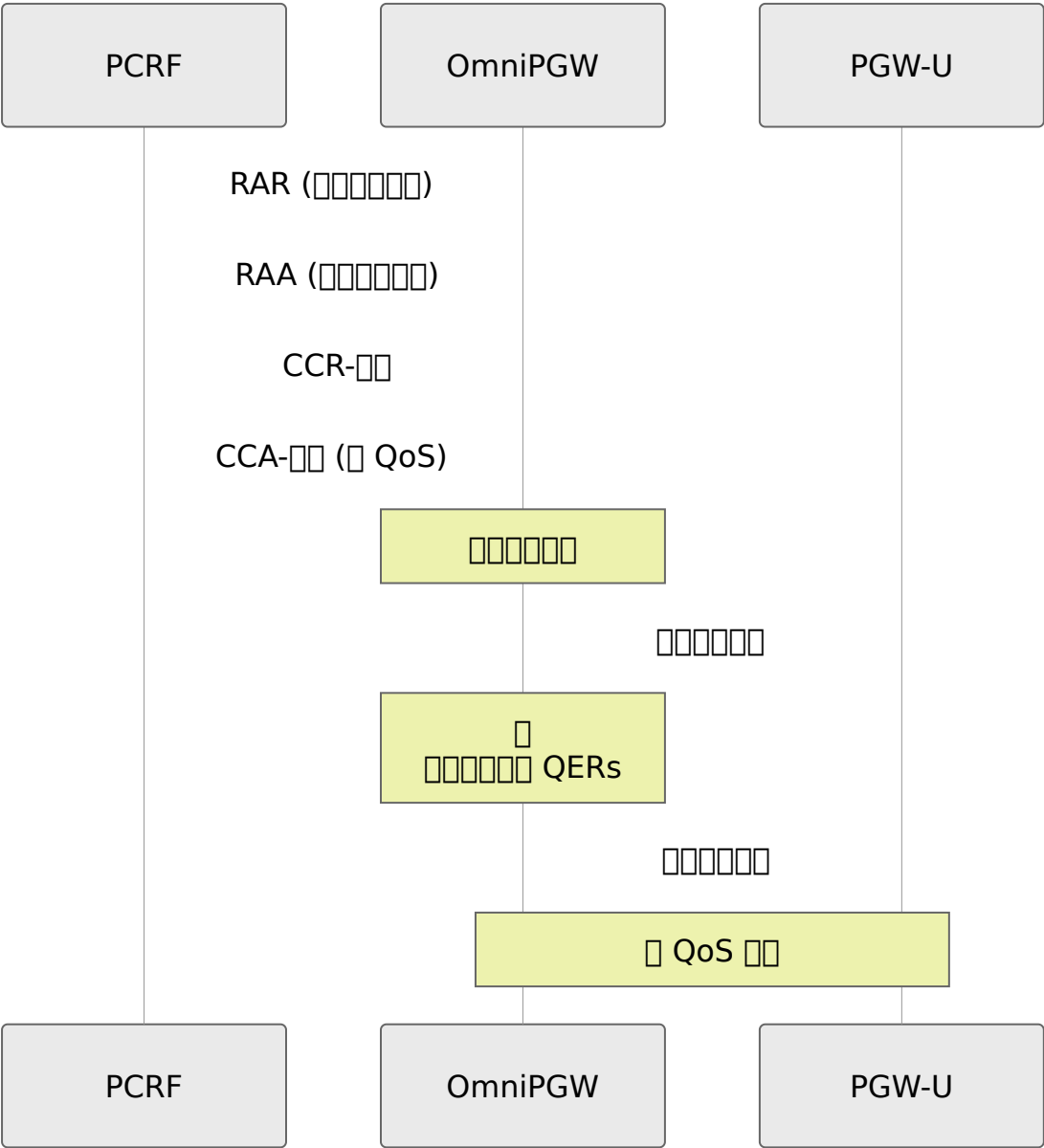
## Network

### QoS

#### QoS Parameters

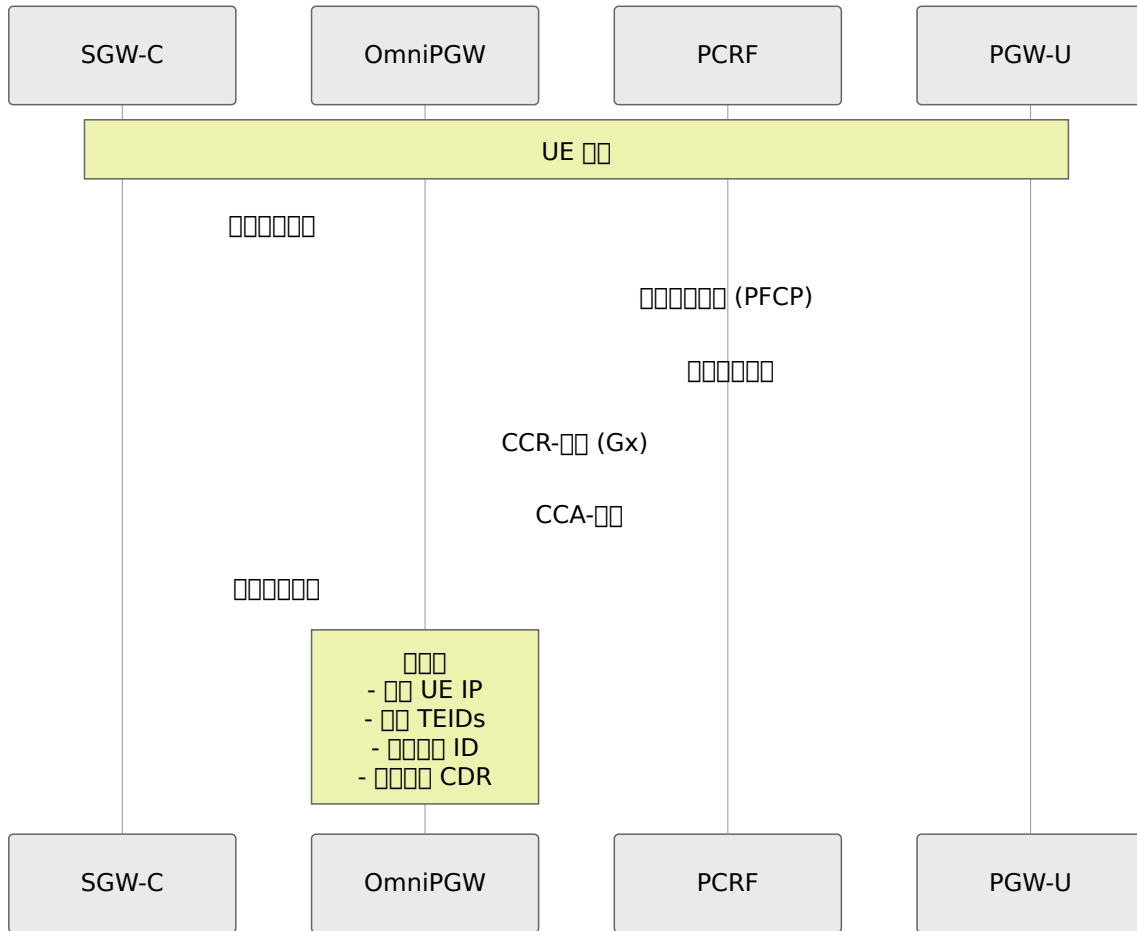
- **QoS** - PCRF
- - /
- - SGW
- - PCRF PCC

QoS



□□□□

□□□□



□□□□

□□□□□□

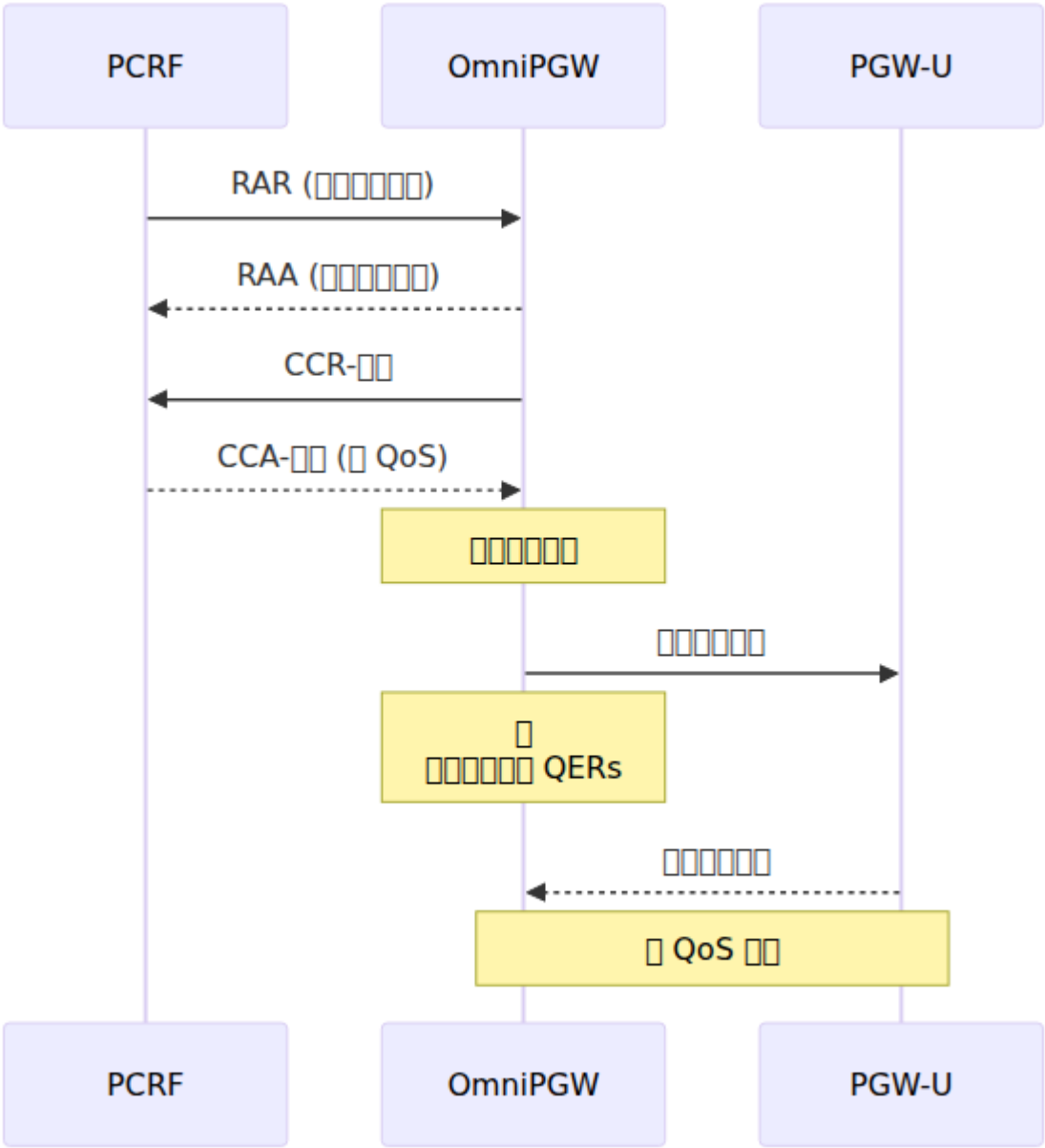
1. UE IP □□ → □□□□
2. TEID → □□□□□□
3. SEID → □□□□□□
4. □□-ID → □□□□□□
5. □□-ID → □□
6. □□□□□□

□□□□□□□□

- 3GPP CDR (3GPP) - 3GPP CDR

3GPP

3GPP



3GPP

3GPP

- ☐ TEID (S5/S8):  
TEID 0x12345678 → ☐☐ PID
- ☐ SEID (Sxb):  
SEID 0xABCDEF → ☐☐ PID
- ☐ ☐☐-ID (Gx):  
"pgw.example.com;123;456" → ☐☐ PID
- ☐ UE IP:  
100.64.1.42 → ☐☐ PID
- ☐ IMSI + EBI:  
"310260123456789" + EBI 5 → ☐☐ PID

☐☐☐☐

☐☐☐☐☐☐

```
# ☐☐☐☐☐  
teid_registry_count  
  
# PFCP ☐☐  
seid_registry_count  
  
# Gx ☐☐  
session_id_registry_count
```

## Queries

```
# Session creation
rate(s5s8_inbound_messages_total{message_type="create_session_request"})

# Session deletion
rate(s5s8_inbound_messages_total{message_type="delete_session_request"})

# Session duration (p95)
histogram_quantile(0.95,
  rate(s5s8_inbound_handling_duration_bucket{request_message_type="create_session_request", duration_bucket="5m"})
)
```

## Queries

### Queries

#### Queries

1. **IP** - IP address
2. **PCRF** - PCRF Gx
3. **PGW-U** - PGW-U PFCP
4. **PCRF** - PCRF

#### Queries

```
# IP
curl http://pgw:9090/metrics | grep address_registry_count

# PCRF
# Gx

# PGW-U
# PFCP
```

## 設定/項目

項目

- 項目1
- 項目2
- 項目3

項目

- 項目1
- 項目2
- 項目3

項目

```
# 0mniPGW (項目)
```

```
# 項目
```

## UE 項目

項目

- UE 項目
- 項目

項目

| 項目 | 項目                | 項目        |
|----|-------------------|-----------|
| 項目 | PCRF 項目 (IMSI 項目) | 項目        |
| 項目 | IP 項目             | 項目 IP 項目  |
| 項目 | PCRF/PGW-U 項目     | 項目        |
| 項目 | 項目 APN            | 項目 APN 項目 |

---

□□□□

□□□□

□□□□□□□□

□□□□□□□10,000

□□□□□□□□~10KB RAM

□□□ RAM~100MB

Erlang VM □□□

- □□□□□262,144 (□□)
- □□□□□□□□□□□□

□□□□

□□□□□□□□

1. □□□□□□□□□□
2. □□□□□□□□□□□□□□
3. □□□□□□□□□□□□

□□□□

□□□□□

- □□□□□□□ (□□□□□□□□)
  - □□□□□□□□□□□□□□□ (□□)
  - DNS/□□□□□□□□□□□□
-

□□□□□□

□□□□□□□□

□□□□ PDN □□□□□□□□

**UE** □□

- IMSI: "310260123456789" (□□□□)
- MSISDN: "14155551234" (□□□□)
- MEI/IMEI: □□□□

**PDN** □□□□

- APN: "internet" (□□□□)
- UE IP □□: 100.64.1.42 (□□□ IP)
- PDN □□: IPv4, IPv6 □ IPv4v6

□□□□□

- □□ ID: □□□□□□□□
- □□□□ EBI: EPS □□□□□ (□□□ 5)

**QoS** □□

- APN-AMBR: □□□□□□□
  - □□□100 Mbps
  - □□□50 Mbps

□□□□□

- PDRs (□□□□□□□): □□□□□
- FARs (□□□□□□): □□/□□□□
- QERs (QoS □□□□): □□□□
- BAR (□□□□□□): □□□□

□□□□□□

- S5/S8 **接口**: **接口**/接口 TEIDs, SGW-C **接口**
- Sxb **接口**: **接口**/接口 SEIDs, PGW-U **接口**
- Gx **接口**: Diameter **接口**-ID, **接口**

# Web UI - **接口**

OmniPGW **接口** **Web UI** **接口**

## UE **接口**

**接口**: `http://<omnipgw-ip>:<web-port>/ue_search`

**接口**: **接口** UE **接口**

**接口**

### 1. **接口** **接口**

- **IMSI** (**接口**“310170123456789”)
- **MSISDN** (**接口**)

- **IP** 地址 (例如“100.64.1.42”)

## 2. 网络

- 网络地址
- 网络地址
- 网络地址

## 3. 网络地址 网络地址

### a) 网络

- 网络地址
- IMSI, MSISDN, UE IP 地址
- APN, RAT 类型
- PGW TEID, SGW TEID

### b) 网络地址 网络地址

- **TAC** (网络地址) - UE 网络地址
- **ID (ECI)** - E-UTRAN 网络地址
- **ECGI** - E-UTRAN 网络地址 (PLMN + ECI)
- **MCC/MNC** - 网络地址 / 网络地址

网络地址 网络地址 OpenCellID 网络地址

- 网络地址 (网络/网络)
- 网络 Google 网络地址
- UE 网络地址

网络地址 网络地址 网络地址

### c) 网络 网络地址 QoS 网络

网络:

- EBI (EPS 网络地址)
- QCI (QoS 网络地址)
- 网络地址

- APN-AMBR (UL/DL)

QoS (QoS) 1

- EBI, QCI, 5QI
- MBR UL/DL (UL/DL)
- GBR UL/DL (UL/DL)

**d) Gy (Gy 1)**

- Gy 1 ID
- 1, 1
- 1

**e) Gx (Gx 1)**

- Gx 1 ID
- PCRF 1/1
- CC 1
- 1 (1 PCC 1)

**f) 1**

- 1
- 1/1/1

□□□

- □□□□□□□□
- □□□□□□
- □□□□□ IP □□
- □□□□□□

## PGW □□□□

□□: `http://<omnipgw-ip>:<web-port>/pgw_sessions`

例: 3GPP PDN

例

### 1. 例

- 3GPP (2 例)
- 3GPP
- 3GPP - 3GPP

### 2. 例 例例例

- **IMSI** - 例
- **UE IP** - 例 IP
- **SGW TEID** - 例 SGW S5/S8 ID
- **PGW TEID** - 例 OmniPGW S5/S8 ID
- **APN** - 例

### 3. 例 例例例例例

- IMSI (例“310260”)
- UE IP (例“100.64”)
- MSISDN / 例

- APN □□

**4.** □□□□□□ □□□□□□□□□□□□□□□□

- IMSI, MSISDN, IMEI
- RAT type, MCC/MNC
- QoS type (AMBR type/maximum bit rate)
- TEIDs (TEIDs)
- ID
- (type)

|  |  |  |  |  |  |
|--|--|--|--|--|--|
|  |  |  |  |  |  |
|--|--|--|--|--|--|

```
❏❏: http://<omnipgw-ip>:<web-port>/topology
```

00: 0000000000000000

000

1. 00000

- 0000000000
- 00 PGW-C (0000) 00
- 000 HSS (00000000) 000
- 000000000

2. 0000

- 0000 (+/-)
- 000000
- 00000000000
- 000000 (00 = 00, 00 = 00)

3. 0000

- 0000000000
- 0000
- 000000000

□□□

- □□□□□□□□□□
- □□□□□□
- □□□□□□
- □□□□□□□□

□□□□□□□□□□

□□: `http://<omnipgw-ip>:<web-port>/session_history`

□□: □□□□□□□□□□□□□□

□□□

## 1. □□□□

- □□□□□□□□ (□□□□, □□□□, □□□□, □)
- □□□□□□ (□□□□ / □□□□)
- □ IMSI, MSISDN, IP □□□□ TEID □□

## 2. □□□□

- 透過 CSV 匯入
- 透過 API 匯入
- 透過網頁匯入

3. 設定

- 設定
- 設定
- 設定
- 設定

設定

- 設定
- 設定
- 設定
- 設定
- 設定

設定

設定

1. 設定
2. 透過 IMSI 設定 Web UI
3. 設定 UE 的 IP
4. 設定 QoS
5. 設定

設定

- 設定
- 設定
- 設定 APN

設定

- `ssh -p 22 root@<ip>`
- `ssh -p 22 root@<ip> SSH/IEx`
- `ssh -p 22 root@<ip> SGW PGW TEIDs`
- `ssh -p 22 root@<ip> PCRF AMBR`

## Network Settings

- `ssh -p 22 root@<ip> (root@<ip>)`
- `ssh -p 22 root@<ip>`
- `ssh -p 22 root@<ip> (root@<ip> Mbps root@<ip> bps)`
- `ssh -p 22 root@<ip>`
- `ssh -p 22 root@<ip>`

## OpenCellID

OmniPGW `ssh -p 22 root@<ip> OpenCellID` `ssh -p 22 root@<ip> UE` `ssh -p 22 root@<ip>`

## UE

`ssh -p 22 root@<ip> UE` `ssh -p 22 root@<ip>`

- `ssh -p 22 root@<ip> (root@<ip>)`
- `ssh -p 22 root@<ip> Google`
- `ssh -p 22 root@<ip>`
- `ssh -p 22 root@<ip>`

## UE

`ssh -p 22 root@<ip> http://<omnipgw-ip>:<web-port>/cell_towers` `ssh -p 22 root@<ip> "ssh -p 22 root@<ip>"`

## UE

- `ssh -p 22 root@<ip> OpenCellID.org`

- 使用 SQLite
- 数据集 (约 10-15 万)
- 使用 Web 应用程序
- 使用 Python 进行数据分析和可视化

数据集 使用 Python 进行数据分析和可视化“数据集”数据集

## 数据集

### 数据集

- SQLite DB: `priv/cell_towers.db`
- CSV 文件 (约 10-15 万): `priv/data/cell_towers.csv.gz`
- 数据集 MCC, MNC, LAC, CellID 数据集

### 数据集

- 使用 OpenCellID.org 数据集 107 MB
- 数据集 10-15 万

### 数据集

- 数据集 (<1ms)
- 数据集
- 使用 UE 数据集

## 数据集

### 数据集

### UE 数据集

- 数据集
- 使用 Google 数据集
- 数据集

### Web UI

- 電話番号 (国番号, 地域番号, 番号)
- 電話番号 - 電話番号 OpenCellID 番号
- 電話番号
- MCC, MNC, LAC, Cell ID 番号
- 電話番号
- 電話番号電話番号電話番号

## 電話番号

- 電話番号電話番号
- 電話番号
- 電話番号電話番号
- 電話番号電話番号

## 電話番号

OpenCellID 電話番号電話番号電話番号

## 電話番号

1. 番号 `http://<omnipgw-ip>:<web-port>/cell_towers`
2. 番号 “電話番号” 番号
3. 電話番号電話番号
4. 番号 10-15 電話番号電話番号/番号
5. 電話番号電話番号電話番号

## 電話番号 電話番号

番号 OpenCellID 電話番号電話番号電話番号電話番号電話番号電話番号

## 電話番号

## 電話番号

- 番号 OpenCellID.org 電話番号
- 電話番号 HTTPS 番号
- 電話番号 (番号 ~200 MB 電話番号)

- 00000000000000000000
- OpenCellID 00000000 - 0000000000
- 00 Web UI 0000000000000000

## 00000000

- 00 `priv/` 000000000000
- 0000000000 (~150 MB 000000)
- 000000000000 `priv/` 000/0000

## 000000

- 000000000000
- OpenCellID 0000000000000000
- 0000000000000000

## 000000

- 00000 JavaScript 00000000
- 00 Google 000000
- 0000000000



## 000000

- **PFCP** 00 - 00000000, PDRs, FARs, QERs, URRs
- **UE IP** 00 - IP 0000, APN 000
- **PCO** 00 - 000 UE 0 DNS, P-CSCF, MTU 00
- 0000 - UPF 00, 000000

## 000000

- **Diameter Gx** 00 - PCRF 0000, PCC 00, QoS 00
- **Diameter Gy** 00 - OCS 0000, 0000

- **CDR** -

- **S5/S8** - GTP-C , SGW-C
- **QoS** - QoS

- - , ,
- **P-CSCF** - IMS

**OmniPGW** - *Omnitouch*

# OmniPGW

OmniPGW

OmniPGW

## OmniPGW

- OmniPGW
- OmniPGW
- OmniPGW
- PFCP /
- Diameter (Gx/Gy)
- IP
- 
- 
- 
- 

## OmniPGW

OmniPGW

- 
- 
- 
- 
- 

## OmniPGW

- Prometheus

- **API** - **API**

API

## Web UI

API `http://<omnipgw_ip>:4000`

API

- **/pgw\_sessions** - IMSI, IP, MSISDN, APN
- **/diameter** - Diameter Gx, PCRF, Gy, OCS
- **/pfcps** - PFCP PGW-U
- **/logs** -

## Prometheus

API `http://<omnipgw_ip>:9090/metrics`

API

- `teid_registry_count` -
- `address_registry_count` - UE IP
- `sxb_inbound_errors_total` - PFCP
- `gx_inbound_errors_total` - Diameter Gx
- `gy_inbound_errors_total` - Diameter Gy

API **API**

API

Web UI **/logs**

API

- "create\_session\_request" - 0000
  - "Credit Control" - Gx/Gy 00
  - "PFCP Session" - 000000
  - "error" 0 "ERROR" - 0000
  - "timeout" - 0000
- 

00000000

00000000000000000000 "000000"

000

- SGW-C 000000000000 "000000" (73)
- 000000000000
- 0000000000
- 000 [PGW-C] 0000000000 - 000000

Wireshark 0000000000000000 "000000"

000000

- 000000 OmniPGW 000

- 000000000000

0000

### 1. 000000000000

```
license_status
```

- 00 0 00000000

### 2. 0000000000000000

- 00 "license" 0 "License"
- 00 "0000000000000000" 00

### 3. 0000000000000000

- 00 config/runtime.exs 0 :license\_client 0000 URL
- 0000 https://localhost:10443/api

000000

### 1. 000000000000000000

```
curl -k https://<license_server_ip>:10443/api/status
```

### 2. 00 config/runtime.exs 0000000000

```
config :license_client,  
  license_server_api_urls:  
  ["https://<license_server_ip>:10443/api"],  
  licensee: "00000000"
```

### 3. 000000000000000000

- 000000 omnipgwc
- 00 Omnitouch 00000000

#### 4. **OmniPGW**

□□□

- □□ `license_status` □□□□□□□□
  - □□□□□□□□□□□□
  - □□□□□□□□□□□□
- 

□□□□□□□□□□□□□□□□□

□□□

- SGW-C □□□□□□□□□□□□
- □□□□□□ PDN □□
- □□□ `s5s8_inbound_errors_total` □□

□□□□□

1. IP □□□
2. PCRF (Gx) □□□□□□□□□□
3. PGW-U (PFCP) □□□
4. □□□ APN □□

□□□

1. □□ **IP** □□□□□

`address_registry_count`

- □□□□□□□□□□□□□□□□

2. □□ **PCRF** □□□□

- Web UI → **/diameter** □□
- □□ PCRF □□□□ = "disconnected"
- □□□□□ "Credit Control Answer" □□□□□

### 3. PCFP

- Web UI → **/pfcpeers**
- "Association: DOWN"
- `pfcpeer_associated = 0`

### 4. APN

- `config/runtime.exs` `ue.apn_map`
- APN

### IP

1. Web UI → **/pgwsessions**
2. `config/runtime.exs` IP

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => "10.0.0.0/23" # /24 /23
    }
  }
```

### 3. OmniPGW

4. `curl http://<ip>:9090/metrics | grep address_registry_count`

### PCRF

1. `ping <pcrf_ip>`
2. PCRF Diameter `telnet <pcrf_ip> 3868`
3. `config/runtime.exs` Diameter
4. OmniPGW
5. Web UI → **/diameter** "connected"

### PCFP

- **PCFP /**

□□□

- □□ IP □□□□□□ 80% □□□
  - □□ PCRF □□□□□□ Diameter □□□□
  - □□□□□□□□□
- 

□□□□□□□□□□□□

□□□

- □□□ Web UI □□□□□□□
- □□□□□□□□□□□□□□□□
- □□□□□□□□□□□

□□□□□

1. PCFP □□□□□□□□ S5/S8 □□□□□
2. PCRF CCR-Initial □□
3. □□□□□□□□□□□□□□
4. □□□□□□□□□□□

□□□

1. □ **Web UI** □□□□□□

- **/pgw\_sessions** → □ IMSI □□
- □□ `pcfcp_seid` □□□□□□□□□□□□□□□□ PCFP □□□
- □□ `gx_session_id` □□□□□□□□□□□□□□□□ Gx □□□

2. □□□□□□ **IMSI**□

- □ IMSI □□□□□
- □□ "Session Establishment Request"□□PCFP□
- □□ "Credit Control Request"□□Gx□
- □□□□□□□□□□

3. □□□□□

```
# 1 TEID 1000 PFCP 1000000
teid_registry_count - seid_registry_count

# 1 TEID 1000 Gx 1000000
teid_registry_count - session_id_registry_count
```

1000000

### 1. 1000 **PFCP** 1000000

- 1000 PGW-U 1000000000
- 1000 PFCP 1000 Web UI → **/pfcpeers**
- 1 SGW-C 1000000000000

### 2. 1000 **Gx** 1000000

- 1000 PCRF 1000 `histogram_quantile(0.95, rate(gx_outbound_transaction_duration_bucket[5m]))`
- 10000000 `config/runtime.exs` 1000 Gx 1000
- 1000000000000

### 3. 1000000000000

- 10000000 OmniPGW 1000000000
- 100000000 `teid_registry_count` 1000000

1000

- 1000 PFCP 1 Gx 1000000
  - 1000000/1000000000000000000
  - 1000000000000000
-

# PFCP / 設定

## PFCP 設定

確認

- Web UI → **/pfcpeer** 確認 "Association: DOWN"
- 確認
- 確認 `pfcpeer_associated = 0`
- 確認 "PFCP 設定" 設定 "設定"

確認

1. PGW-U 確認
2. PGW-U 確認
3. PFCP 確認 IP
4. 確認 UDP 8805

確認

1. 確認

```
ping <pgw_u_ip>  
nc -u -v <pgw_u_ip> 8805
```

2. 確認 **PFCP** 設定

- 確認 `config/runtime.exs` 確認 `upf.peer_list`
- 確認 IP 確認 ID 確認 PGW-U 確認

3. 確認 **PGW-U** 設定

- 確認 PGW-U 確認
- 確認 PGW-U 確認 `systemctl status omnipgw_u` 確認

4. 確認

```
# 設定
pfcg_consecutive_heartbeat_failures

# PFCP 設定
rate(sxb_inbound_errors_total[5m])
```

手順

### 1. 設定確認

- 設定 `traceroute <pgw_u_ip>`
- 設定 `UDP 8805` 設定
- 設定 `pgw_u_ip`

### 2. PGW-U 設定

- PGW-U 設定
- 30 秒間隔
- Web UI → **/pfcg\_peers** 設定 "Association: UP"

### 3. 監視設定

- 設定 `config/runtime.exs` PFCP 設定
- OmniPGW
- 設定

確認

- 設定 `pfcg_peer_associated` 設定
- 設定 `pfcg_consecutive_heartbeat_failures` 設定 > 2
- PGW-U 設定
- PFCP 設定/監視設定

---

## PGW-U PFCP 設定

設定

- 00000000
- QoS 00000000 PCRF RAR000
- 000"00000000"
- 000

```
sxb_inbound_errors_total{message_type="session_modification_respon
se"} 00
```

00000

1. 000 PFCP 000PDR/FAR/QER 000
2. PGW-U 0000
3. 00 ID 00
4. PGW-U 0000

000

1. 00000
  - 00 "0000" 0 SEID
  - 00 PFCP 0000000000
  - 00000"00 ID 000"0"0000"
2. 00 **PGW-U** 000
  - 00 PFCP 0000
  - 00000000CPU0000
3. 0 **Web UI** 00000000
  - **/pgw\_sessions** → 0 IMSI 0000
  - 00 pdr\_map 0 far\_map 0 qer\_map 000000
  - 0000 ID

00000

1. 00000000
  - 000000000000
  - 00000000000000 UE 0000

## 2. PGW-U

- PGW-U PDR
- PGW-U
- PGW-U

## 3.

- Web UI
- PFCP
- 

- PGW-U
- 
- `sxb_inbound_errors_total`

# Diameter (Gx/Gy)

## PCRF Gx

- Web UI → **diameter** PCRF "disconnected"
- QoS QCI=5
- "Diameter" "CER/CEA"

- PCRF
- PCRF
- Diameter Origin-Host Realm
- TCP 3868

## 1. ping & telnet

```
ping <pcrf_ip>
telnet <pcrf_ip> 3868
```

## 2. 設定 Diameter

- config/runtime.exs 設定 diameter.peer\_list
- host, realm, ip 設定 PCRF 設定
- origin\_host 設定 PCRF 設定

## 3. 設定 PCRF

- 設定 PGW-C 設定 CER 設定
- 設定

## 4. 設定

```
# Diameter 設定
diameter_peer_connected{peer="<pcrf_host>"}
```

## 確認

### 1. ping & telnet

- 設定 PCRF 設定
- 設定 TCP 3868 設定
- 設定 nc -v <pcrf\_ip> 3868

### 2. 設定 PCRF

- 設定 PCRF 設定
- 設定 30 設定
- 設定 Web UI → /diameter 設定

### 3. 設定

- 設定 config/runtime.exs 設定 Diameter 設定

```

config :pgw_c,
  diameter: %{
    host: "pgw-c.epc.mnc999.mcc999.3gppnetwork.org", # 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
      }
    ]
  }
}

```

- OmniPGW
- PCRF

PCRF

- Diameter PCRF
- PCRF
- Diameter

## CCR/CCA Gx

PCRF

- > 5
- "QoS"
- `gx_outbound_transaction_duration` > 5s
- QoS

PCRF

1. PCRF
2. PCRF
3. QoS

#### 4. PCRF 設定

設定

##### 1. Gx 設定

```
# P95 設定
histogram_quantile(0.95,
rate(gx_outbound_transaction_duration_bucket[5m]))

# P99 設定
histogram_quantile(0.99,
rate(gx_outbound_transaction_duration_bucket[5m]))
```

##### 2. PCRF 設定

- PCRF 設定
- CPU使用率
- PCRF 設定

##### 3. 設定

```
ping -c 100 <pcrf_ip> | tail -1 # 設定
```

##### 4. 設定

- CCR/CCA 設定 "設定"
- "CCR" と "CCA" 設定

設定

##### 1. PCRF 設定

- PCRF設定
- CCR 設定
- PCRF 設定/設定

##### 2. 設定

- 確認する
- 確認 PGW-C と PCRF 確認する

### 3. 確認する

- 確認 `config/runtime.exs`

```
config :pgw_c,
  diameter: %{
    transaction_timeout_ms: 10000 # 確認 5000 確認
  }
```

- 確認 OmniPGW
- 確認 確認する

確認

- 確認 Gx 確認する > 1s確認 > 5s
- 確認する PCRF 確認
- 確認 PCRF 確認

## 確認 OCS 確認 Gy

確認

- Web UI → **/diameter** 確認 OCS 確認 "disconnected"
- 確認
- 確認 "Gy 確認"

確認

確認 PCRF 確認 Gy 確認

確認

- 確認 TCP 3868確認 Gx 確認
- 確認
- 確認 `diameter.peer_list` 確認 OCS 確認

# IP 📄📄📄

## 📄📄IP 📄📄

📄📄

- 📄📄📄📄📄📄📄 "📄📄📄📄"
- 📄📄 `address_registry_count` 📄📄📄📄📄
- Web UI → **/pgw\_sessions** 📄📄📄📄📄
- 📄📄"IP 📄📄📄📄📄"

📄📄📄📄

1. 📄📄📄📄📄📄
2. 📄📄📄 IP📄📄📄📄📄
3. 📄📄📄📄📄📄
4. IP 📄📄📄

📄📄

1. 📄📄📄📄📄

```
# 📄 /24 📄📄254 📄 IP📄  
(address_registry_count / 254) * 100
```

2. 📄📄📄📄📄📄

- 📄 `config/runtime.exs` 📄 `ue.subnet_map`
- 📄📄"10.0.0.0/24" = 254 📄📄 IP

3. 📄📄📄📄📄 **IP** 📄📄

```
# [] [] [] []
teid_registry_count
address_registry_count
```

#### 4. [] [] [] [] [] []

- Web UI → **/pgw\_sessions**
- [] [] [] [] [] [] [] []
- [] [] [] [] [] [] [] [] [] [] [] []

[] [] [] []

[] [] [] [] [] [] [] []

#### 1. [] [] `config/runtime.exs` []

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => "10.0.0.0/22" # 1022 [] IP [] [] [] /24 = 254 []
    }
  }
  IP []
}
```

#### 2. [] [] OmniPGW

#### 3. [] [] [] [] [] [] [] [] [] []

[] [] [] [] [] []

1. [] Web UI [] [] [] [] [] [] [] []
2. [] SGW-C [] [] [] [] [] [] [] [] [] []
3. [] PCRF/SGW [] [] [] [] [] [] [] []
4. [] [] `address_registry_count` [] [] [] [] [] [] [] [] [] []

[] [] []

- [] [] IP [] [] [] [] [] [] [] []
  - [] [] > 70%
  - [] [] > 85%

- 0000000000
- 0000000000
- 000000

---

## 00000000 IP 00

000

- UE 00 IP 0000
- 00000 "IP 000"
- Web UI 0000000000 IP 00

00000

1. 0000000000
2. 0000000000
3. 000000

000

1. 0 Web UI 000 IP
  - /pgw\_sessions → 0 IP 0000
  - 0000000 IMSI 00000 IP
2. 00000
  - 00 IP 00
  - 00 "IP 00" 00

00000

1. 0000000000
  - 0000000 IP 000 IMSI
2. 00000000
  - 0 SGW-C 0000000000000000 IMSI

- 3GPP TS 23.002

### 3. UE 3GPP

- UE 3GPP
- 3GPP IP

### 4. 3GPP

- 3GPP OmniPGW 3GPP IP
- 3GPP 3GPP

3GPP

- 3GPP 3GPP
  - 3GPP 3GPP
-

□□□□

## □□ Prometheus □□

```
# □□□□
teid_registry_count

# □□□□□□□□□□
rate(s5s8_inbound_messages_total{message_type="create_session_request

# IP □□□□□□□□ /24 □□□
(address_registry_count / 254) * 100

# P95 □□□□□□
histogram_quantile(0.95,
rate(s5s8_inbound_handling_duration_bucket{request_message_type="crea
[5m]))

# □□□
rate(s5s8_inbound_errors_total[5m])

# PCRF □□
histogram_quantile(0.95, rate(gx_outbound_transaction_duration_bucket

# PFCP □□□□
pfcf_peer_associated
```

Web UI

| 項目               | 値          |
|------------------|------------|
| IMSI             | XXXXXXXXXX |
| "create_session" | XXXXXXXX   |
| "delete_session" | XXXXXXXX   |
| "Credit Control" | Gx PCRF 値  |
| "PFCP Session"   | XXXXXXXX   |
| "error"          | XXXXXXXX   |
| "timeout"        | XXXX       |
| "Association"    | PFCP XXXX  |

## コマンド

```
# サービスの状態を確認
systemctl status omnipgw_c

# Web UI へのアクセス
curl http://<omnipgw_ip>:4000

# サービスのメトリクスを確認
curl http://<omnipgw_ip>:9090/metrics

# サービスのメトリクスを確認 (teid_registry_count を含む)
curl http://<omnipgw_ip>:9090/metrics | grep teid_registry_count

# PFCP サービスのメトリクスを確認
curl http://<omnipgw_ip>:9090/metrics | grep pfcpeer_associated

# IP アドレス登録のメトリクスを確認
curl http://<omnipgw_ip>:9090/metrics | grep address_registry_count
```

## 機能

- **監視** - Prometheus と Grafana を利用して監視
- **ログ** - ログ出力
- **メトリクス** - サービスのメトリクスを出力
- **PFCP サービス** - PFCP サービスを提供
- **Diameter Gx サービス** - Gx サービスを提供
- **Diameter Gy サービス** - Gy サービスを提供
- **QoS サービス** - QoS サービスを提供

## インストール

**OmniPGW** のインストール - *OmniTouch* のインストール

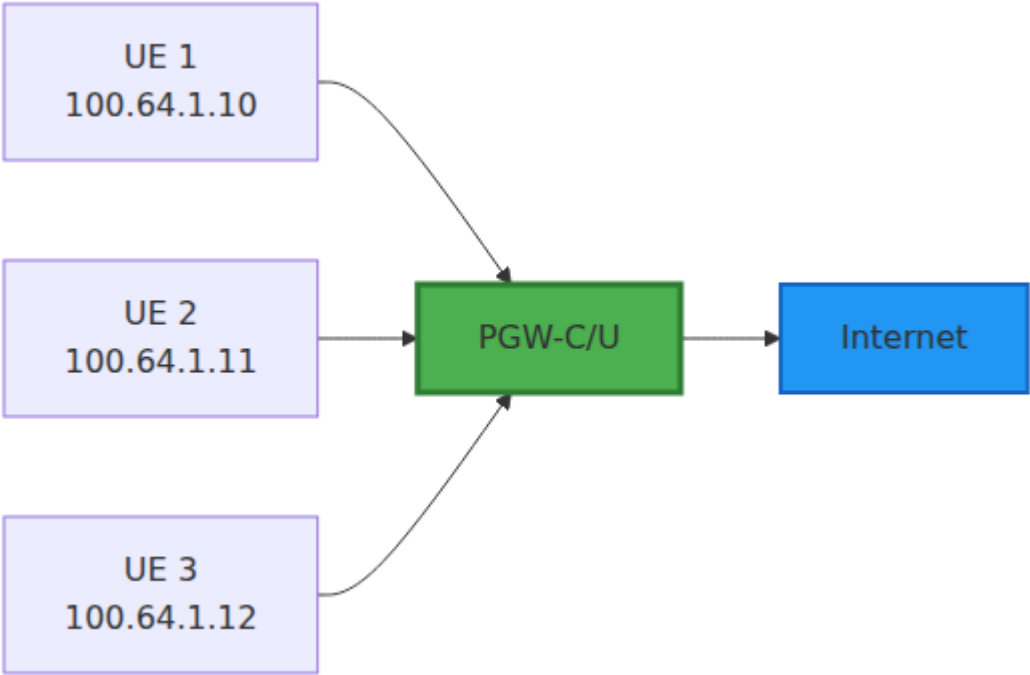
# UE IP

IP

- 1.
2. IP
- 3.
- 4.
- 5.
- 6.
- 7.

PGW-C UE PDN IP

UE IP 할당



UE가 PGW-C에 접속할 때 IP 할당

- IP 할당
- IP 할당
- IP 할당
- PDN 할당

UE IP

| IP     | 할당 | 할당                 |
|--------|----|--------------------|
| IPv4   | 할당 | 할당 IPv4 할당         |
| IPv6   | 할당 | IPv6 할당            |
| IPv4v6 | 할당 | 할당 IPv4 할당 IPv6 할당 |

# IP 設定

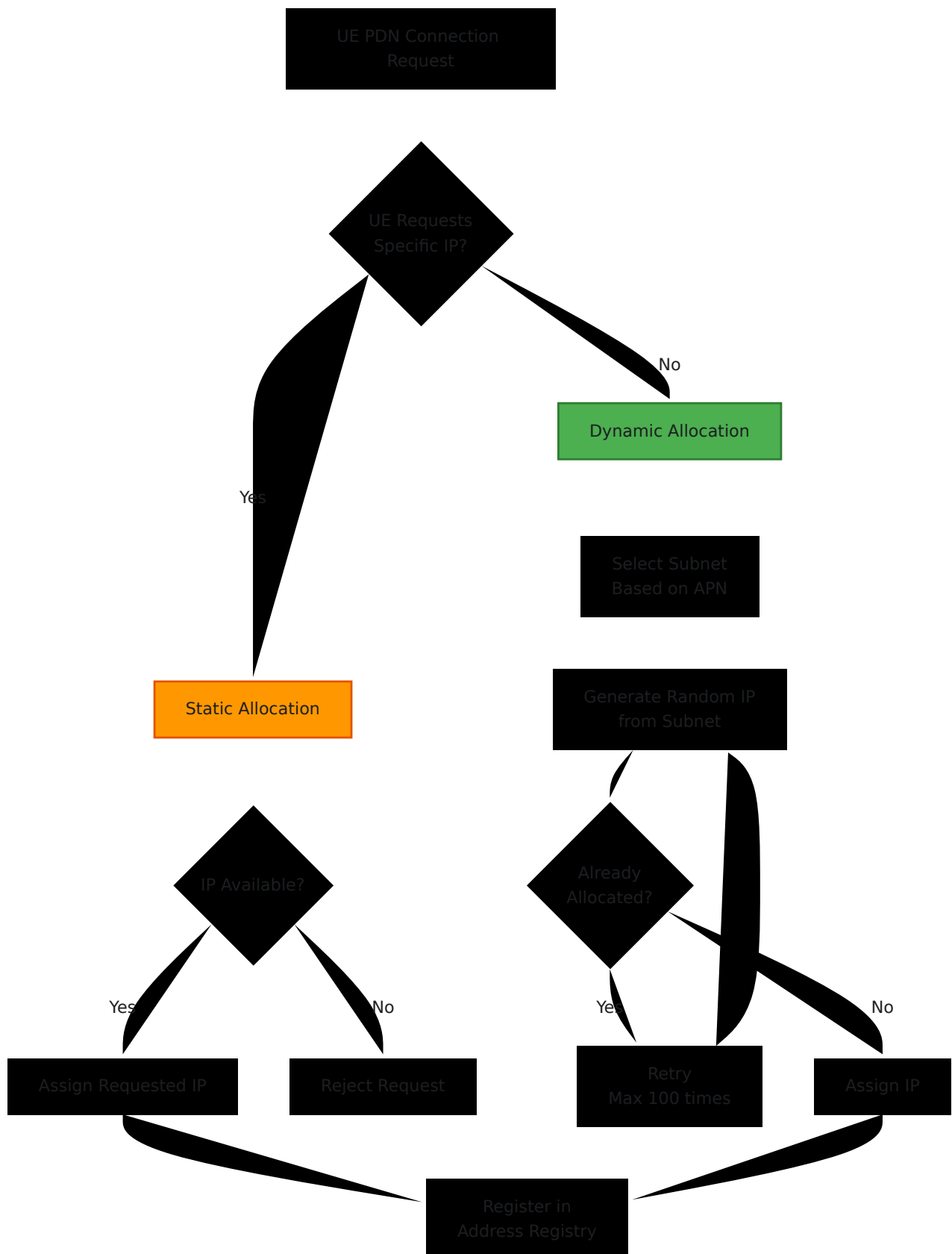
## PDN 設定

この UE の PDN 設定を定義する **PDN 設定**

| PDN 設定 | 設定      | 説明              |
|--------|---------|-----------------|
| IPv4   | IPv4 設定 | IPv4 設定         |
| IPv6   | IPv6 設定 | IPv6 設定/64      |
| IPv4v6 | 設定      | IPv4 設定 IPv6 設定 |

設定

PGW-C 設定 IP 設定



# 1. IP Allocation

- PGW-C allocates IP
- IP Allocation

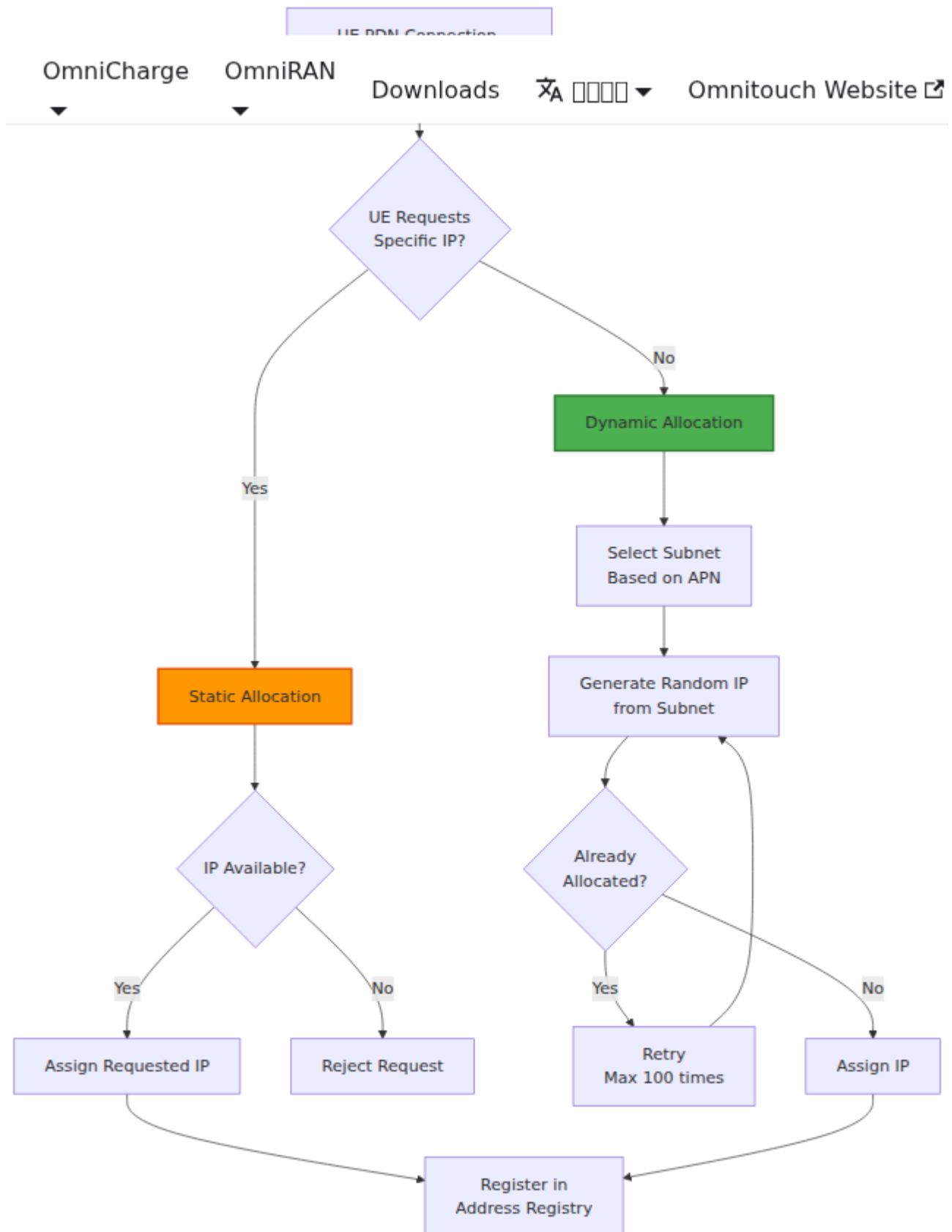
- 0000000000

## 2. 00000

- UE の GTP-C 00000000 IP
- PGW-C 00000
- 000000 IP 0000000000

## 00 **APN** 00000

000 **APN**0000000 0000000 IP 00



□□□

- □□□□ - □□ APN □□□□□□□□
- □□□□□□ - □□ APN □□□□□□□□

- 0000 - 0000000000000000
- 00 - 000000000000

000000

000000 000000 IP

| 00   | 00                  |
|------|---------------------|
| 00   | 00 UE IP → 0000 PID |
| 00   | 00 UE IP 0000       |
| 00   | 00000000 IP         |
| 0000 | 00000000            |

00

00000

00 config/runtime.exs

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      # APN "internet"
      "internet" => [
        "100.64.1.0/24",    # 254 IP
        "100.64.2.0/24"    # 254 IP
      ],

      # APN "ims"
      "ims" => [
        "100.64.10.0/24"
      ],

      # Default APN
      default: [
        "42.42.42.0/24"
      ]
    }
  }
}
```

IP

**CIDR** <network>/<prefix\_length>

| CIDR | IP    |                             |
|------|-------|-----------------------------|
| /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 |

- 100.64.1.0/24
- 100.64.1.255/24
- PGW-C 1 <network> + 1 <broadcast> - 1

## APN

```
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"
      ]
    }
  }
}
```

- PGW-C
- 
- 

- 
- 
-

## 配置示例

```
config :pgw_c,  
  ue: %{  
    subnet_map: %{  
      # 互联网  
      "internet" => [  
        "100.64.0.0/20"      # 4094 互联网 IP  
      ],  
  
      # IMS  
      "ims" => [  
        "100.64.16.0/22"    # 1022 IMS IP  
      ],  
  
      # APN  
      "enterprise.corp" => [  
        "10.100.0.0/16"     # 65534 APN IP  
      ],  
  
      # IoT  
      "iot.m2m" => [  
        "100.64.20.0/22"    # 1022 IoT IP  
      ],  
  
      # 默认  
      default: [  
        "42.42.42.0/24"     # 254 默认 APN IP  
      ]  
    }  
  }  
}
```

## IPv6 配置

```
config :pgw_c,  
  ue: %{  
    subnet_map: %{  
      "internet" => [  
        # IPv4 网段  
        "100.64.1.0/24"  
      ],  
      "internet.ipv6" => [  
        # IPv6 网段  
        "2001:db8:1::/48"  
      ],  
      default: [  
        "42.42.42.0/24"  
      ]  
    }  
  }  
}
```

## IPv6 地址规划

- UE 地址 /64 网段
- 每个 UE 地址 IP 地址
- 每个 UE 地址 `2001:db8:1:a::/64`

## IPv4v6 配置

```
config :pgw_c,  
  ue: %{  
    subnet_map: %{  
      "internet" => [  
        "100.64.1.0/24",      # IPv4 网段  
        "2001:db8:1::/48"    # IPv6 网段  
      ],  
      default: []  
    }  
  }  
}
```

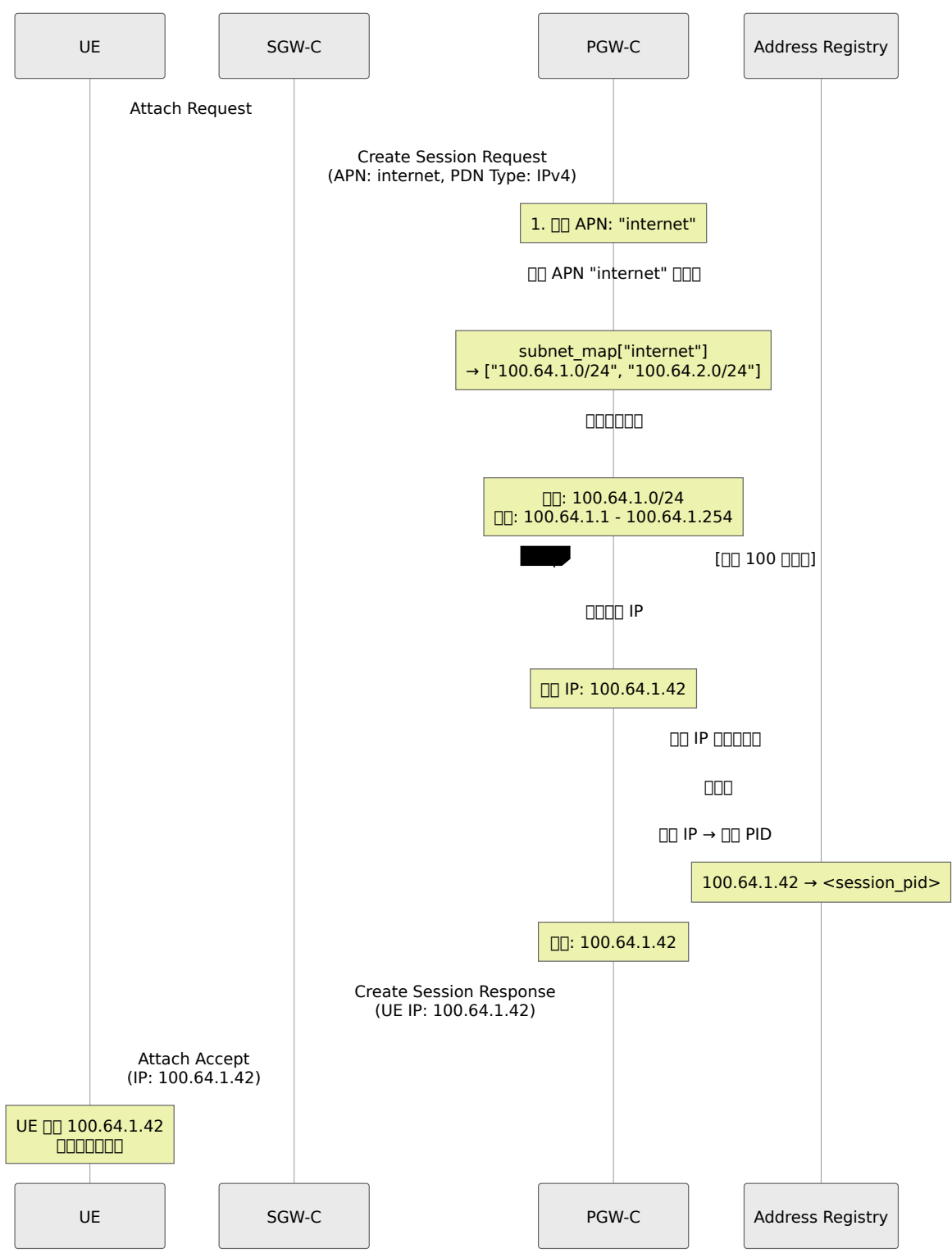
配置

- UE 通过 PDN 连接IPv4v6
  - PGW-C 支持 IPv4 和 IPv6 地址
  - 网络地址转换
- 

网络

IP 地址分配由 PGW-C 通过 S5/S8 接口与 GTP-C 接口通过 S5/S8 接口与网络地址转换

IPv4



IPv4

IPv4

1. 選擇合適的 APN 服務商
2. 選擇合適的服務商
3. **IP** 地址 選擇合適的 IP 地址
4. 選擇合適的 IP 地址
5. 選擇合適的服務商 100 個合適的 IP
6. 選擇合適的 IP 地址

選擇合適的

- 選擇 **100** 個合適的服務商
- 選擇合適的 IP 地址
- 選擇合適的服務商
- 選擇合適的服務商 APN 服務商

選擇合適的

選擇合適的服務商 IP



- 16 bits
- 8 bits
- 8 bits IP 00
- 8 bits IP 000
- 16 bits IP

## 設定方法

UE の APN

設定

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

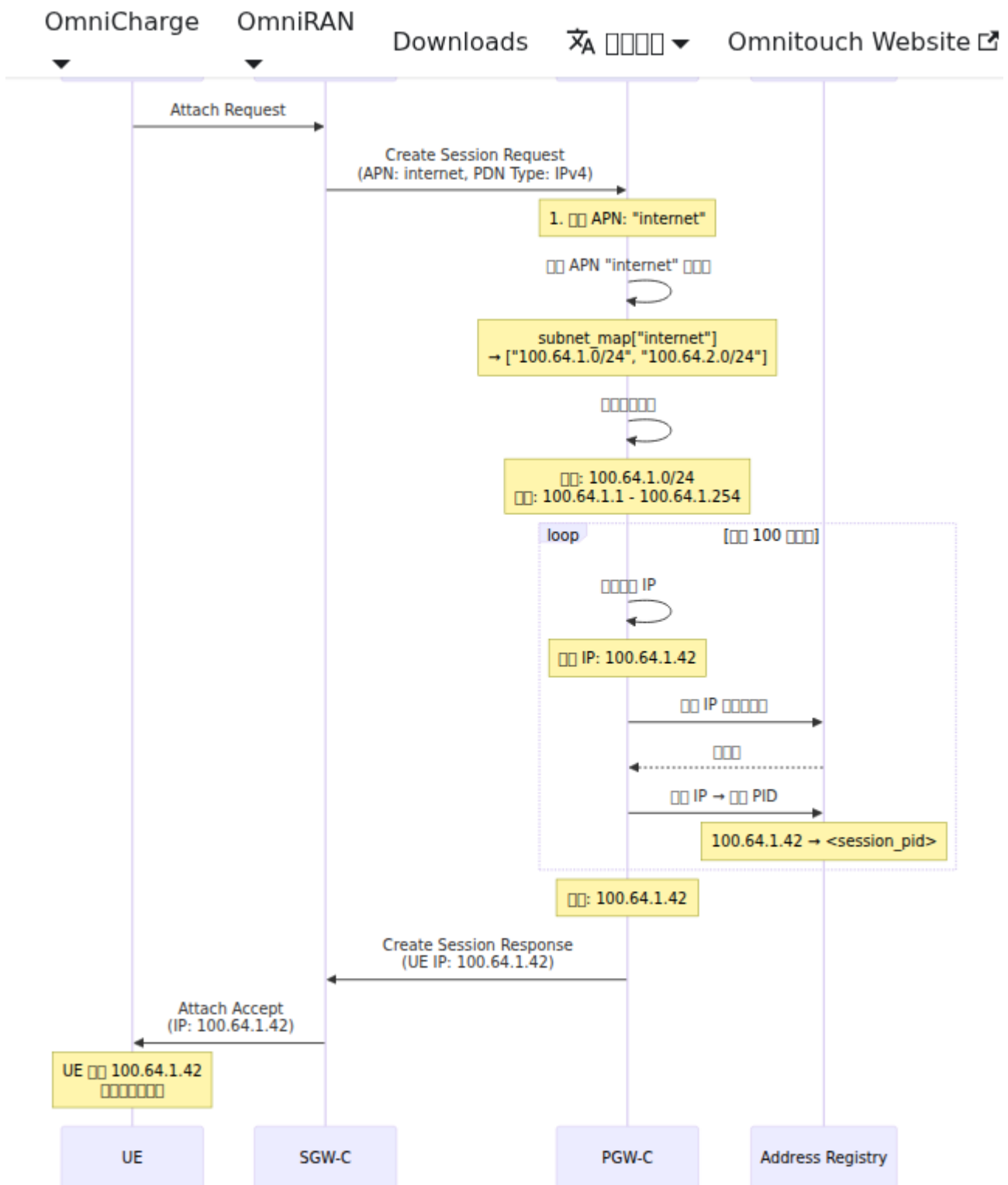
例

- UE の APN: "unknown.apn"
- subnet\_map の "unknown.apn"
- 設定
- 42.42.42.0/24 の IP

設定

1. APN の設定
2. default の設定
3. 設定

# Sequence Diagram



## Notes

- IP allocation process
- IP address range
- IP address

## 問題

### 背景

PGW-C において IP 不足

例: 100.64.1.0/24 (254 個の IP)

制限: 254 個の IP

アドレス → 不足

原因

1. PGW-C 側 100 個のアドレス
2. 不足の IP 不足
3. ログ: `{:error, :ue_ip_address_allocation_failed}`
4. 不足
5. SGW-C 側

原因

```
# 不足
address_registry_count / total_pool_size > 0.8 # 80% 不足

# 不足
"internet" => [
  "100.64.1.0/24",
  "100.64.2.0/24", # 不足
  "100.64.3.0/24"
]
```

## IP 不足

不足の IP

GTP-C 不足

### Create Session Request

└─ IMSI: 310260123456789  
└─ APN: enterprise.corp  
└─ PDN 類型 (IE)  
| └─ PDN 類型: IPv4  
| └─ IPv4 地址: 10.100.0.50 ← UE 固有 IP

## OmniPGW 動作

1. 固有 IP による PDN 類型 IE
2. 固有 IP による IP 類型 APN 固有
3. 固有 IP による IP 固有
4. 固有 IP
  - 固有 IP 固有
  - 固有 IP 固有

## 動作

- 固有 UE 固有 IP 固有
- 固有 IP 固有 - IP 固有
- 固有 IP 固有 - IP 固有

## IPv6 動作

### UE 固有 IPv6

### Create Session Request

└─ PDN 類型: IPv6

## PGW-C 固有 /64 固有

prefix: 2001:db8:1:a::/64

UE prefix

- 2001:db8:1:a::1
- 2001:db8:1:a::2
- ... (18 quintillion)

UE

- UE prefix IP address
- SLAAC
- NAT

UE

UE IPv4v6

Create Session Request

PDN: IPv4v6

PGW-C

IPv4: 100.64.1.42

IPv6: 2001:db8:1:a::/64

UE

- IPv4 address IPv4
- IPv6 address IPv6
- 
- GTP

IP

IP RFC 1918

```
# 配置子网映射
subnet_map: %{
  "internet" => [
    "10.0.0.0/8",
    "172.16.0.0/12",
    "192.168.0.0/16"
  ]
}
```

配置 **PGW-U** 的 **NAT** 规则

配置 **IP** 池

```
# 配置 IP 池
subnet_map: %{
  "internet" => [
    "203.0.113.0/24" # 公网 IP 池
  ]
}
```

配置 **NAT** - 规则

配置

- 配置 **IP** **RFC 6598** 的 **100.64.0.0/10** 公网 NAT
- 配置 **IP**

配置

## Web UI - IP 配置

OmniPGW 的 Web 界面 IP 地址

地址 `http://<omnipgw-ip>:<web-port>/ip_pools`

□□□

### 1. □□□

- □□□□ IP □
- □□□□□□□□
- □□□□ IP
- □□□□□□□□

### 2. □ **APN** □□□ □□□□□□□□

- □□□ - APN □□□□□□□□“default”□“ims.something.else”□“Internet”□
- **APN** □□ - □□□ APN □□□□
- **IP** □□ - □□□□□□□□ CIDR □□□
- □□□ - □□□□□□□□□□□□□□
- □□□□□
  - □□□□□□ IP □□
  - □□□□□□□□□□ IP
  - □□□□□□□□□□ IP

### 3. □□□□

- 2 個
- 個
- 個

個

- 個
- 個
- 個
- 1 APN 個

個

個

```
# 個 IP
address_registry_count

# 個
address_registry_count / <total_pool_size> * 100
```

個

```
個: 100.64.1.0/24 (254 個 IP)
個: 150 個 IP
個: 150 / 254 = 59%
```

📄

```
# 📄📄📄📄
- alert: UEIPPoolUtilizationHigh
  expr: address_registry_count > 200 # 📄📄 /24 📄
  for: 10m
  annotations:
    summary: "UE IP 📄📄📄📄 80%"
    description: "📄📄: {{ $value }} / 254 📄 IP 📄📄"

# 📄📄📄📄
- alert: UEIPPoolExhausted
  expr: address_registry_count >= 254 # 📄📄 /24 📄
  for: 1m
  annotations:
    summary: "UE IP 📄📄 - 📄📄 IP"

# 📄📄📄📄
- alert: UEIPAllocationFailures
  expr: rate(ue_ip_allocation_failures_total[5m]) > 0
  for: 5m
  annotations:
    summary: "UE IP 📄📄📄📄"
```

## Grafana 📄📄

📄📄 1: IP 📄📄📄

```
# 📄📄📄📄📄📄
(address_registry_count / 254) * 100
```

📄📄 2: 📄📄📄📄📄📄 IP

```
# 📄📄📄📄
address_registry_count
```

📄📄 3: 📄📄📄

```
# 计算速率
rate(address_registry_count[5m])
```

图 4: 速率

```
# 计算地址注册表的占比
(254 - address_registry_count) / rate(address_registry_count[1h])
```

图 5

图 1: 地址注册表的 IP 地址

图 6

- 地址注册表的 "地址"
- 地址 "UE IP 地址"

图 7

1. 图 8

```
# 计算速率
curl http://<pgw_c_ip>:42069/metrics | grep
address_registry_count
```

2. 图 9

```
# 例
config :pgw_c,
  ue: %{
    subnet_map: %{
      "internet" => [
        "100.64.1.0/24" # 例 CIDR
      ]
    }
  }
}
```

### 3. APN 例

```
# 例 APN
# 例
subnet_map: %{
  default: ["42.42.42.0/24"]
}
```

例

- 例 例
- 例 PGW-C 例 IP
- 例 `runtime.exs` 例

## 2: IP 例

例

- 例 UE 例 IP
- 例

例

- 例

例

```
# 確認する IP
grep "already_registered" /var/log/pgw_c.log
```

確認

- 確認する IP
- 確認する IP

## 3: 確認 IP

確認

- UE 確認する IP
- APN "internet" の "ims" 確認 IP

確認

- 確認する IP

確認

```
# 確認する APN 確認
subnet_map: %{
  "internet" => [...],      # 確認
  "Internet" => [...],     # 確認 APN!
}
```

確認

- 確認 APN 確認する IP
- 確認する IP

## 4: IPv6 確認

確認

- UE 確認 IPv6 確認

□□□□□

## 1. □□ IPv6 □

```
# □□ IPv6 □  
subnet_map: %  
  "internet" => [  
    "100.64.1.0/24" # □ IPv4  
  ]  
}
```

## 2. □□ IPv6 □

```
# □□□□□□ /48 □□□  
"internet" => [  
  "2001:db8::/128" # □□ - □□□□□  
]
```

□□□□□

```
# □□ IPv6 □  
subnet_map: %  
  "internet" => [  
    "100.64.1.0/24",  
    "2001:db8:1::/48" # IPv6 □  
  ]  
}
```

## □□ 5: □□□□□

□□□

- □□□□□
- `address_registry_count` □□□□□

□□□□□

## 1. □□□□□

```
"internet" => [
  "100.64.1.0/24",    # 1
  "100.64.2.0/24",    # 256 1 IP
  "100.64.3.0/24"     # 256 1 IP
]
```

## 2. 100.64.0.0/22

```
# 100.64.0.0/22
"internet" => [
  "100.64.0.0/22" # 1024 1 IP
]
```

## 3. 100.64.0.0/22

- 100.64.0.0/22
- 100.64.0.0/22

100.64.0.0/22

100.64.0.0/22

100.64.0.0/22

```
100.64.0.0/22 10,000
100.64.0.0/22 30% 3,000 100.64.0.0/22
100.64.0.0/22 50%
100.64.0.0/22 3,000 * 1.5 = 4,500 100.64.0.0/22
```

```
100.64.0.0/20 4,094 100.64.0.0/22 - 100.64.0.0/22
100.64.0.0/19 8,190 100.64.0.0/22 - 100.64.0.0/22
```

100.64.0.0/22

100.64.0.0/22

- 100.64.0.0/10 RFC 6598 - NAT
- 400 IP
- NAT

- IP
- VPN

```
config :pgw_c,
  ue: %{
    subnet_map: %{
      # APN - 
      "internet" => [
        "100.64.0.0/18" # 16,382 IP
      ],

      # IMS - 
      "ims" => [
        "100.64.64.0/22" # 1,022 IP
      ],

      # - 
      "enterprise.corp" => [
        "100.64.68.0/22" # 1,022 IP
      ],

      # IoT - 
      "iot.m2m" => [
        "100.64.72.0/20" # 4,094 IP
      ],

      # - 
      default: [
        "100.64.127.0/24" # 254 IP
      ]
    }
  }
}
```

---

## □□□□

### □□

- □□□□ - UE IP □□□□APN □□□□
- **PCO** □□ - □ IP □□□□□□□□ DNS□P-CSCF□MTU
- □□□□ - □□□□□□□□PDN □□□□□ IP □□
- **PFCP** □□ - □□ PFCP □□ UE □□□□ UPF

### □□□□

- **S5/S8** □□ - □□ GTP-C □□ IP □□
- **Diameter Gx** □□ - IP □□□□□□□□

### □□

- □□□□ - IP □□□□□□□□□□□□
- □□ **CDR** □□ - CDR □□ UE IP □□□□□□□□

---

## □□□□□□

# OmniPGW 部署

## OmniPGW - 部署 (PGW-C)

OmniPGW 部署

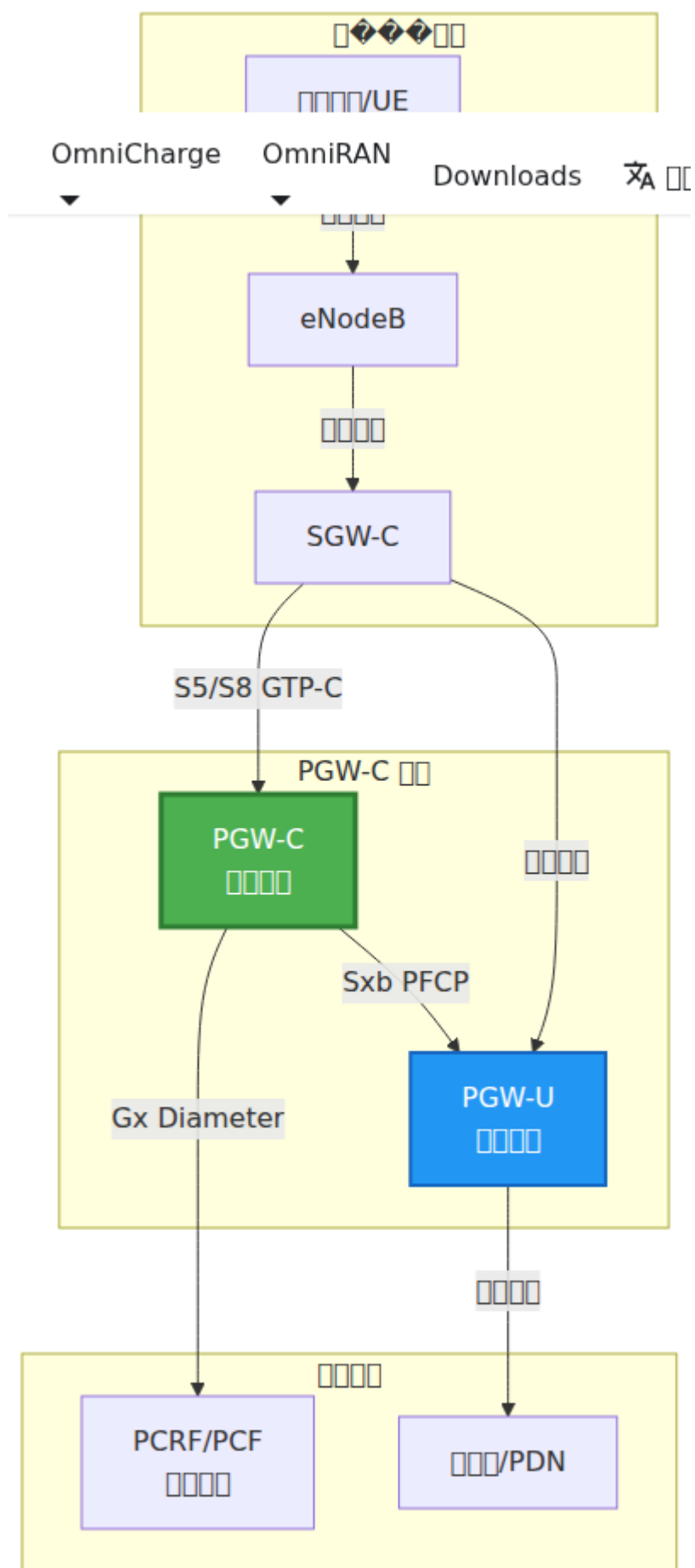
### 部署

1. 部署
2. 部署
3. 部署
4. 部署
5. 部署
6. 部署
7. Web UI - 部署
8. 部署
9. 部署
10. 部署
11. 部署
12. 部署

### 部署

OmniPGW 部署 (PGW-C) 部署 3GPP LTE 部署 (EPC) 部署  
OmniPGW 部署

- 部署 - 部署 (UE) 部署
- IP 部署 - 部署 IP 部署
- 部署 - 部署 PCRF 部署
- 部署 - 部署 PGW-U (部署) 部署



## PGW-C

- **SGW-C** S5/S8 (GTP-C)
  - **UE IP**
  - **Gx (Diameter) PCRF**
  - **Sxb (PFCP) PGW-U**
  - **QoS QoS**
  -
-



-  -  - 
-  -  PDN  GenServer
-  -  (IPTEIDSEID )
- **PFCP**  -  PGW-U  PFCP 

---

PGW-C  3GPP 

## S5/S8 (GTP-C v2)

 SGW-C  PGW-C 

  UDP  GTP-C  2



-  
-  
-  
-  

  S5/S8 

## Sxb (PFCP)

 PGW-C  PGW-U 

  UDP  PFCP (            



-  
-  

- 3GPP TS 23.060/36.423
- 3GPP TS 23.060/36.423
- 3GPP TS 23.060/36.423

3GPP TS 23.060 PFCP/Sxb 3GPP

## Gx (Diameter)

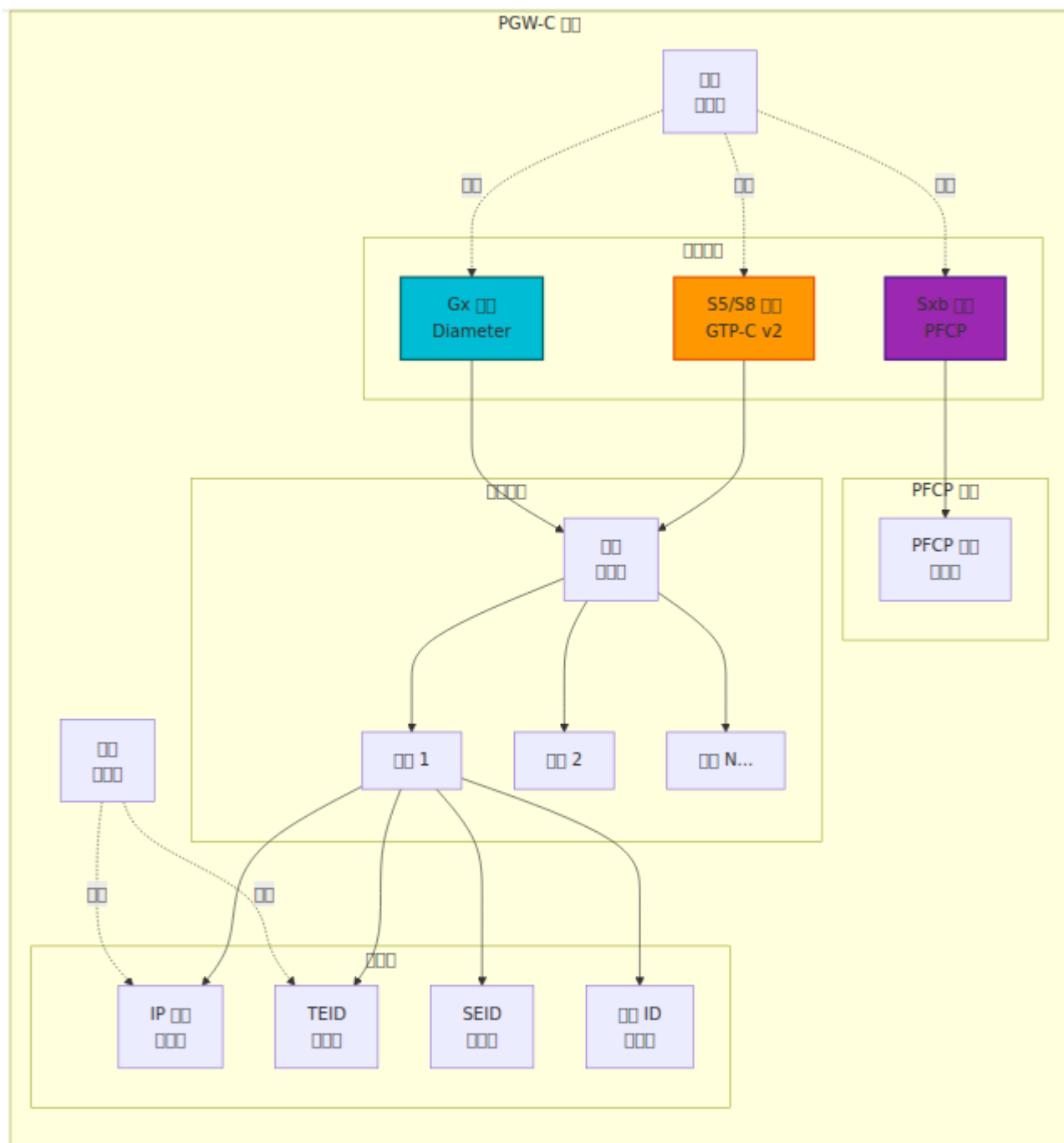
3GPP TS 23.060 (PCRF) 3GPP

3GPP Diameter (IETF RFC 6733)

3GPP

- 3GPP TS 23.060 (CCR-I/CCA-I)
- 3GPP TS 23.060/3GPP (CCR-T/CCA-T)

3GPP TS 23.060 Diameter Gx 3GPP



□□□□

## PDN □□

PDN (□□□□□□□□) □□□□ UE □□□□□ (□□□□□) □□□□□□□□□□□□□□

- **UE IP** □□ - □□□□□□□□□□

- **APN** (Access Point Name) - 訪問点名
- **QoS** (Quality of Service) - サービス品質
- **ID** (Identifier) - 識別子
- **TEID** (Tunnel Endpoint Identifier) - S5/S8 チューネルエンドポイント識別子
- **SEID** (Service Endpoint Identifier) - Sxb サービスエンドポイント識別子

## QoS

QoS (Quality of Service) 機能

- **PDN** (PDN Type) - PDN タイプ
- **QoS** (QoS Class Identifier) - サービス品質クラス識別子
- **EBI** (EPS Bearer ID) - EPS ベアラー ID
- **QoS** (QoS Class Identifier) - QCI/ARP (MBR/GBR)

## PFCP

PGW-C と PGW-U の間での通信

- **PDR** (Policy Decision Rule) - ポリシー決定ルール (ルール/アクション)
- **FAR** (Forwarding Action Rule) - 転送アクションルール
- **QER** (QoS Enforcement Rule) - QoS 強制ルール
- **BAR** (Bearer Authorization Rule) - ベアラー承認ルール

QoS 機能 PFCP 機能

## IP

UE IP アドレス

- **APN** (Access Point Name) - APN 識別子
- **IP** (IP Address) - IP アドレス
- **UE** (User Equipment) - ユーザ機器 IP アドレス
- **IP** (IP Address) - IP アドレス

QoS 機能 UE IP アドレス

---

## 環境

### 前提条件

- Elixir ~1.16
- Erlang/OTP 26+
- SGW-C/PGW-U / PCRF 対応
- LTE EPC 対応

## OmniPGW

1. `config/runtime.exs` を確認

2. 依存関係のインストール

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

3. アプリケーションの実行

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

### 確認

#### ログの確認

```
[info] OmniPGW...
[info] 127.0.0.42:42069 接続
[info] 127.0.0.10 から S5/S8 接続
[info] 127.0.0.20 から Sxb 接続
[info] Gx 接続
[info] PCRF 接続
[info] OmniPGW 起動
```

メトリクスを確認: `http://127.0.0.42:42069/metrics` (ブラウザ)

---



| Label           | Component                        | Feature                          |
|-----------------|----------------------------------|----------------------------------|
| <b>metrics</b>  | Prometheus <input type="text"/>  | <input type="text"/>             |
| <b>diameter</b> | Gx <input type="text"/> PCRF     | Diameter Gx <input type="text"/> |
| <b>s5s8</b>     | GTP-C <input type="text"/> SGW-C | S5/S8 <input type="text"/>       |
| <b>sxb</b>      | PFCP <input type="text"/> PGW-U  | PFCP <input type="text"/>        |
| <b>ue</b>       | UE IP <input type="text"/>       | IP <input type="text"/>          |
| <b>pco</b>      | <input type="text"/>             | PCO <input type="text"/>         |
| <b>CDR</b>      | <input type="text"/>             | CDR <input type="text"/>         |

# Web UI -

OmniPGW  **Web UI**

# Web UI

http://<omnipgw-ip>:<web-port>/

Table with 4 columns: Component, URL, Description, and Count.

| Component | URL              | Description          | Count |
|-----------|------------------|----------------------|-------|
| UE        | /ue_search       | UE Search            | 1     |
| PGW       | /pgw_sessions    | PGW Sessions         | 2     |
| Session   | /session_history | Session History      | 5     |
| Topology  | /topology        | Network Topology     | 5     |
| IP        | /ip_pools        | UE IP Pools          | 2     |
| PFCP      | /pfcpsessions    | PGW-U PFCP Sessions  | 2     |
| UPF       | /upf_status      | UPF PFCP Sessions    | 2     |
| UPF       | /upf_selection   | UPF Selection P-CSCF | 1     |
| Diameter  | /diameter        | PCRF Diameter        | 1     |
| P-CSCF    | /pcscf_monitor   | P-CSCF DNS Monitor   | 5     |
| Gy        | /gy_simulator    | Gy/Ro Simulator      | 1     |
| Cell      | /cell_towers     | OpenCellID Towers    | 1     |
| Logs      | /logs            | System Logs          | 1     |

## □□□□

### □□□□

- □□□□□□□□ (□□□□□□□□)
- □□ OmniPGW □□□□□□□□
- □□□□□□□□□□ (□□/□□)

### □□□□□□

- □ IMSI□IP□MSISDN □ APN □□□□
- □□□□□□□□□□□□□□

### □□□□□□□□

- □□□□□□□□□□□□□□
- □□□□□□□□
- □□□□□□□□□□

### □□□□□□ (□□□□)□

- □□□□□□□□□□
- □□□□ NOC/□□□□□□□□
- □□□□□□ IP □□□□□□

## □□□□□□□□

### □□□□□□ (□□□□)□

1. 검색
2. UE 검색 (/ue\_search)
3. IMSI/MSISDN IP 검색
4. 검색결과
  - a) 검색 - 검색결과
  - b) 검색 - TAC ID
  - c) 검색 -
    - QCI/MBR/GBR
    - APN-AMBR
  - d) 검색 - Gy ID
  - e) 검색 - Gx PCC
  - f) 검색 -
5. 검색 → Diameter PCRF
6. 검색 →

검색

1. 검색
2. PGW (/pgw\_sessions)
3. IMSI
4.
  - UE IP
  - QoS
  -
5. 검색 → UE

검색

1. UPF → PGW-U "
2. Diameter → PCRF "
3. PGW →

검색

- PGW
- /
- 
- APN

# Web UI

## Web UI

- (UE )
- 
- (PFCPDiameter)
- 
- IMSI/MSISDN/IP
- ( )
- QoS (MBRGBRQCI)
- 
- 
- IP
- 

## Prometheus

- 
- 
- 
- 
- 

- Web UI Prometheus

## 

Web UIOmniPGW Prometheus

## 

-

- `teid_registry_count` - S5/S8 消息
- `seid_registry_count` - PFCP 消息
- `session_id_registry_count` - Gx 消息
- `address_registry_count` - UE IP 地址
- `charging_id_registry_count` - 计费 ID

- 入站消息

- `s5s8_inbound_messages_total` - S5/S8 GTP-C 消息
- `sxb_inbound_messages_total` - PFCP 消息
- `gx_inbound_messages_total` - Diameter 消息
- 其他消息

- 入站错误

- `s5s8_inbound_errors_total` - S5/S8 错误
- `sxb_inbound_errors_total` - PFCP 错误
- `gx_inbound_errors_total` - Diameter 错误

## 配置

通过 HTTP 接口配置

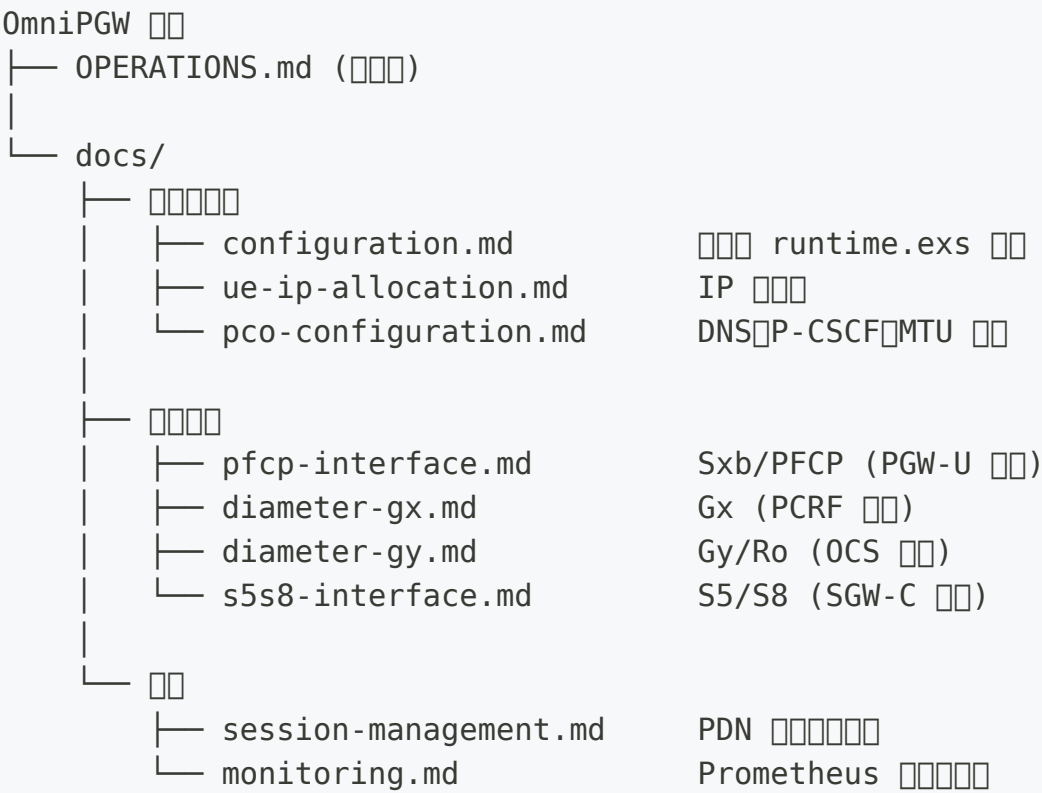
```
curl http://127.0.0.42:42069/metrics
```

通过 `omni` 配置

## 部署

部署 OmniPGW 需要以下步骤

0000



0000000000

0 00

| 00            | 00           | 00       |
|---------------|--------------|----------|
| OPERATIONS.md | 000000 (000) | 00000000 |

⚙ 00

| 파일명                                  | 설명                        | 라인 수   |
|--------------------------------------|---------------------------|--------|
| <a href="#">configuration.md</a>     | runtime.exs 런타임 설정        | 1,600+ |
| <a href="#">ue-ip-allocation.md</a>  | UE IP 할당                  | 943    |
| <a href="#">pco-configuration.md</a> | PCO 설정 (DNS, P-CSCF, MTU) | 344    |

### 네트워크

| 파일명                               | 설명                             | 라인 수   |
|-----------------------------------|--------------------------------|--------|
| <a href="#">pfcp-interface.md</a> | PFCP/Sxb 인터페이스 PGW-U           | 1,355  |
| <a href="#">diameter-gx.md</a>    | Diameter Gx 인터페이스 PCRF (HSS)   | 941    |
| <a href="#">diameter-gy.md</a>    | Diameter Gy/Ro 인터페이스 OCS (HSS) | 1,100+ |
| <a href="#">s5s8-interface.md</a> | GTP-C S5/S8 인터페이스 SGW-C        | 456    |

### 데이터

| 파일명                                   | 설명                       | 라인 수 |
|---------------------------------------|--------------------------|------|
| <a href="#">session-management.md</a> | PDN 세션 관리                | 435  |
| <a href="#">monitoring.md</a>         | Prometheus, Grafana 모니터링 | 807  |
| <a href="#">data-cdr-format.md</a>    | CDR 데이터 URR 형식           | 847  |
| <a href="#">qos-bearers.md</a>        | QoS 베어러                  | 448  |
| <a href="#">troubleshooting.md</a>    | 문제 해결                    | 687  |

### 기타

| 項目                  | 項目          | 項目  |
|---------------------|-------------|-----|
| pcscf-monitoring.md | P-CSCF 監視機能 | 894 |

目次

■ Mermaid 図

監視機能 Mermaid 図 監視機能

- 監視
- 監視 (監視)
- 監視
- 監視

図 監視

監視機能

- 監視機能
- 監視機能
- 監視

図 監視

監視機能

- 監視機能
- 監視
- 監視

図 監視

監視機能

## □□□□

### □□□□□□□

1. [OPERATIONS.md](#) - □□ (□□□)
2. [configuration.md](#) - □□
3. [monitoring.md](#) - □□
4. [session-management.md](#) - □□□□

### □□□□□□□

1. [OPERATIONS.md](#) - □□□□ (□□□)
2. [pfcg-interface.md](#) - □□□□□□
3. [diameter-gx.md](#) - □□□□
4. [diameter-gy.md](#) - □□□□
5. [s5s8-interface.md](#) - □□□□
6. [ue-ip-allocation.md](#) - IP □□

### □□□□□□□

1. [configuration.md](#) - □□□□
2. [ue-ip-allocation.md](#) - IP □
3. [pco-configuration.md](#) - □□□□
4. [monitoring.md](#) - □□□□

## □□□□

- □□□□□ 14
- □□□□ ~10,900+
- □□□□ ~265 KB
- **Mermaid** □□□ 75+
- □□□□□ 150+

## □□□□□□□□

□□

- 0 00/0000000
- 0 OTP/Elixir 00
- 0 0000
- 0 00 GenServer 000

00

- 0 PFCP (0000000000)
- 0 GTP-C v2 (GPRS 0000)
- 0 Diameter (RFC 6733)

### 3GPP 00

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

00

- 0 0000
  - 0 IP 0000
  - 0 QoS 00
  - 0 0000
  - 0 000000
-

□□□□

3GPP □□

| □□        | □□                    |
|-----------|-----------------------|
| TS 29.274 | GTP-C v2 (S5/S8 □□)   |
| TS 29.244 | PFCP (Sxb □□)         |
| TS 29.212 | Diameter Gx □□ (□□□□) |
| TS 32.299 | Diameter □□□□ (Gy/Ro) |
| TS 32.251 | □□□□□□□□              |
| TS 23.401 | EPC □□                |

□□□□

- □□□□ [config/runtime.exs](#)
-