



000000 DRA 0000 Diameter 000000000000000000 000000 000000000000000000
000 ID000000 AVP 00000000000000000000000000000000

priority weight
 Gx (PCRF) Gy (OCS)

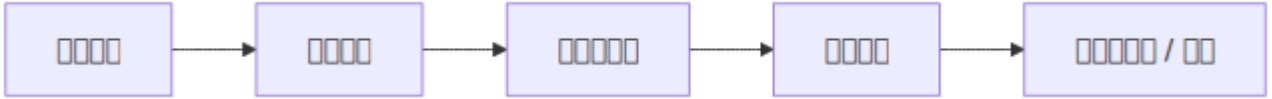
Diameter _____ DRA _____
_____ / _____



DRA 00000000 0000 0000000000000000000000000000DRA 000000000000000000
0000000000 0000 / 000 000

--

DRA



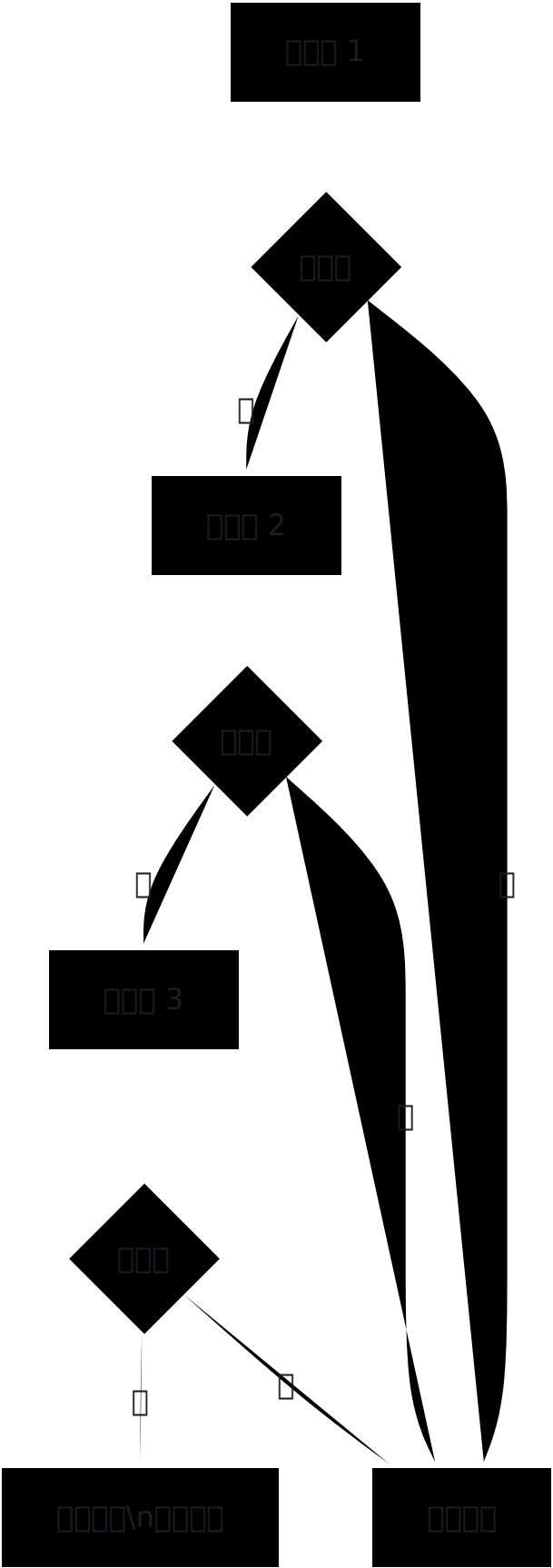
1. DRA 00000000 0000 00000000
2. DRA 00000 match 000 :all, :any, 0 :none0000000000
3. 000000000000DRA 00000000000
4. 0000000000DRA 00000000

□ □ □ □

match □□□□ DRA □□□□□□□□□□

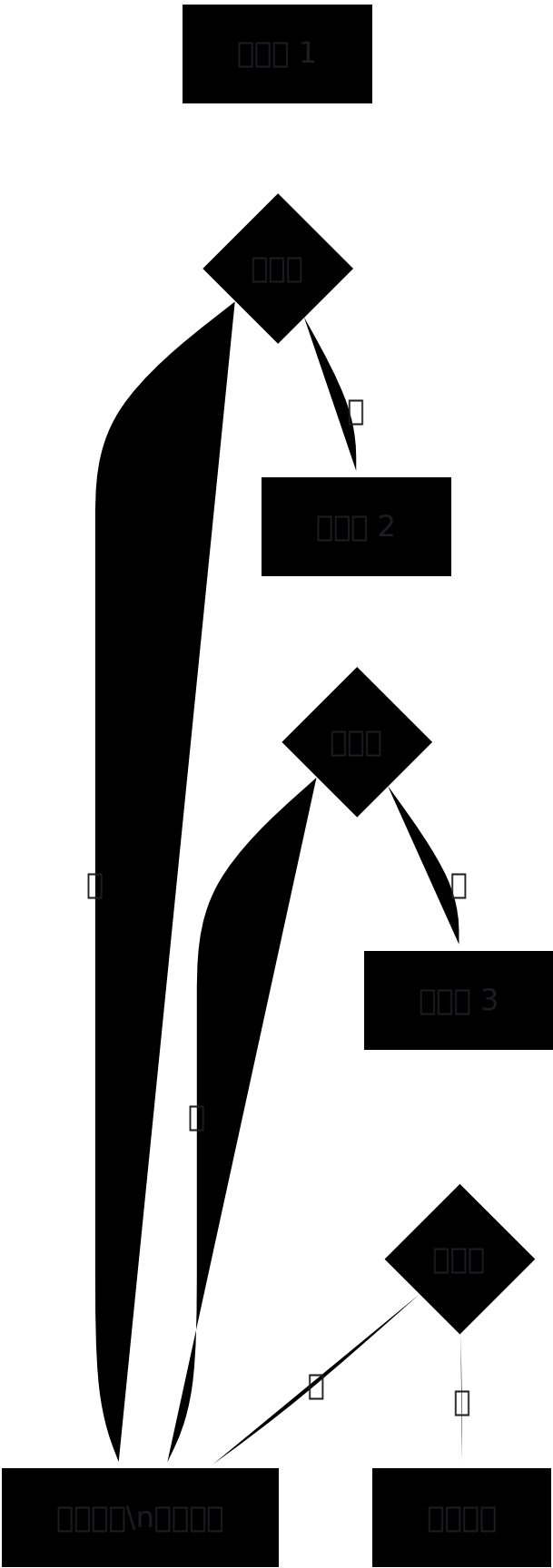
```
:all ( )
```

filter1 AND filter2 AND filter3



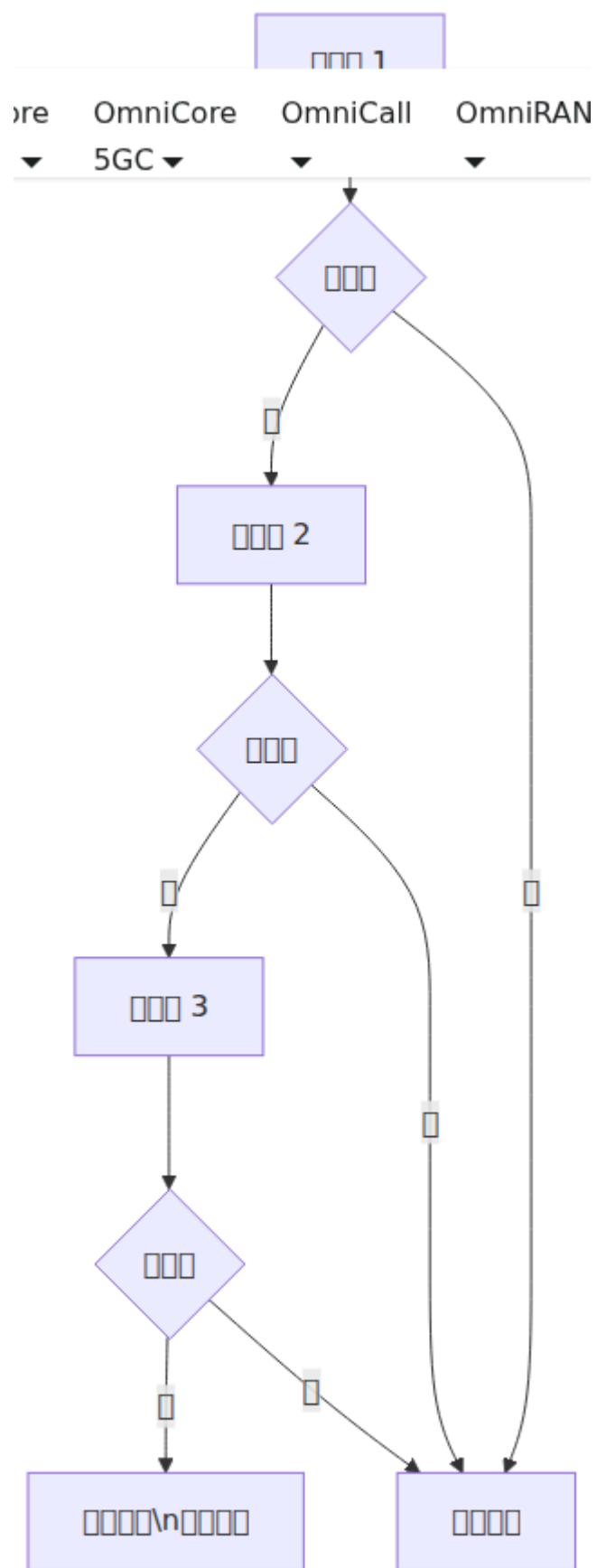
:any (节点)

filter1 OR filter2 OR filter3



:none ()

NOT filter1 AND NOT filter2 AND NOT filter3



概要

このドキュメントは、DRA の `peer_selection_algorithm` の設定方法と、DRA の動作について説明します。

設定方法

`diameter` の設定で `peer_selection_algorithm` を `:failover` または `:random` に設定することで、DRA の Diameter の動作を変更できます。DRA の動作は、DRA の設定で指定された動作に従って実行されます。

```
route: %{peers: ["hss01.example.com", "hss02.example.com"]}
```

- `:failover` - 優先度の高いピアから接続を試みます。失敗した場合は、優先度の低いピアに接続を試みます。
- `:random` - ランダムにピアを選択して接続を試みます。

動作

DRA は、`priority` と `weight` を使ってピアの優先度を決定します。

```
route: %{
  peers: [
    %{host: "pcrf01.example.com", priority: 10, weight: 50},
    %{host: "pcrf02.example.com", priority: 10, weight: 50},
    %{host: "pcrf-dr.example.com", priority: 20, weight: 100}
  ]
}
```

この “動作” は、DRA の動作を説明します。

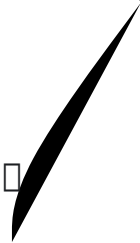
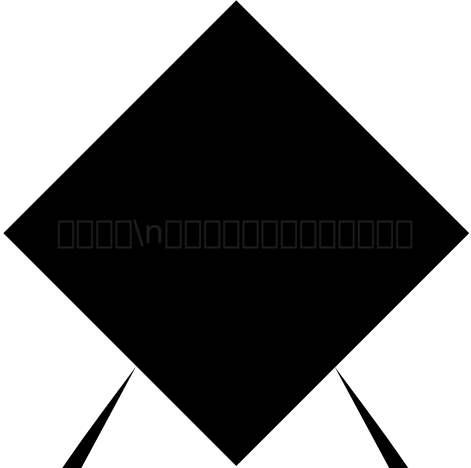
1. DRA は、ピアの優先度を決定します。
2. DRA は、優先度の高いピアから接続を試みます。DNS SRV の結果が、DRA の動作に影響を与えます。
3. DRA は、ピアの `weight` を使ってピアの優先度を決定します。この `weight` は、DRA の動作に影響を与えます。

DRA の `priority` と `weight` の設定は、DRA の動作に影響を与えます。

□□□□□□□□

□□□□□□□□

□□□□□□□□□□



□□□□□□□□\n□□□□□□

□□□□□□□□□□

□□□□□□□□□□□□□□

項目	説明
間隔	10, 20, 30秒単位
単位	秒
範囲	70 / 30
初期値	10 秒 20 秒

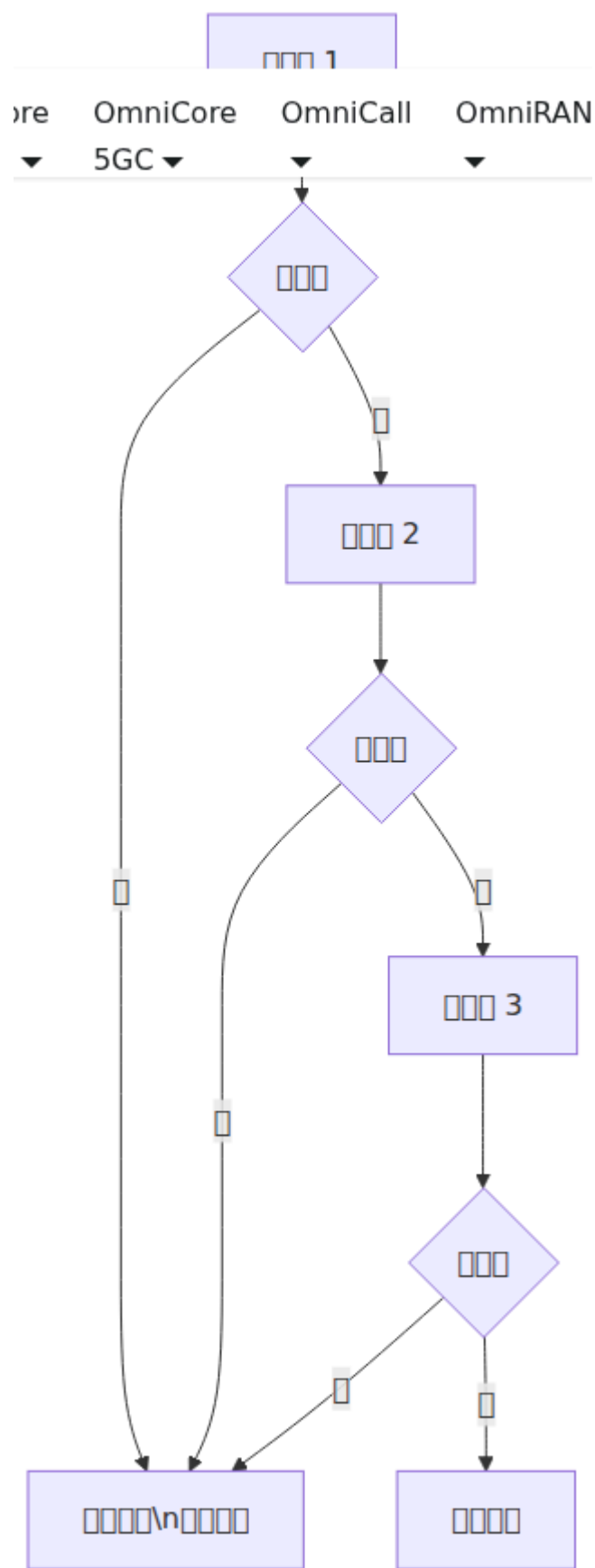
設定 (Gx / Gy)

設定 ID 設定 DRA 設定
DIAMETER_UNKNOWN_SESSION_ID (5002)

Diameter 設定 **AVP (293)**

- DRA 設定
- DRA 設定

Gx CCR-Initial 設定 DRA 設定 Origin-Host
CCR-Update, CCR-Terminate 設定
Gx/Gy **CC-Request-Type AVP (416)**



配置

在 `config/runtime.exs` 中配置 `module_advanced_routing` 模块

```
module_advanced_routing: %{
  # 是否启用
  enabled: false,

  # 配置规则
  rules: [
    %{
      rule_name: "route_s6a_air_to_hss",
      match: :all,
      filters: [
        {:application_id, 16_777_251},
        {:command_code, 318}
      ],
      route: %{
        peers: ["hss01.example.com"]
      }
    }
  ]
}
```

配置项

配置项	数据类型	默认值	说明
<code>enabled</code>	布尔值	<code>false</code>	是否启用 DRA 模块
<code>rules</code>	列表	<code>[]</code>	DRA 配置规则列表

필드

필드	타입	필드 명	필드 값	설명
rule_name	문자열	필수	-	필드명 DRA 필드명 UI 필드명
match	문자열	필수	-	필드명 :all 필드명 :any 필드명 :none 필드명
filters	문자열	필수	-	필드명
route	문자열 문자열	필수	-	DRA 필드명 필드명 필드명 필드명

필드

필드 {filter_type, value} 필드명

필드	필드	필드
:via_peer	{:via_peer, "mme01.example.com"}	필드명 Origin- Host 필드명
:application_id	{:application_id, 16_777_251}	Diameter 필드명 ID 필드명 필드명 ID
:command_code	{:command_code, 318}	Diameter 필드명
:avp	{:avp, {1, "9999990000000001"}}	AVP 필드명 {avp_code, value}

필드

- 필드명 필드명 DRA 필드명 {:command_code, [317,
318]} 필드명 317 필드명 318 {:via_peer, ["dra01.example.com",

```
"dra02.example.com"]}
```

- `AVP { :avp, {1, ~r"999001.*"} }` `{ :avp, {1, ["505571234567", ~r"999001.*", ~r"505057.*"]} }`
- `:any` `:any` `{ :avp, {264, :any} }` `Origin-Host AVP { :via_peer, :any }`

AVP

```

00 00 AVP 00 AVP 0000000000Origin-Host 0000000000 AVP0000000000000000
0000000000 {:avp, {code, value}} 000

```

□ □ □ □ □

route

項目	項目名	説明
接続先 IP	<code>%{peers: [...]}</code>	接続先 IP アドレス / ネットワーク
接続先 ホスト名	<code>:destination_host</code>	接続先ホスト名 AVP (293) によって指定された AVP によって 指定された DRA によって指定された
接続先 ポート	<code>:drop</code>	接続先ポート番号 接続先ポート番号
接続先 エラーコード	<code>{:error, 3004}</code>	接続先エラーコード Diameter によって指定された DRA によって Origin-Host Origin-Realm Session-Id DRA によって指定された 接続先

00 00 00000000 3xxx000000000000000000000000 3002 3003 3004 3007 0000000000
5012 DIAMETER UNABLE TO COMPLY000000000000000000000000

配置示例/配置

名称	类型	单位	默认值	描述
host	字符串	-	-	Origin-Host 和 Diameter 主机名
priority	整数	100	100	DRA 的优先级
weight	整数	100	100	DRA 的权重 0

配置

配置示例

S6a 配置示例 HSS HSS 配置示例 HSS

```
{
  rule_name: "s6a_air_failover",
  match: :all,
  filters: [
    {application_id, 16_777_251},
    {command_code, 318}
  ],
  route: {
    peers: [
      {host: "hss01.example.com", priority: 10},
      {host: "hss-dr.example.com", priority: 20}
    ]
  }
}
```

hss01 hss01 hss-dr
DRA 3002

Gx PCRF

```
%{
  rule_name: "gx_load_balance",
  match: :all,
  filters: [
    {:application_id, 16_777_238}
  ],
  route: %{
    peers: [
      %{host: "pcrf01.example.com", priority: 10, weight: 50},
      %{host: "pcrf02.example.com", priority: 10, weight: 50}
    ]
  }
}
```

DRA CCR-Initial 50/50 PCRF DRA
PCRF

PCRF PCRF

```
%{
  rule_name: "gx_weighted_with_backup",
  match: :all,
  filters: [
    {:application_id, 16_777_238}
  ],
  route: %{
    peers: [
      %{host: "pcrf01.example.com", priority: 10, weight: 70},
      %{host: "pcrf02.example.com", priority: 10, weight: 30},
      %{host: "pcrf-dr.example.com", priority: 20, weight: 100}
    ]
  }
}
```

PCRF 10 PCRf DRA 70/30 10 PCRf pcrf-dr

DRA S6a HSS

```
%{
  rule_name: "s6a_via_peer_to_local_hss",
  match: :all,
  filters: [
    {:application_id, 16_777_251},
    {:via_peer, ["dra01.example.com", "dra02.example.com"]},
    {:avp, {296, "epc.mnc001.mcc001.3gppnetwork.org"}}
  ],
  route: %{
    peers: ["hss01.example.com", "hss02.example.com"]
  }
}
```

DRA dra01 dra02 Origin-Realm AVP 296 S6a HSS

IMSI

IMSI AVP 1

```
%{
  rule_name: "inbound_s6a_roaming",
  match: :all,
  filters: [
    {application_id, 16_777_251},
    {avp, {296, "epc.mnc001.mcc001.3gppnetwork.org"}},
    {avp, {1, ["505571234567", ~r"999001.*"]}}
  ],
  route: %{
    peers: ["dra01.example.com", "dra02.example.com"]
  }
}
```

Origin-Realm S6a 505571234567 999001
DRA

AVP

```
%{
  rule_name: "honour_destination_host",
  match: :all,
  filters: [
    {avp, {1, [~r"90199.*"]}}
  ],
  route: :destination_host
}
```

IMSI DRA AVP (293) AVP DRA

```
%{
  rule_name: "drop_test_subscribers",
  match: :all,
  filters: [
    {:application_id, 16_777_251},
    {:avp, {1, [~r"999999.*"]}}
  ],
  route: :drop
}
```

DRA IMSI 999999 S6a

□ □ □ □ □ □ □ □ □ □ □

DIAMETER_TOO_BUSY

```
%{
  rule_name: "rate_limit_high_volume_peer",
  match: :all,
  filters: [
    {:via_peer, "mme-overloaded-01.example.com"},
    {:application_id, 16_777_251}
  ],
  route: {:error, 3004}
}
```

0000 DRA 0 **DIAMETER_TOO_BUSY (3004)** 0000000000 S6a 00000000000000
00DRA 000000000000

□ □ □ □ □ □

□ □

```
%{
  rule_name: "block_purge_requests",
  match: :all,
  filters: [
    {:application_id, 16_777_251},
    {:command_code, 321}
  ],
  route: {:error, 5012}
}
```

DRACORE **DIAMETER_UNABLE_TO_COMPLY (5012)** S6a UE 321

DRACORE

API GET /api/rules routing_algorithm

API

- route.selection / "priority_weight" "failover" "random"
- priority_weight API priority weight

```
{
  "status": "success",
  "response": {
    "routing_algorithm": "random",
    "routing_rules": [
      {
        "rule_name": "gx_weighted_with_backup",
        "match": "all",
        "route": {
          "selection": "priority_weight",
          "peers": [
            { "host": "pcrf01.example.com", "priority": 10,
"weight": 70 },
            { "host": "pcrf02.example.com", "priority": 10,
"weight": 30 },
            { "host": "pcrf-dr.example.com", "priority": 20,
"weight": 100 }
          ]
        }
      }
    ],
    "transform_rules": []
  }
}
```

1. 00000000 000000000000000000000000000000000000
2. 00000000 000000000000000000000000
3. 000000000 00000000000000000000000000
4. 000000 0000000 rule_name 00000000 UI 00000000
5. 00000000000000000000/0000 00 Gx/Gy 0000000/0000000000DRA 00000000000000

--	--	--	--

UNKNOWN_SESSION_ID

Gx Gy 5002

□□□□□

- PCEF/OCF $\rightarrow$ DRA $\rightarrow$ $\rightarrow$
- : random $\rightarrow$ $\rightarrow$

□ □ □ □ □

1.
2. / DRA

□□□□□□ **3002**

DRG DIAMETER_UNABLE_TO_DELIVER

□□□□□

- `XXXXXXXXXXXXXXXXXXXX`
- `XXXXXXXXXXXXXXXXXXXX` Origin-Host `XXXXXXXXXX`

□□□□□

1. `UI` `Origin-Host`
2. `host` `Origin-Host`

□ □ □ □ □

□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

□ □ ◆ ◆ □ □

- enabled false
-

- `enable_avp_dra` AVP `enable_avp_dra` AVP

1. `enable_avp_dra`

1. `enable_avp_dra` `true` DRA
2. `enable_avp_dra`
3. `enable_avp_dra` DRA `enable_avp_dra` AVP

2. `enable_avp_dra`

Diameter

Code	Message	Description	Reference
3002	DIAMETER_UNABLE_TO_DELIVER	Unable to deliver message	RFC 6733 §7.1.3
3003	DIAMETER_REALM_NOT_SERVED	Realm not served	RFC 6733 §7.1.3
3004	DIAMETER_TOO_BUSY	Too busy	RFC 6733 §7.1.3
5002	DIAMETER_UNKNOWN_SESSION_ID	Unknown session ID	RFC 6733 §7.1.5
5012	DIAMETER_UNABLE_TO_COMPLY	DRA unable to comply	RFC 6733 §7.1.5

AVP

AVP	Code	Meaning	Reference
Session ID	293	Session ID	RFC 6733 §6.5
Peer ID	263	Peer ID	RFC 6733 §8.8
CC-Request Type	416	CC Gx/Gy Peer INITIAL/UPDATE / TERMINATION Peer	RFC 4006 §8.3

Peer ID

ID	Code	Meaning	Reference
16777238	Gx	PCEF / PCRF Peer	3GPP TS 29.212
16777251	S6a/S6d	MME/SGSN / HSS Peer	3GPP TS 29.272
4	Gy	OCF / OCS Peer	RFC 4006

Notes

- Peer - Peer / Peer
- Peer - Peer
- Peer - Peer `config/runtime.exs` Peer `peer_selection_algorithm`

AAA

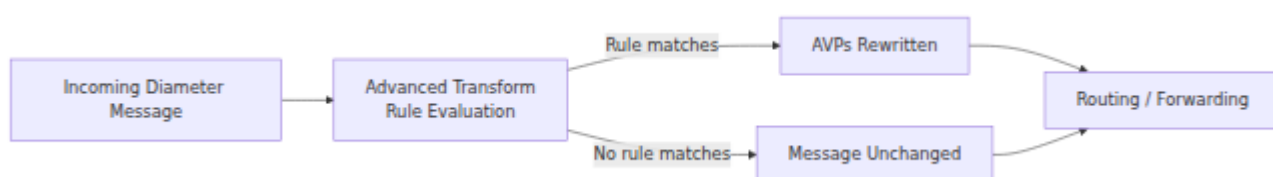
AAA is a Diameter-based AAA solution that integrates with OmniDRA to provide a unified AAA architecture.

Overview

AAA is a Diameter-based AAA solution that integrates with OmniDRA to provide a unified AAA architecture.

AAA supports the following operations: **:edit** (edit AVP), **:remove** (remove AVP), and **:overwrite** (overwrite AVP).

AAA is a Diameter-based AAA solution that integrates with OmniDRA to provide a unified AAA architecture.



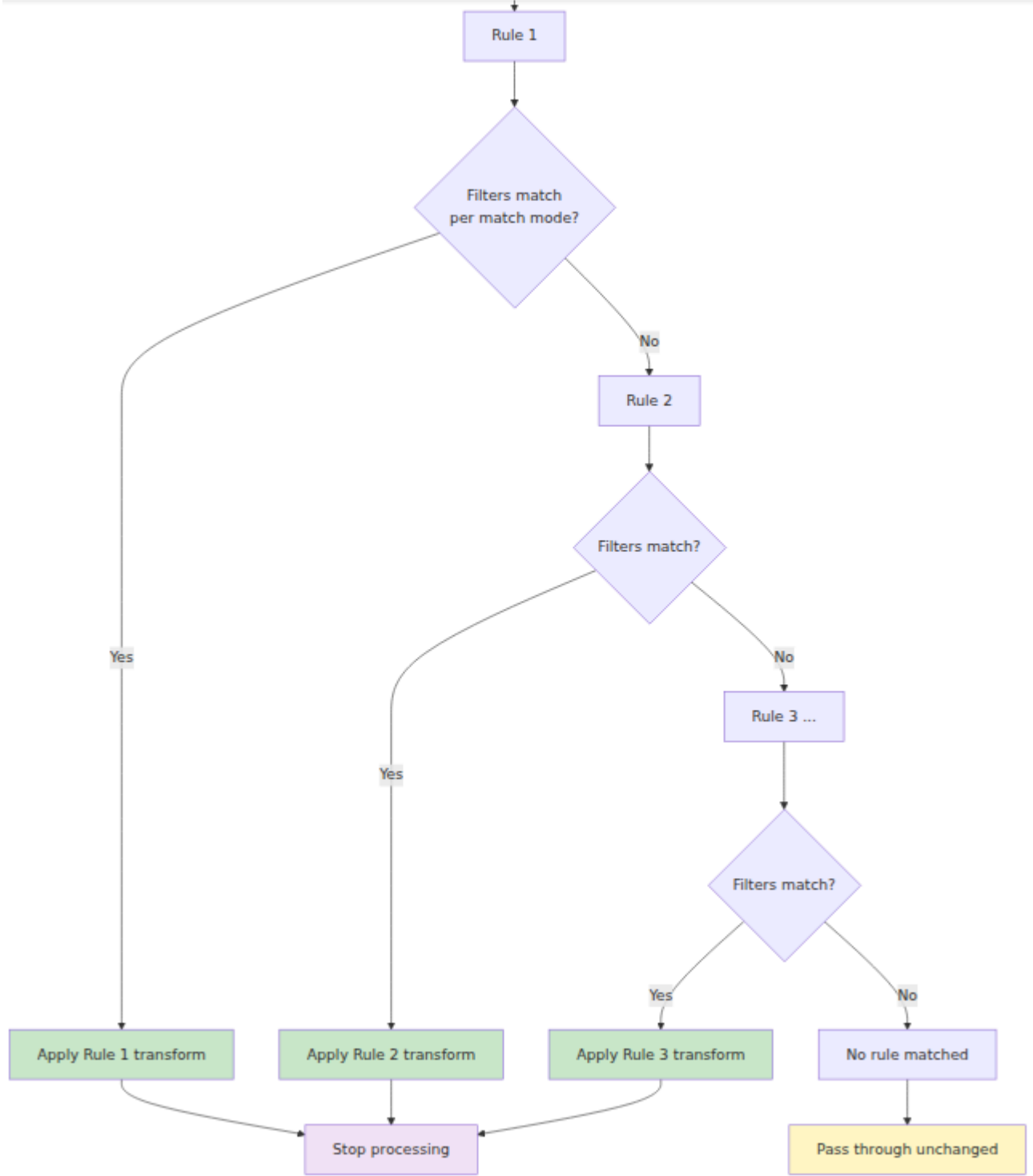
Configuration

AAA is a Diameter-based AAA solution that integrates with OmniDRA to provide a unified AAA architecture.

AAA supports the following match conditions:

- :all** - AND all conditions
- :any** - OR any conditions
- :none** - NOR none conditions

□ □



AAA

AA

項目	形式	説明
:packet_type	{:packet_type, :request}	パケットの種類: request :answerはパケットの種類 番号
:application_id	{:application_id, <id>}	Diameter ID S6a/S6d 16777251 ID
:command_code	{:command_code, <code>}	Diameter AIR 318
:avp	{:avp, {:code, <value>}}	AVP AVP
:to_peer	{:to_peer, "hss01.example.com"}	宛先 hss01.example.com 番号
:from_peer	{:from_peer, "hss01.example.com"}	送信元 hss01.example.com 番号

パケット

:packet_typeはパケットの種類番号

- :request - Diameter リクエスト
- :answer - Diameter レスポンス

:application_id :command_code :avp :packet_type

AVP

:avp {avp_id, avp_value} avp_id AVP avp_value
番号

AVP User-Name Origin-Host :avp AVP

- { :avp, {1, "9999990000000001"} }
- { :avp, {1, ["50557...", "505057..."]} } AVP OR
- { :avp, {1, [~r"9999999.*"]} }

~r"pattern"

~r"999001.*"	999001 User-Name/IMSI
~r"^310[0-9]{3}.*"	MCC/MNC
~r".*test\$"	test

:to_peer :from_peer

- :to_peer DRA
- :from_peer

OR

項目	値	説明
<code>:to_peer</code>	<code>{:to_peer, "hss01.example.com"}</code>	送信元アドレス <code>hss01.example.com</code> に送信 する
<code>:to_peer</code>	<code>{:to_peer, ["hss01.example.com", "hss02.example.com"]}</code>	送信元アドレスリスト
<code>:from_peer</code>	<code>{:from_peer, "hss01.example.com"}</code>	受信元アドレス <code>hss01.example.com</code> から受信 する

変換

`transform`は、`action`で指定された変換を実行する。

変換	説明	成功	失敗
<code>:edit</code>	指定したAVPを編集する	○	○
<code>:remove</code>	指定したAVPを削除する	○	○
<code>:overwrite</code>	指定したAVPを上書きする	○	○

`:edit`

```
transform: %{\n  action: :edit,\n  avps: [{:avp, {1, "9999990000000002"}}]\n}
```

`:edit`は、

- Diameter AVP
- AVP AVP DRA
- avps { :avp, { <code>, <new_value> } } AVP
- :edit AVP AVP :overwrite

: remove

```
transform: %{
  action: :remove,
  avps: [{:avp, {264, :any}}]
}
```

: remove

- avps AVP
- AVP avps :any
-

: overwrite

```
transform: %{
  action: :overwrite,
  dictionary: :diameter_gen_3gpp_s6a,
  avps: [{:avp, {1, "999999000000002"}}]
}
```

: overwrite

- AVP dictionary
- :overwrite edit / AVP
- dictionary Diameter :diameter_gen_3gpp_s6a
-


```
module_advanced_transform: %{\n  enabled: true,\n  rules: [\n    %{\n      rule_name: "transform_user_name",\n      match: :all,\n      filters: [\n        {application_id, 16_777_251},\n        {command_code, 318},\n        {avp, {1, "9999990000000001"}}\n      ],\n      transform: %{\n        action: :edit,\n        avps: [{avp, {1, "9999990000000002"}}]\n      }\n    }\n  ]\n}
```

Table

Field	Type	Value	Default	Description
enabled	Boolean	false	false	Whether the module is enabled. If false, the module will not be loaded.
rules	Array	-	-	Array of rules to be applied. Each rule is a hash with the following keys: rule_name, match, filters, and transform. The transform key is optional.

Table

rules

match: :all S6a 16777251 318
User-Name AVP 1 999999000000001 :edit User-Name
999999000000002 User-Name AVP DRA

DRA IMSI

2 IMSI

```
module_advanced_transform: %  
  enabled: true,  
  rules: [  
    %  
      rule_name: "rewrite_destination_realm_roaming_partner",  
      match: :all,  
      filters: [  
        { :avp, { 296, "epc.mnc057.mcc505.3gppnetwork.org" } },  
        { :avp, { 1, [~r"50557.*"] } }  
      ],  
      transform: %  
        action: :edit,  
        avps: [{ :avp, { 283, "epc.mnc030.mcc310.3gppnetwork.org" } }]  
      }  
    ]  
  ]  
}
```

Origin-Realm AVP 296 User-Name/IMSI AVP 1
50557 Destination-Realm AVP 283
IMSI

MVNO DRA IMSI

3 AVP

```
module_advanced_transform: %{\n  enabled: true,\n  rules: [\n    %{\n      rule_name: "strip_origin_host_from_partner_answers",\n      match: :all,\n      filters: [\n        { :packet_type, :answer },\n        { :from_peer, "hss01.partner.example.com" },\n        { :command_code, 318 }\n      ],\n      transform: %{\n        action: :remove,\n        avps: [{ :avp, { 264, :any } }]\n      }\n    }\n  ]\n}
```

packet_type from_peer
hss01.partner.example.com command_code AIA 318
:remove Origin-Host AVP 264 AVP

AVP AVP AVP

4S6aAVP

```
module_advanced_transform: %{\n  enabled: true,\n  rules: [\n    %{\n      rule_name: "overwrite_user_name_s6a",\n      match: :all,\n      filters: [\n        { :packet_type, :request },\n        { :application_id, 16_777_251 },\n        { :command_code, 318 }\n      ],\n      transform: %{\n        action: :overwrite,\n        dictionary: :diameter_gen_3gpp_s6a,\n        avps: [{ :avp, { 1, "9999990000000002" } }]\n      }\n    }\n  ]\n}
```

16777251AIR318:overwrite
:diameter_gen_3gpp_s6aAVP:overwrite
edit/AVP
AVPAVP:editAVP

ID

ID			
16777251	S6a/S6d	MME/SGSN ↔ HSS	3GPP TS 29.272

--	--	--	--

項目	名称	規格	規格
318	無線LAN/無線LAN/AIR/AIA	S6a/S6d	3GPP TS 29.272 §7.2.5
316	無線LAN/無線LAN/ULR/ULA	S6a/S6d	3GPP TS 29.272 §7.2.3

AVP

AVP	AVP	AVP	AVP
1	User-Name	3GPP TS 29.272 / RFC 6733 §8.14	
264	Origin-Host	RFC 6733 §6.3	
283	Destination-Realm	RFC 6733 §6.6	
296	Origin-Realm	RFC 6733 §6.4	

--

1. 名前 - 名前空間のパス
2. 名前 - 名前空間のパス
3. 名前 - 名前空間のパス
4. 名前 - 名前空間のパス `rule_name` ???
5. 名前 - 名前空間のパス

□□□□

- □□□□ - □□□□□□□□□□□□□□□□

DRA 簡介

OmniDRA 是一個 **Diameter** 代理 (DRA) 實現。Diameter 是 3GPP 標準的 MME/SGSN 與 HSS/PCRF 及 OCS 之間使用的協議。OmniDRA 基於 Elixir/Erlang 的 OTP `:diameter` 庫實現。Diameter 是一個基於 TCP 的協議。

目錄

1. DRA 簡介
2. 安裝
3. 配置
4. 測試
5. 部署
6. 參考
7. 附錄

DRA 功能

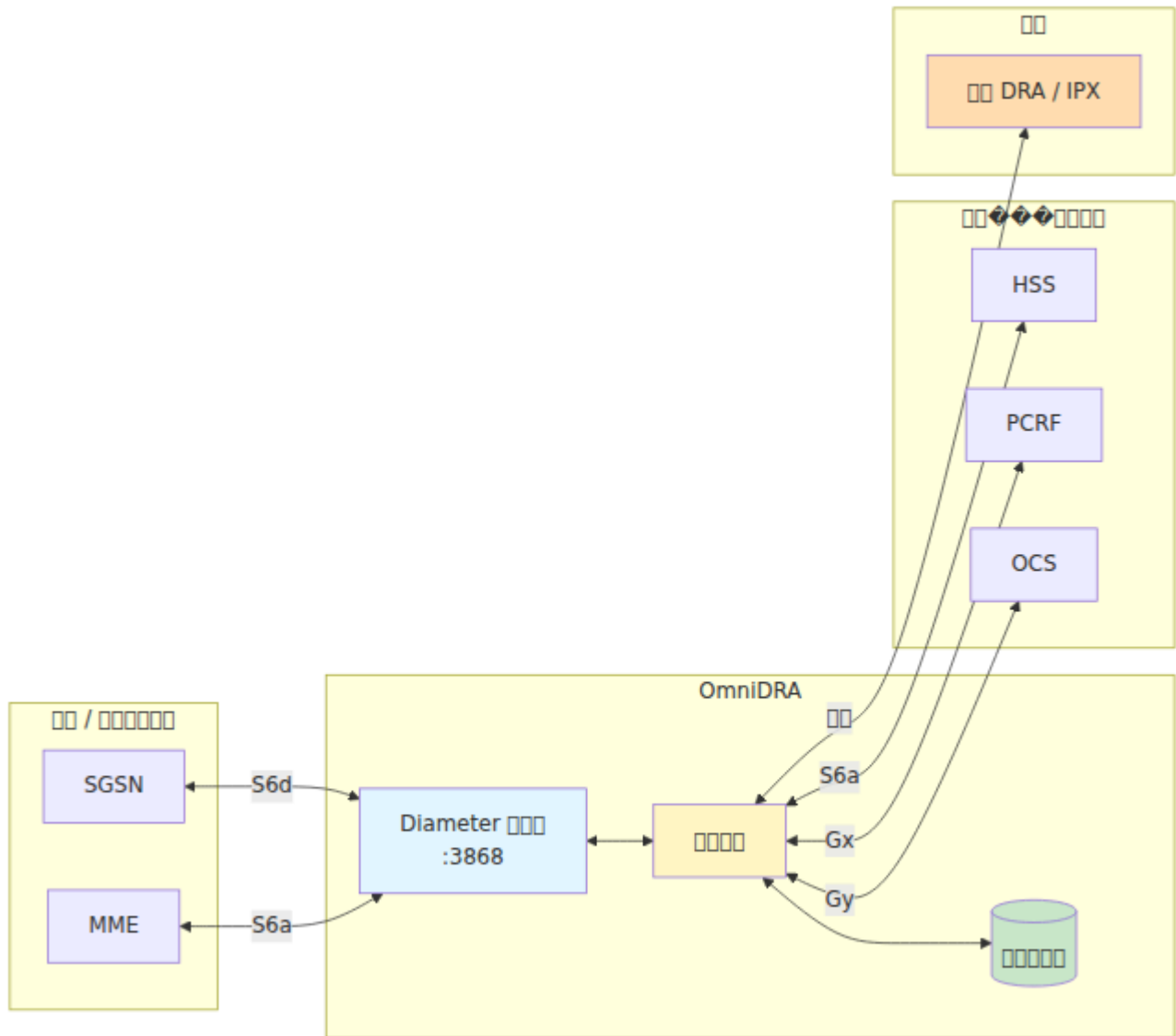
OmniDRA 是一個 Diameter 代理實現，支持 Diameter 協議的 AVP 值。OmniDRA 支持 Diameter 協議的 S6a、Gx 和 Gy 接口。Diameter 協議的 AVP 值在 RFC 6733 中有定義。

功能列表

- 支持 Diameter 協議 (RFC 6733) 的 AVP 值。
Destination-Host、**Destination-Realm** 和 **Application-Id** 等 AVP 值。
- 支持 Diameter 協議的 Hop-by-Hop Identifier 值。
- 支持 Diameter 協議的 AVP 值。
- 支持 Diameter 協議的 AVP 值。

概要

この図は、MME、SGSN、HSS、PCRF、OCS、DRA、DRAのDiameter、DRAのDiameterの概要を示しています。



DRAはTCP/SCTPポート3868でDiameterプロトコル(RFC 6733)をサポートしています。DRAはCER/CEAをサポートし、DRAはDiameterプロトコルをサポートしています。

詳細

この図は、DRAのDiameterプロトコルをサポートする詳細を示しています。

☐☐ Diameter ☐☐

☐☐☐☐☐
☐☐☐☐☐☐☐

Diameter ☐☐☐
GSMA FS.19 ☐☐

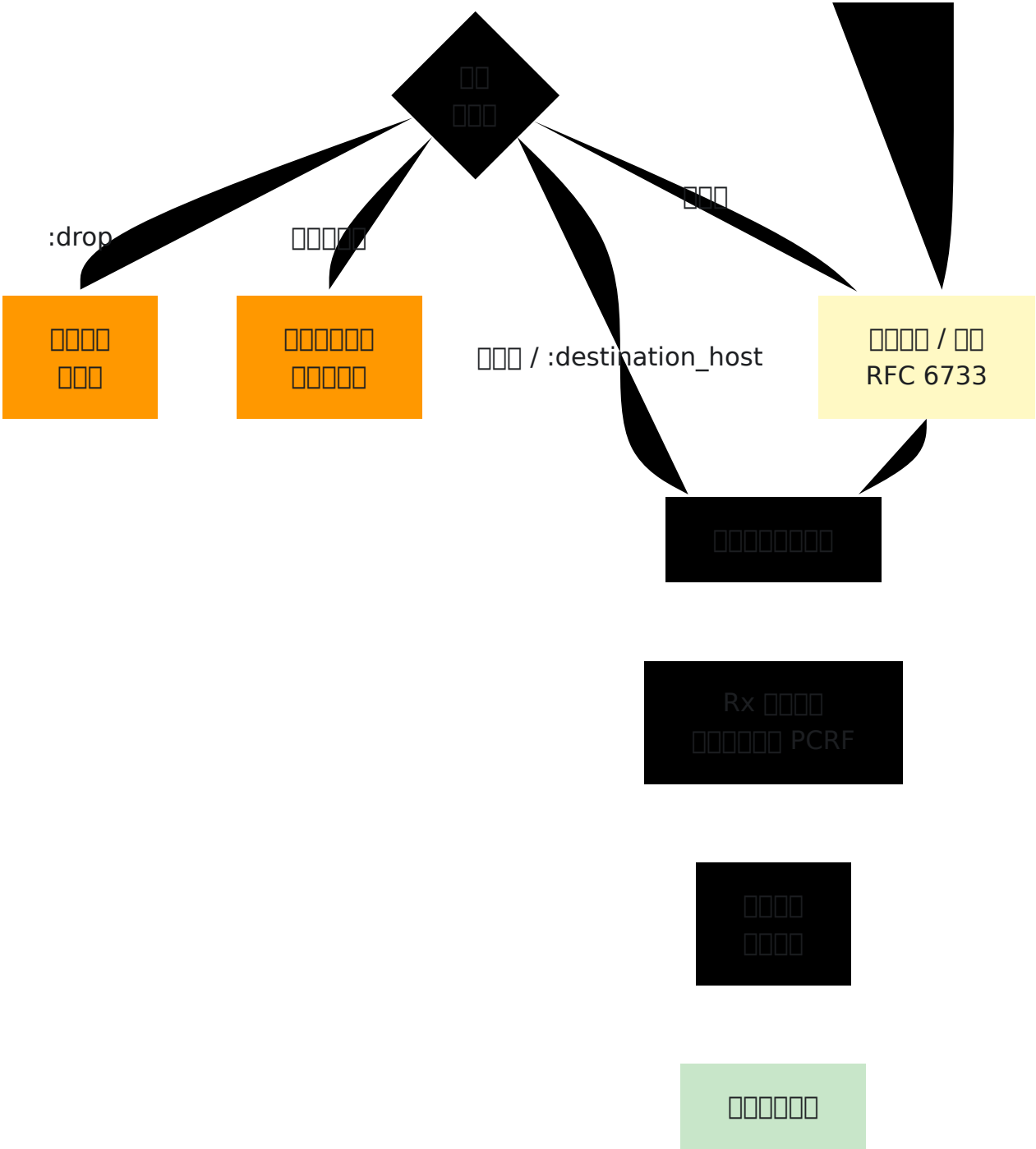
AVP ☐☐☐
☐☐☐☐☐ / ☐☐

☐☐☐☐

☐☐☐☐☐

☐☐☐☐
☐☐☐

☐☐☐☐☐☐☐



IP

IP	IP
IP	IP
Diameter IP	IP GSMA FS.19 IP 1/2/3 IP IP

項目	説明
AVP 項目	AVP 項目 AVP 項目
項目	項目
項目	項目
項目	項目
項目	項目
項目 / 項目	項目 RFC 6733 項目
Rx 項目	Rx 項目 PCRF 項目 Rx Session-Termination-Request 項目 Rx 項目
項目	項目 AVP 項目 Route-Record 項目 Origin-Host/Realm 項目

00000000 :drop 0000DRA 000000000000error 000000000000 **Result-Code** 0
Diameter 000000DRA 000000000000000000 **Destination-Host** 0000000000000000
000

```
Rx [REDACTED] Rx[REDACTED]
:drop [REDACTED]error [REDACTED] Destination-Host[REDACTED]
```

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

□□□□□□□□□□ DRA □□ RFC 6733 §6.1 □□□□□□□□□□□□

1. **Destination-Host AVP (293)** - 32 ビット AVP DRA 宛ホストの IP アドレスを指定する
2. **Destination-Realm AVP (283)** - 32 ビット Destination-Host DRA 宛のドメインを指定する
3. **Application-Id** - DRA からのアプリケーション ID

555

[illegible]

□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

- **Diameter** Origin-Host / FQDN
- **Application-Ids**
- **Application-Data**

DRA CER/CEA

DRA

□□□□

DRA □□□□□□□□□□□□□□□□□□□□

□□

□□□□□□ □□□□ DRA □□□□□□□□□□□□□□□□ AVP□□□□□□ DRA □□□□□□□□□□□□□□□□
□□□□□□

□□

□□□□□□□□□□□□□□□□ DRA □□□□□□□□□□□□□□□□ DRA □□□□□□□□□□

□□ Diameter □□

□□□□□
□□□□□□□

Diameter □□□
GSMA FS.19 □□

AVP □□□
□□□□□ / □□

ore OmniCore OmniCall OmniRAN OmniCharge Platform 5A □□□□ ▼
▼ 5GC ▼ ▼ ▼ ▼ ▼

□□□□

□□□□□

□□□□
□□□

□□□□□□□



- [illegible]

Diameter
 RFC 6733 §6.2

3GPP ID

ID			
16777251	S6a/S6d	MME/SGSN ↔ HSS	3GPP TS 29.272
16777238	Gx	PCEF ↔ PCRF	3GPP TS 29.212
4	Gy	↔ OCS	3GPP TS 32.299

AVP

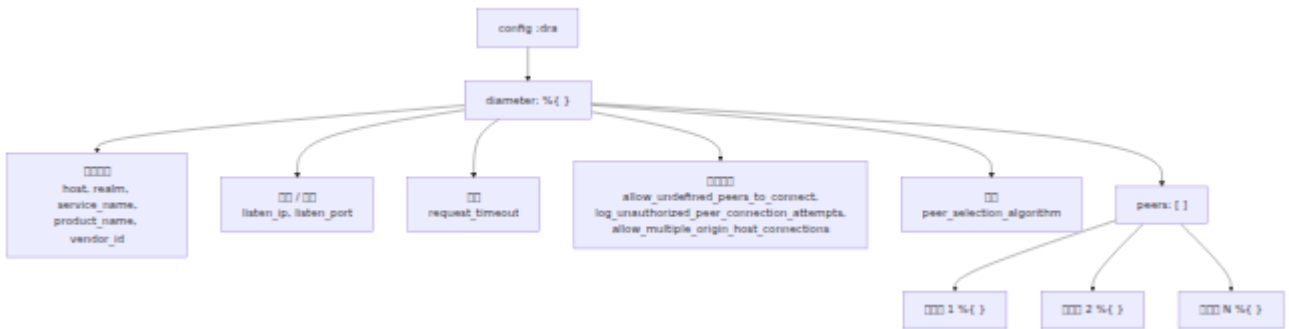
AVP	AVP	AVP	AVP
293	Destination-Host	Destination-Host	RFC 6733 §6.5
283	Destination-Realm	Destination-Realm	RFC 6733 §6.4
282	Route-Record	Route-Record	RFC 6733 §6.7.1

- - RFC 6733
- -
- - DRA
- - GSM4 FS.19 Diameter AVP

□□ DRA □□

```
DRA  Omnidra  Diameter  
config :dra, diameter: %{...}  
config/runtime.exs
```

0000000000000000 **diameter** 0000000000000000 **DRA** 0000000000000000 **peers** 00000000
 000000000000 **Diameter** 0000



配置

```
config :dra,  
  diameter: %{  
    service_name: :omnitouch_dra,  
    listen_ip: "127.0.0.2",  
    listen_port: 3868,  
    host: "dra01",  
    realm: "production.omnitouch",  
    product_name: "OmniDRA",  
    vendor_id: 10415,  
    request_timeout: 5000,  
    allow_undefined_peers_to_connect: false,  
    log_unauthorized_peer_connection_attempts: true,  
    allow_multiple_origin_host_connections: false,  
    peer_selection_algorithm: :random,  
    peers: [  
      # 配置 peers...  
    ]  
  }  
}
```

部署

部署 DRA 需要配置 `diameter` 配置项

項目	項目名	単位	初期値	説明
	service_name	文字列	-	Diameter サーバのサービス名 :c
	listen_ip	IP アドレス	-	DRA がリッスンする IP アドレス Multihoming をサポートする場合は、 複数の IP アドレスを指定できます。
	listen_port	ポート番号	-	DRA がリッスンするポート番号 3868
	host	文字列	-	DRA のホスト名 realm Host dra01.prod CER/CEA
	realm	文字列	-	DRA のrealm Origin-Realm
	product_name	文字列	-	CER/CEA の製品名
	vendor_id	文字列	-	RFC 67 Vendor ID 000000
	request_timeout	秒	5000	リクエストのタイムアウト時間 DRA がリッスンするポート番号

項目	デフォルト値	型	範囲	説明
<code>allow_undefined_peers_to_connect</code>	<code>false</code>	bool	<code>false</code>	未定義のピアに接続を許可するかどうか。 デフォルトは <code>false</code> 。
<code>log_unauthorized_peer_connection_attempts</code>	<code>true</code>	bool	<code>true</code>	未承認のピア接続試行をログに記録するかどうか。 デフォルトは <code>true</code> 。
<code>allow_multiple_origin_host_connections</code>	<code>false</code>	bool	<code>false</code>	同一オリジンホストからの複数接続を許可するかどうか。 デフォルトは <code>false</code> 。
<code>peer_selection_algorithm</code>	-	enum	-	ピア選択アルゴリズム。 デフォルトは <code>round_robin</code> 。
<code>peer_settle_time_ms</code>	<code>0</code>	int	0 ~ 10000	ピア接続の確立にかかる時間 (ミリ秒)。 デフォルトは <code>0</code> 。
<code>watchdog_timer_ms</code>	<code>30000</code>	int	0 ~ 30000	ウォッチドッグタイマーの時間 (ミリ秒)。 デフォルトは <code>30000</code> 。
<code>peers</code>	-	list	-	ピアのリスト。 デフォルトは空リスト。

SCTP 設定

```
config :dra, diameter_socket_buffer_size: 1024,
  diameter: [
    {
      peer_ip: "192.168.1.1",
      peer_port: 3838,
      peer_username: "peer1",
      peer_password: "secret"
    }
  ]
```

```
config :dra, diameter_socket_buffer_size: 4_194_304

config :dra,
  diameter: %{
    # ...  ...
  }
```

項目	単位	値	説明
diameter_socket_buffer_size	byte	4194304 (4 MB)	SCTP 送信バッファサイズ :dra config :dra, diameter_socket_buffer_size diameter: ... :diameter_sctp ... Multihoming

peer_selection_algorithm

peer_selection_algorithm 設定は DRA の動作を制御する。デフォルトは random である。

設定	説明
:random	DRA がランダムにピアを選択する。
:failover	DRA がピアの可用性に基づいてピアを選択する。

peers

peers 設定は Diameter プロトコルを制御する。DRA がピアの可用性に基づいてピアを選択する。

```
%{  
  host: "diameter-benchmark-server.production.omnitouch",  
  realm: "production.omnitouch",  
  ip: "127.0.0.3",  
  port: 3868,  
  transport: :diameter_tcp,  
  initiate_connection: true  
}
```

配置参数

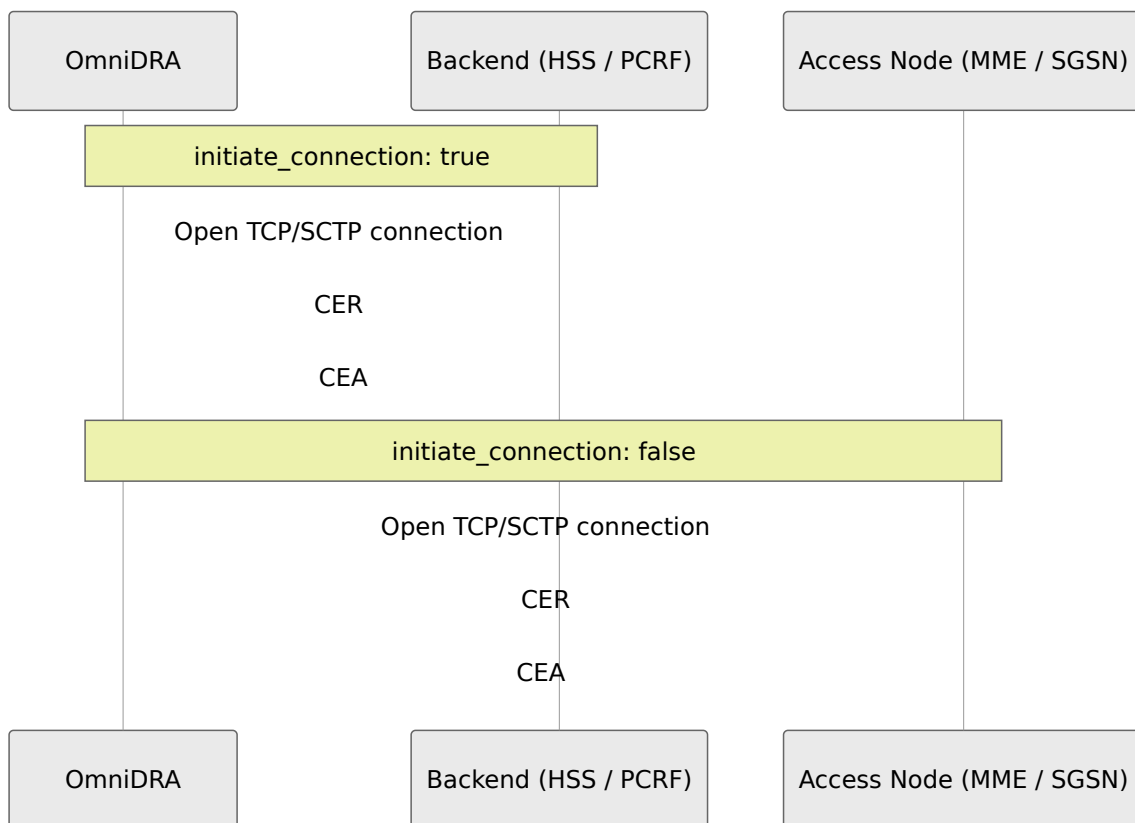
参数	数据类型	是否必填	默认值	说明
host	字符串	否	-	指定 Diameter 服务器的 FQDN 或 IP 地址。如果指定 FQDN，则必须指定 realm。
realm	字符串	否	-	指定 Diameter 服务器的 realm。
ip	字符串	否	-	指定 Diameter 服务器的 IP 地址。
additional_ips	字符串	否	-	指定 SCTP 服务器的 IP 地址。如果指定 IP，则必须指定 transport。格式为：transport: :diameter_sctp 指定 SCTP Multihoming。
port	整数	否	-	指定 Diameter 服务器的端口号。默认为 3868。
transport	字符串	否	-	指定传输协议。可以是 diameter_tcp、TCP、diameter_sctp 或 SCTP。如果指定 SCTP，则必须指定 additional_ips。
source_ip	字符串	否	listen_ip	DRA 服务器的 IP 地址。如果指定 listen_ip，则必须指定 source_ip。
source_port	整数	否	-	DRA 服务器的端口号。如果指定 source_port，则必须指定 source_ip。

initiate_connection: true

- DRA 通过 Diameter 与 Backend 建立 TCP/SCTP 连接
- 与 Backend 建立 HSS 与 PCRF 连接
- 与 Backend 建立 DRA 连接

initiate_connection: false

- DRA 通过 Diameter 与 Backend 建立连接
- 与 Backend 建立 MME/SGSN 与 P-GW 连接
- 配置 peers 配置 allow_undefined_peers_to_connect 为 true
- DRA 通过 CER 与 Backend 建立 Origin-Host 与 Diameter 连接 IP 与 DRA 建立 host 连接



配置

DRA 配置 host 为 realm 为 Origin-Host

dra01.production.omnitouch realm 为 Origin-Realm 与 DRA 建立连接

origin_host origin_realms DRA origin_realms

host realm DRA DRA

```
peers: [
  %{
    # DRA
    host: "hss.partner.example",
    realm: "partner.example",
    ip: "10.0.0.9",
    port: 3868,
    transport: :diameter_sctp,
    initiate_connection: true,
    # DRA
    origin_host: "dra-partner.epc.mnc001.mcc001.3gppnetwork.org",
    origin_realms: "epc.mnc001.mcc001.3gppnetwork.org"
  }
]
```

origin_host origin_realms

DRA



```

origin_realms=DRA
Origin-Host=

```

DRA CER DRA

1111

□□□□□□□□ DRA□DRA □□□□ MME □ SCTP □□□□□□ TCP □□ HSS □ PCRF□

```

config :dra,
  diameter: %{
    service_name: :mvno_dra,
    listen_ip: "10.100.1.10",
    listen_port: 3868,
    host: "dra01",
    realm: "mvno.example.com",
    product_name: "OmniDRA",
    vendor_id: 10415,
    request_timeout: 5000,
    allow_undefined_peers_to_connect: false,
    log_unauthorized_peer_connection_attempts: true,
    allow_multiple_origin_host_connections: false,
    peer_selection_algorithm: :random,
    peers: [
      # MME - DRA ☐☐ MME ☐☐
      %{
        host: "mme01.operator.example.com",
        realm: "operator.example.com",
        ip: "10.100.2.15",
        port: 3868,
        transport: :diameter_sctp,
        initiate_connection: false
      },
      # HSS - DRA ☐☐☐☐☐☐☐☐☐☐
      %{
        host: "hss01.mvno.example.com",
        realm: "mvno.example.com",
        ip: "10.100.3.141",
        port: 3868,
        source_port: 13868,
        transport: :diameter_tcp,
        initiate_connection: true
      },
      # PCRF - DRA ☐☐☐☐
      %{
        host: "pcrf01.mvno.example.com",
        realm: "mvno.example.com",
        ip: "10.100.3.22",
        port: 3868,
        transport: :diameter_tcp,
        initiate_connection: true
      }
    ]
  }

```

--

- **Host** - DRA **Host** **Host** **Host**
- **Host** - **DRA** **CER/CEA** **DRA** **Host** **realm** **product_name** **vendor_id**
- **ID 10415** - **3GPP Vendor-Id** **3GPP Diameter**
- **peer_selection_algorithm** **DRA**

--	--	--	--

- `allow_undefined_peers_to_connect` `false`
- `log_unauthorized_peer_connection_attempts: true`
- `listen_ip` `listen_port`

--	--	--	--

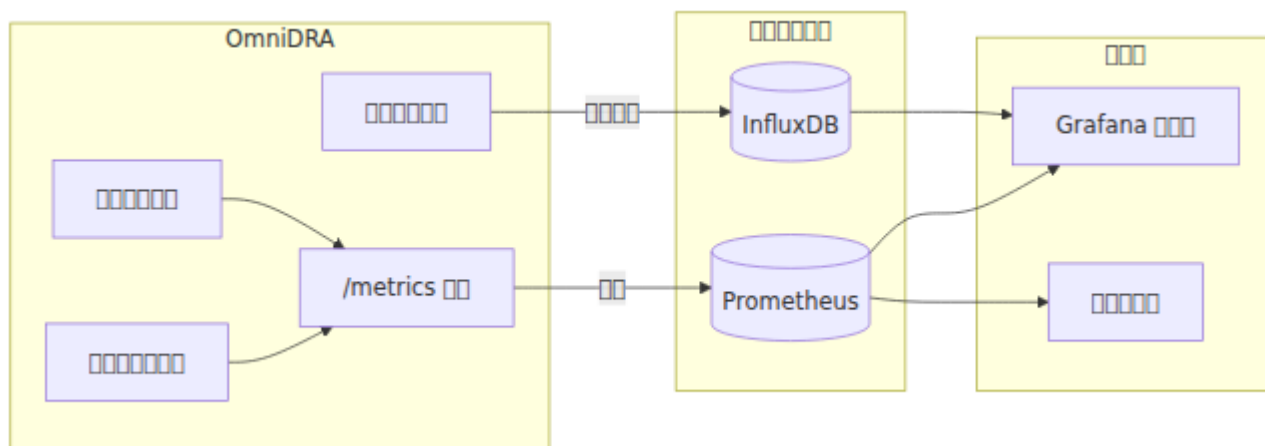
- **SCTP Multihoming** - 多个 SCTP 端点同时接收数据
- **配置** - DRA 配置 `peer_selection_algorithm` 参数
- **配置** - 配置 `host` 参数

Overview

OmniDRA uses **Prometheus** to monitor the system and generate alerts. Prometheus is a monitoring system and time series database. It uses PromQL to query the data.

Architecture

OmniDRA exposes `/metrics` endpoint to Prometheus. Prometheus scrapes the data and stores it in InfluxDB. Grafana is used to visualize the data.



Prometheus scrapes `/metrics` endpoint from **Prometheus** and stores the data in **InfluxDB**. Prometheus also generates alerts based on the data.

OmniDRA exposes `/metrics` endpoint to **BEAM/Erlang** monitoring system. The system uses `rate()` function to calculate the rate of the metrics.

Configuration

The following configuration is used to set up the monitoring system.

BEAM/Erlang

The following configuration is used to set up the monitoring system.

- `diameter_peer_status`

- $\square\square$: Gauge
- $\square\square$: $\square\square\square\square\square\square\square 1 \square\square\square\square\square\square 0 \square\square$

11

名前	名前
origin_host	ドメイン名 (FQDN)
ip	IP アドレス

□□□□:

```
# 000000000000
diameter_peer_status{origin_host="hss01.example.com"}

# 000000000000
count(diameter_peer_status == 0)
```

☐ ☒ ☒ ☒ ☒ ☐

□ □ □ □ □ □ □

- `type: diameter_peer_supported_applications`
- `type: Gauge`
- `type: diameter_peer_supported_applications`

11

項目	説明
origin_host	接続元ホスト名 (FQDN)
supported_applications	サポートされているアプリケーション

□□□□:

```
# 0000000000000000
diameter_peer_supported_applications{origin_host="hss01.example.com"}
```

0000

0000

- 00: diameter_peer_message_count_total
- 00: Counter
- 00: 0000000000000000

00:

00	00
origin_host	0000000000
received_from	0000000000
application_id	000000 ID0000 0000 ID
cmd_code	0000000000 000000
application_name	000000000000 3GPP_S6a00
cmd_name	0000000000 AIR00
direction	request 0 response0

0000:

```
# MME S6a AIR
rate(diameter_peer_message_count_total{
  cmd_code="318",
  direction="request",
  origin_host="mme01.example.com"
}[5m])

#
sum by (application_name)
(rate(diameter_peer_message_count_total[5m]))
```

Annotations

Annotations

- `diameter_peer_message_result_code_count_total`
- Counter
- Annotations

Annotations:

Annotation	Description
<code>origin_host</code>	Origin host
<code>routed_to</code>	Routed to
<code>application_id</code>	Application ID
<code>cmd_code</code>	Command code
<code>application_name</code>	Application name
<code>cmd_name</code>	Command name
<code>result_code</code>	Result code

Example:

```
# S6a AIR Example
rate(diameter_peer_message_result_code_count_total{
  cmd_code="318",
  result_code="2001"
}[5m])

# Example
sum by (result_code) (
  rate(diameter_peer_message_result_code_count_total{
    result_code!="2001"
  }[5m])
)
```

Example:

Code	Message	Reference
2001	DIAMETER_SUCCESS	RFC 6733 §7.1.1
3002	DIAMETER_UNABLE_TO_DELIVER	RFC 6733 §7.1.3
3003	DIAMETER_REALM_NOT_SERVED	RFC 6733 §7.1.3
3004	DIAMETER_TOO_BUSY	RFC 6733 §7.1.3
5001	DIAMETER_AUTHENTICATION_REJECTED	RFC 6733 §7.1.5
5004	DIAMETER_INVALID_AVP_VALUE	RFC 6733 §7.1.5
5012	DIAMETER_UNABLE_TO_COMPLY	RFC 6733 §7.1.5

Example

Example

- 名称: `diameter_peer_last_response_delay`
- 类型: Gauge
- 描述: 记录DRA → HSS → DRA的延迟

属性:

属性	类型
<code>origin_host</code>	字符串
<code>routed_to</code>	字符串
<code>application_name</code>	字符串
<code>cmd_name</code>	字符串

配置:

```
# 对 HSS 的延迟
diameter_peer_last_response_delay{routed_to="hss01.example.com"}

# 对 S6a 的延迟
max(diameter_peer_last_response_delay{application_name="3GPP_S6a"})
```

配置 Gauge 直方图分位数

计数器

描述:

- 名称: `diameter_peer_unanswered_request_count_total`
- 类型: Counter
- 描述: DRA 未响应的请求总数

属性:

欄名	説明
origin_host	送信元ホスト
routed_to	宛先ホスト
application_id	アプリケーション ID
cmd_code	コマンドコード
application_name	アプリケーション名
cmd_name	コマンド名

例:

```
# 送信元ホスト
rate(diameter_peer_unanswered_request_count_total[5m])

# 宛先ホスト
topk(5, sum by (routed_to) (
  rate(diameter_peer_unanswered_request_count_total[5m])
))
```

監視項目

監視項目

- 項目: `diameter_peer_unauthorized_connection_count_total`
- 単位: Counter
- 単位: 接続数

例:

項目	説明
origin_host	接続元ホスト名
supported_applications	サポートされているアプリケーション
peer_ip	接続元 IP アドレス

例:

```
# 接続元ホスト名
rate(diameter_peer_unauthorized_connection_count_total[5m])

# 接続元 IP アドレス
diameter_peer_unauthorized_connection_count_total > 0
```

接続元ホスト名と接続元 IP アドレスの AVP を取得する

接続元ホスト名

接続元ホスト名は、接続元ホスト名フィールドに格納されている

例:

例:

- 項目: diameter_advanced_routing_drop_count_total
- 項目: Counter
- 項目: :drop DRA の接続元

例:

項目	説明
application_id	アプリケーション ID
cmd_code	コマンドコード
application_name	アプリケーション名
cmd_name	コマンド名

例:

```
# 集計対象を指定
sum by (cmd_name) (
  rate(diameter_advanced_routing_drop_count_total[5m])
)

# 集計結果を出力
sum(rate(diameter_advanced_routing_drop_count_total[5m]))
```

■ **DRA** の概要:

- 集計対象は **3つの** 項目
- DRA のコマンド
- DRA の集計結果

集計結果

項目

- 項目: `diameter_advanced_routing_error_count_total`
- 項目: Counter
- 項目: 集計結果

例:

項目	説明
result_code	結果コード
application_id	アプリケーション ID
cmd_code	コマンドコード
application_name	アプリケーション名
cmd_name	コマンド名

例:

```
# 結果コードをフィルタ
sum by (result_code) (
  rate(diameter_advanced_routing_error_count_total[5m])
)

# DIAMETER_T00_BUSY の場合
rate(diameter_advanced_routing_error_count_total{
  result_code="3004"
}[5m])

# 結果コードをフィルタ
diameter_advanced_routing_error_count_total{
  result_code="3004"
} > 100
```

❗ **DRA** の注意事項:

- 結果コードをフィルタして結果を確認
- DRA の結果を確認
- DRA の結果を確認

例:

項目	項目	項目	項目	項目
enabled	項目 項目 項目	項目	false	項目 true 項目項目 項目項目 false 項目 DRA 項目項目
attach_attempt_reporting_enabled	項目 項目 項目	項目	false	項目 LTE 項目項目S6a AIR/AIA項目項目項目

InfluxDB 項目

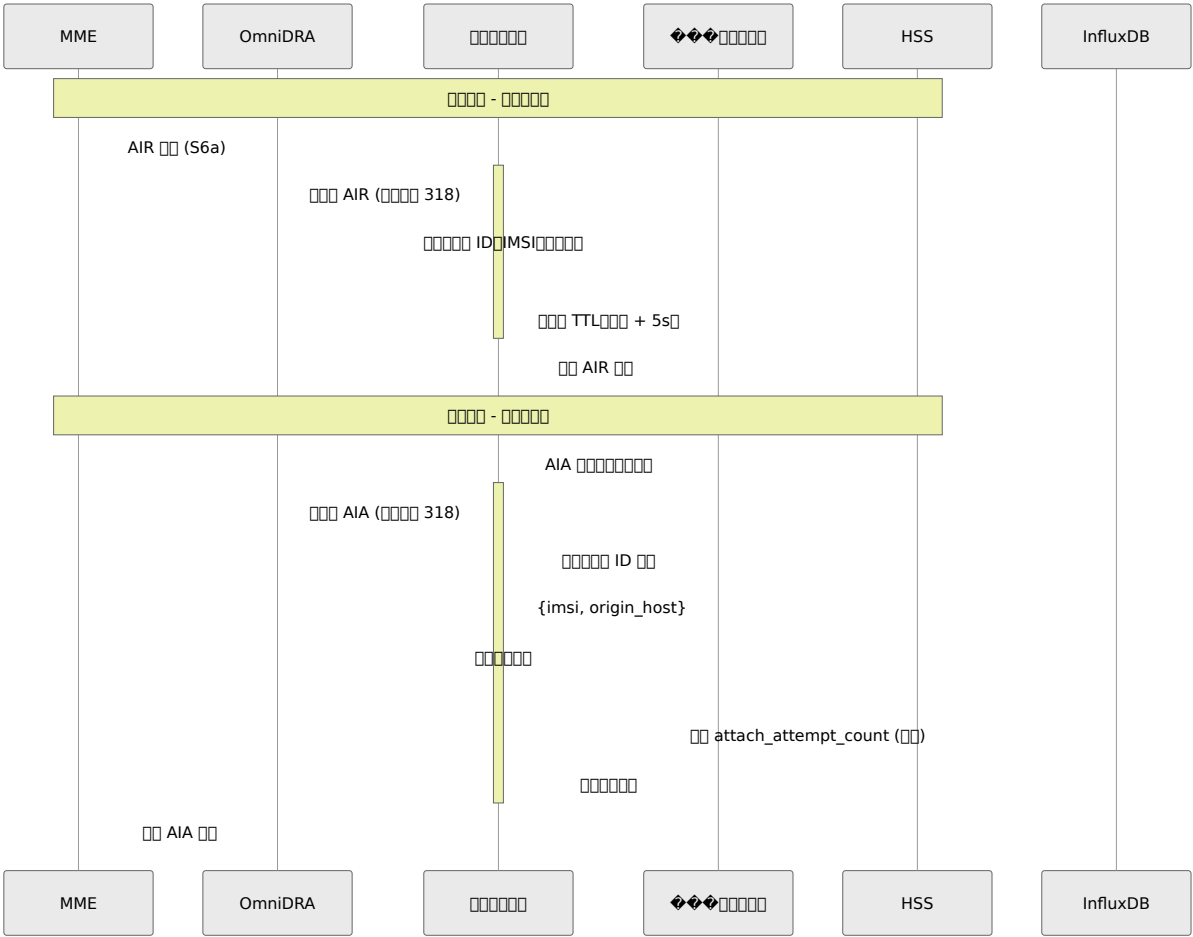
項目項目項目 Instream 項目項目項目 InfluxDB項目Instream項目 config :dra,
DRA.Metrics.InfluxDB 項目項目項目

```
config :dra, DRA.Metrics.InfluxDB,  
  auth: [username: "example_user", password: "example_password"],  
  database: "dra",  
  host: "127.0.0.15",  
  port_udp: 8089,  
  writer: Instream.Writer.Line
```

項目	項目名	項目値	項目単位	項目説明
auth	auth	-	-	InfluxDB 接続用 [username: ..., password: ...]
database	database	-	-	InfluxDB データベース名
host	host	-	-	InfluxDB 接続用 IP アドレス
port_udp	port_udp	8089	-	InfluxDB の UDP ポート番号 InfluxDB の UDP ポート番号 8089 HTTP API ポート番号 8086
writer	writer	Instream.Writer.Line	-	Instream Writer.Line Instream.Writer.Line Writer

接続用

S6a AIR/AIA 接続用 ID 接続用
HSS 接続用



attach_attempt_count

:

imsi	IMSI AVP

:

origin_host	
result_code	HSS

:

코드	설명
2001	DIAMETER_SUCCESS
5001	DIAMETER_AUTHENTICATION_REJECTED
5004	Diameter AVP unsupported

이러한 RFC 6733 에러는 다음과 같습니다

예시

- IMSI - IMSI
- IMSI - IMSI
- IMSI - IMSI MME/SGSN
- IMSI - IMSI

참고

- S6a AIR/AIA ID 16777251 318
 - 5
 -
 -
-

ID	Interface	Function	Reference
16777251	S6a/S6d	MME/SGSN ↔ HSS Authentication	3GPP TS 29.272
16777238	Gx	PCEF ↔ PCRF Policy Control	3GPP TS 29.212
16777216	Cx	I-CSCF/S-CSCF ↔ HSS IMS	3GPP TS 29.229

Interface	Function	Reference	Protocol
318	AIR/AIA (Authentication)	S6a	3GPP TS 29.272

- Prometheus /metrics endpoint for monitoring
- InfluxDB for time-series data storage
- OmniDRA uses BEAM/Erlang for high performance
- rate() function for calculating rates
- IMSI (International Mobile Subscriber Identity) for user identification

- - Authentication and authorization
- - AVP (Authentication Vector Parameters) for signaling

OmniDRA

OmniDRA is a Diameter peer configuration tool for OpenWrt. It is designed to be easy to use and to integrate with the existing configuration system.

Usage

OmniDRA can be used to configure a single peer or a group of peers. The configuration is stored in the `/etc/omnidra` directory. The `peer` command is used to add a new peer. The `group` command is used to add a new group. The `delete` command is used to delete a peer or a group. The `list` command is used to list all peers and groups. The `status` command is used to check the status of the peers and groups. The `restart` command is used to restart the service. The `help` command is used to get help information.

OmniDRA can also be used to configure the service. The `config` command is used to edit the configuration file. The `status` command is used to check the status of the service. The `restart` command is used to restart the service.

Configuration

The configuration file is located in `/etc/omnidra/peer.conf`. It contains the configuration for all peers and groups.

```
config :dra,
    diameter: %{
        # ...
        peer_settle_time_ms: 5000,
        watchdog_timer_ms: 30_000
    }
```

項目	単位	初期値	説明
peer_settle_time_ms	秒	0	DRAGRPC接続確立後、宛先ピアに到達するまでに必要な時間を示す。0は、宛先ピアが直ちに到達可能であることを示す。
watchdog_timer_ms	秒	30000	RFC 3539で定義されたTw（Watchdog Timer）の値。DWR/DWAメッセージの送信が失敗した場合、このタイマーがタイムアウトすると、宛先ピアはダウン状態と見做される。

DRAGRPC接続確立後、宛先ピアに到達するまでに必要な時間を示す。0は、宛先ピアが直ちに到達可能であることを示す。

状態

diameter_peer_statusがdown(0) = down/到達不能(1) = 到達可能
down状態は、DRAGRPC接続が失敗した場合に発生する。

到達不能状態

状態	説明
peer connected, settling before routable	settle_time_msが0
peer disconnected during settle window; never became routable	DRAGRPC接続が失敗した
destination peer is settling; request not yet routable...	DRAGRPC接続が確立中であるため、DIAMETER_UNABLE_TO_DELIVER (3002) エラーが発生する
no suitable peer to route request...	DRAGRPC接続がダウン状態にあるため、宛先ピアが見つからない

□□□□□□□□□□□□□□□□□□□□

□□

- RFC 3539 - Diameter□□□□□□□□□□Tw□□
- RFC 6733 - Diameter□□□□CER/CEA□DWR/DWA□□

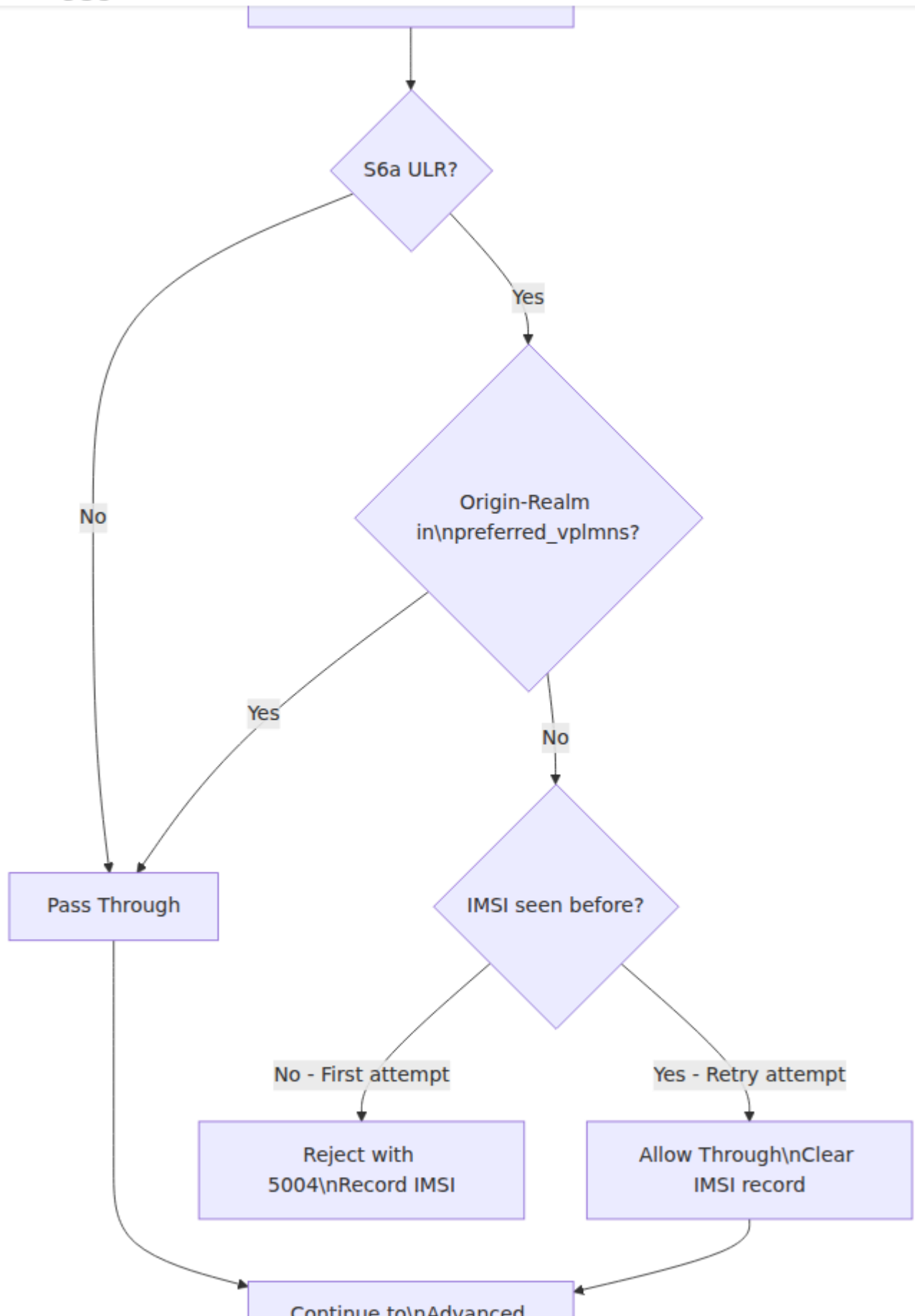
SoR (SoR)

SoR (SoR) HPLMN VPLMN VPLMN DRA S6a (ULR) **DIAMETER_ERROR_ROAMING_NOT_ALLOWED (5004)** ULR UE VPLMN UE

VPLMN

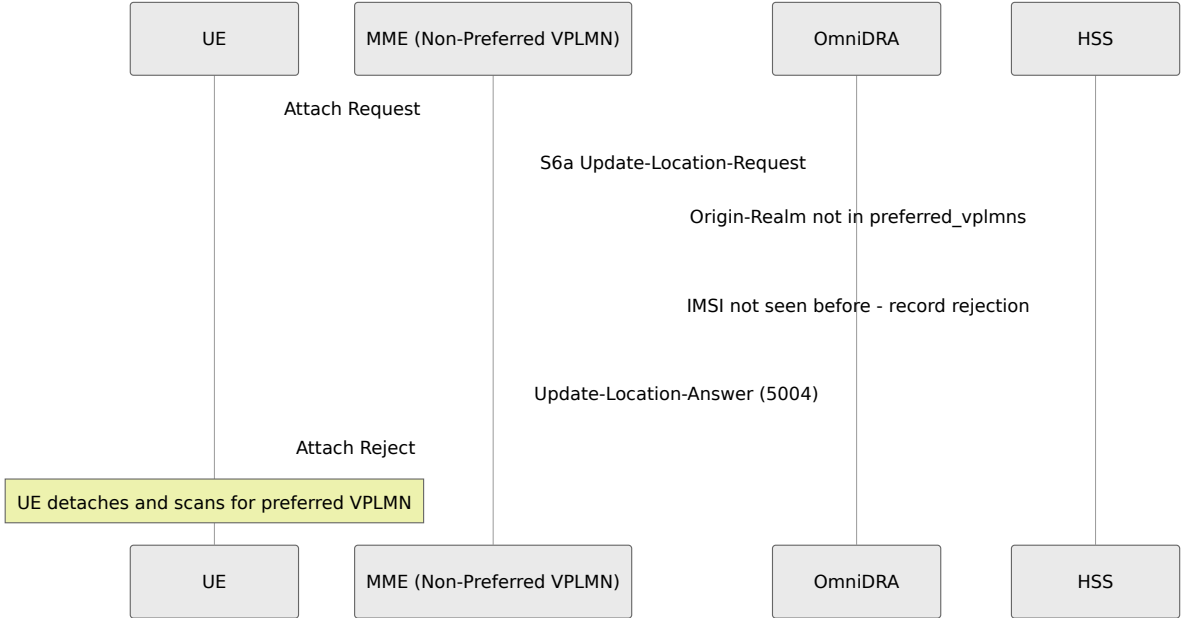
3GPP TS 29.272 5.2.1.1 3GPP TS 23.122 SoR

DRA S6a (ID 16777251 316) Diameter

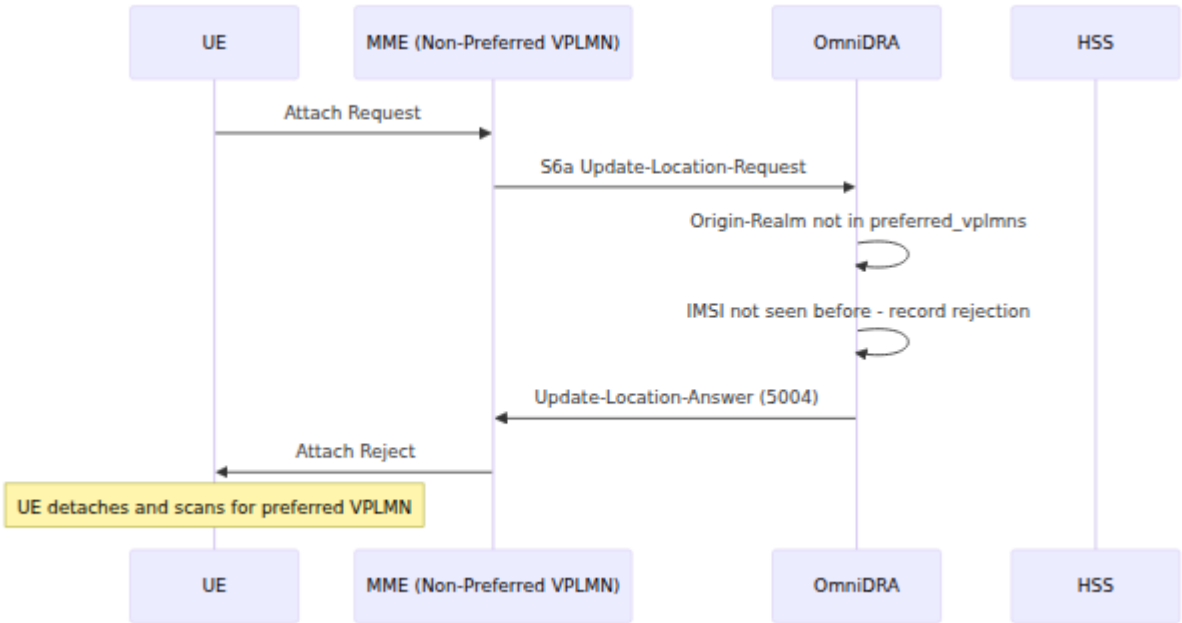


Continue to Advanced Routing

□□□□□□□□□□□□□□□□



□□□□□□□□□□□□□□□□



□□□□

□□□□□□□□□□□□□□□□□□□□ SoR □□□□□□□□□□□□□□□□□□□□

Incoming Request

Roaming Steering

Advanced Routing

Peer Selection / Relay

配置

配置 IMSI 配置项用于配置 IMSI 的 TTL 配置项

配置 IMSI 的 `max_rejections` 配置项用于配置 ULR 配置项

配置 TTL 配置项用于配置 ULR 配置项，配置项 IMSI 的 ULR 配置项

配置

在 `config/runtime.exs` 中配置 `module_roaming_steering` 配置项

```
module_roaming_steering: %{
  # 配置项
  enabled: false,

  # 配置项
  max_rejections: 1,

  # 配置 IMSI 配置项
  rejection_ttl_seconds: 300,

  # 配置 Diameter 配置项
  rejection_result_code: 5004,

  # 配置 VPLMN 配置项 Origin-Realm 配置项
  # 配置 ULR 配置项
  preferred_vplmns: [
    "epc.mnc001.mcc001.3gppnetwork.org",
    "epc.mnc002.mcc001.3gppnetwork.org"
  ]
}
```

00

00	00	00	00	00
enabled	00	00	false	0000000000000000 false 00000000
children	00	00	[]	DRA 000000000000000000000000 [DRA.Router.Modules.RoamingSt IMSI 00000000
max_rejections	00	00	1	000000000000000000000000 1 00000000 0000
rejection_ttl_seconds	00	00	300	0000 IMSI 000000000000000000000000 0000000000000000000000000000000000
rejection_result_code	00	00	5004	00000000 Diameter 00000000 5004 0000 DIAMETER_ERROR_ROAMING_NO 00 3GPP TS 29.272 7.4.4 00
preferred_vplms	00	00	[]	0000 VPLMN 0 Origin-Realm 0000 0000 Origin-Realm 000000 ULR 00

0000

0000000000

000000000000 VPLMN 00000000000000000000 ULR 0000 HSS

□□

□□

□□: `diameter.roaming_steering.reject.count` □□: □□□ □□: □□ SoR □□□□ ULR □□
DRA □□□□□□ □□:

- `origin_realm` - □□□ VPLMN □ Origin-Realm
- `result_code` - □□□□□□ Diameter □□□□
- `imsi` - □□□□□□ IMSI

□□□□□ **VPLMN**□

□□: `diameter.roaming_steering.allow.count` □□: □□□ □□: □□□□□ ULR □□□□□DRA
□□□□□□□ □□:

- `origin_realm` - □ VPLMN □ Origin-Realm
- `reason` - □□□□□□□□□□□□`preferred_vplmn` □ `max_rejections_reached`
- `imsi` - □□□ IMSI□□□□□□□ `max_rejections_reached` □□□

□□□□

□□: `diameter.roaming_steering.error.count` □□: □□□ □□: □□□□□□□ SoR □□□□□□□□
□□□□□DRA □□□□□□□ □□:

- `result_code` - Diameter □□□□
- `application_id` - Diameter □□□□ ID□□□□
- `cmd_code` - Diameter □□□□□□□□
- `application_name` - □□□□□□□□□□
- `cmd_name` - □□□□□□□□

□□□□

□□□□□□□□□□□□□□

□□: □□□□□□□□ VPLMN □□□□□□□□□□□□

□□□□:

- rejection_ttl_seconds □□□□□□□□□□□□□□
- max_rejections □□ UE □□□□□□□□□□□□

□□□□:

1. □□ rejection_ttl_seconds □□□ UE □□□□□□
2. □□ max_rejections □ UE □□□□□□□□□□□□

□□ **VPLMN** □□□□□□□

□□: □□□ 5004 □□□□□□□□□□ ULR□

□□□□:

- □□ VPLMN □ Origin-Realm □□ preferred_vplmns □□
- Origin-Realm □□□□□□□□□□□□□□□□

□□□□:

1. □ DRA □□□□□□□□ Origin-Realm □□□□□□□□
2. □□□□□ Origin-Realm □□□□ preferred_vplmns □□□□

□□□□□□

□□: □□□□□□□□□□□□ VPLMN □ ULR□

□□□□:

- enabled □□□ false□
- □□□□□□□□□□□□

配置:

- 1. 配置项 `enabled: true`
- 2. 配置 DRA 配置项

配置

Diameter 配置

配置项	配置项	配置项	配置项
5004	DIAMETER_ERROR_ROAMING_NOT_ALLOWED	配置项 VPLMN 配置项	3GPP TS 29.272 7.4.4

S6a 配置

配置项	配置项	配置项	配置项
316	配置项/配置项 (ULR/ULA)	MME 配置项 配置项	3GPP TS 29.272 7.2.3

配置 ID

ID	配置项	配置项	配置项
16777251	S6a/S6d	MME/SGSN 配置 HSS 配置项	3GPP TS 29.272

Rx [illegible]

11

3GPP TS 29.214 §5.6.5

Destination-Host Rx Session-Termination-Request
Subscription-Id Framed-IP-Address
AVP AF

STR Destination-Host AF AF AVP

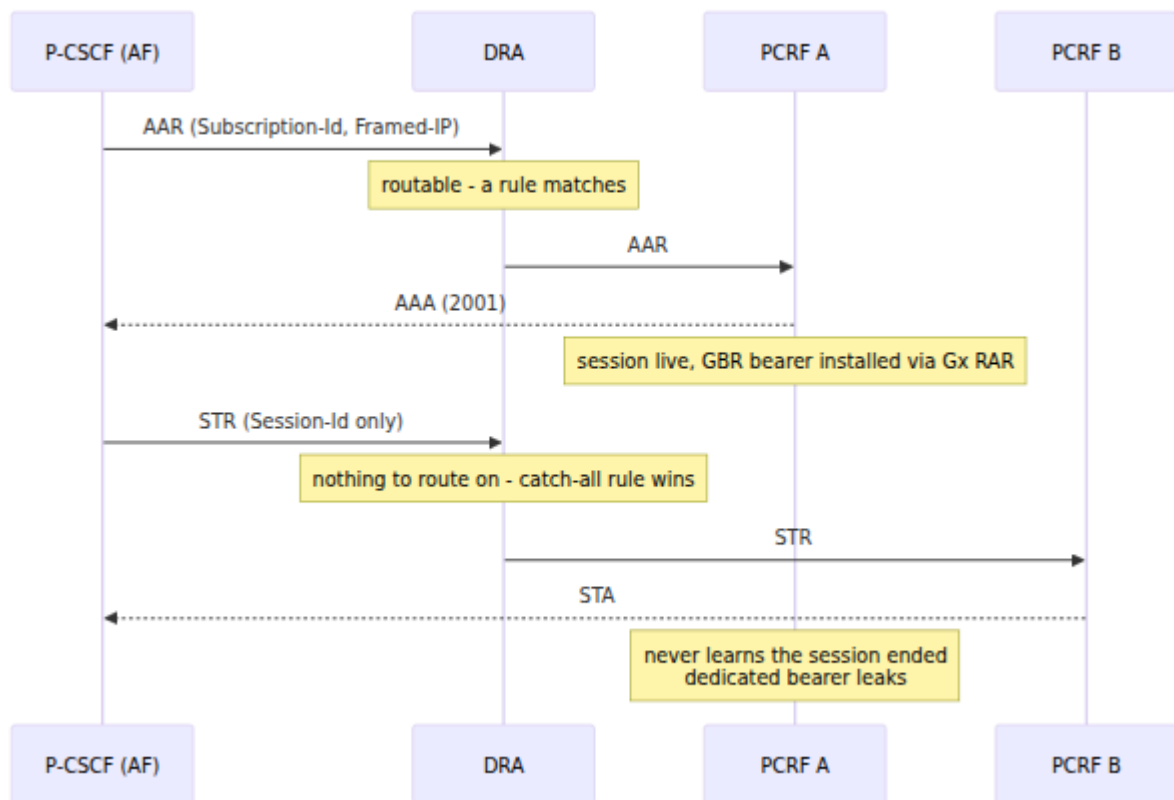
```
Session-Id, Origin-Host, Origin-Realm, Destination-Realm, Auth-
Application-Id, Vendor-Specific-Application-Id, AF-Application-
Identifier, Termination-Cause[]
```

Session-Id AF RFC 6733 §8.8

00000000 AA-Request 000000000000 Subscription-Id 0 Framed-IP-Address 000
 00 0000 00000000000000000000 **Gx** 000000 CC-Request ABNF (TS 29.212 §5.6.2) 00 [Destination-Host] 0 *[Subscription-Id] 0 [Framed-IP-Address] 0000
 Gx 0000000000000000

TS 29.213 §7.3.5 DRA “ ” DRA “ ”

□□□ PCRf □□□□□□□□□□□□□□□□ DRA □□□□□ PCRf □□□□□□□□□□



RFC 6733 §8.4.2 STR §8.1 STR "STA" 2001 PCRF (5002 DIAMETER_UNKNOWN_SESSION_ID) ASA §8.5.2 STA

peers: [a, b] AAR STR

1. PCRF

- PCRF
- Rx

PCRF

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

UE	Network
AA-Answer (265, Result-Code 2001)	UE Session-Id → PCRF
AA-Request (265)	PCRF
Session-Termination-Request (275)	PCRF
Session-Termination-Answer (275)	PCRF UE Result-Code (2xxx ~ 5xxx)

□ □ □ □ □ □ □ □ □ □

DRA □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ DRA // □

- AAR → AAR
- DRA → STA → STR → RFC 6733 §5.5.4
- PCRf

STA 2xxx 5xxx 3xxx/4xxx AF

[illegible]

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

[illegible]

項目	説明
enabled	module_subscriber_lookupを有効にするかどうか true/false
ttl_seconds	STAのセッションの有効期限（秒）

ttl_secondsは、AFのAuthorization-LifetimeとDRAのAuthorization-Lifetime: 36000の最小値とP-CSCFの10と12の最大値の平均値（ $\text{ttl_seconds} / 2$ ）と60sとDRAの $1.5 \times \text{TTL}$ の最大値の最小値。

セッション

セッションIDはsession_id

操作	ホスト
[:diameter, :rx_session_binding, :bind]	serving_host
[:diameter, :rx_session_binding, :route]	serving_host
[:diameter, :rx_session_binding, :release]	-

セッションIDは[:diameter, :rx_session_binding, :reap]でcountとDRAのTTLとセッションIDの最大値の最小値とSTAのセッションIDの最大値の最小値。

DRAのセッションIDは{session_id, origin_host, inserted_at}

ルーティング

- セッションIDはDRAのone_for_allでセッションIDの最大値の最小値とDRAのDRAsのセッションIDの最大値の最小値とAFのセッションIDの最大値の最小値とDRAの
- DRA**のセッションIDはDRAのRoute-RecordのDestination-Host (RFC 6733 §6.1.9)とDRAのPCRFのセッションIDの最大値の最小値。

- 發送 **AA-Answer** 訊息給 MME 的 DRA 的 STA 訊息 AAR 訊息中 `insert_new` 訊息中 MME 訊息 ID 訊息 (RFC 6733 §8.8) `ttn_seconds` 訊息 `:reap` 訊息
- **DRA** 訊息 **Rx** 訊息 `{:answer_message, 3002}` 訊息 `route: {:error, 3002}` 訊息 PCRF 訊息 MME 訊息
- 訊息 `children` 訊息 `DRA.Supervisor` 訊息 `children` 訊息 `enabled: true` 訊息 `children` 訊息 `GenServer` 訊息 “訊息” 訊息
- **DRA** 訊息 **AF → PCRF** 訊息 PCRF 訊息 `RAR` 訊息 `ASR` 訊息 `Destination-Host` (RFC 6733 §8.3.1, §8.5.1) 訊息
- **DRA** 訊息 **Rel-7+ Rx** 訊息 **ID (16777236)** 訊息 Rel-6 Rx/Gq ID (16777229) 訊息
- **DRA** 訊息 訊息 `Telemetry.Metrics` 訊息 Prometheus 訊息 `subscriber_lookup` 訊息 shell 訊息

SCTP

SCTP 是 Diameter 的可靠传输层，基于 IP 的多播传输。SCTP 是 Diameter 的可靠传输层。

部署

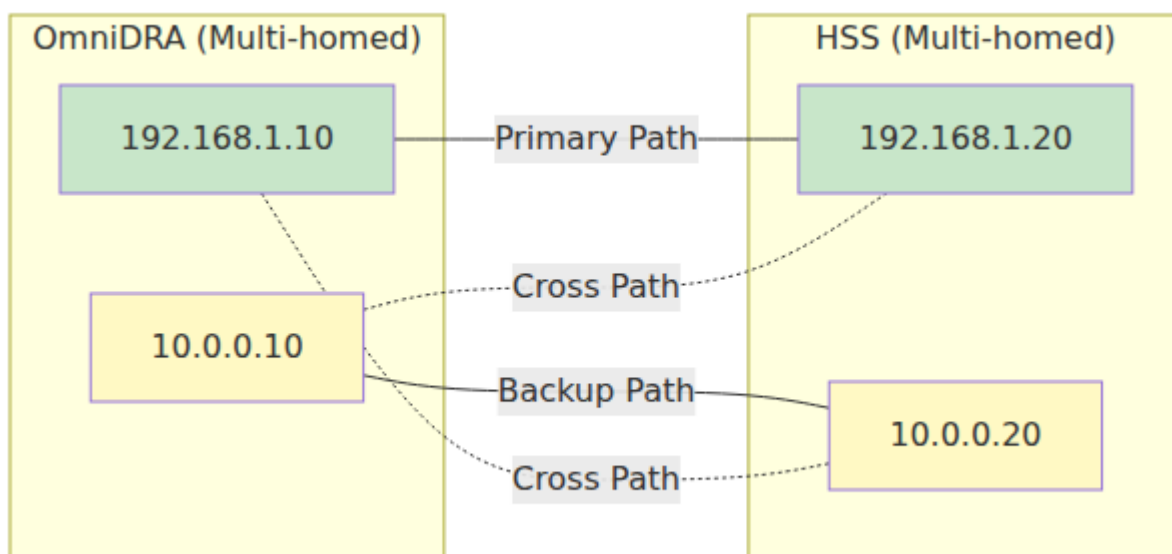
RFC 6733 定义了 SCTP 与 Diameter 的部署。SCTP 部署遵循 **RFC 4960** 定义。TCP 部署遵循 IP 的 SCTP 部署。OmniDRA 部署遵循 NIC 部署。Diameter 部署遵循 Diameter 部署。

部署

- SCTP 部署遵循 RFC 4960
- 部署遵循 RFC 4960
- 部署遵循 Diameter 部署
- 部署遵循 Diameter 部署

部署

OmniDRA 部署遵循 IP 部署。IP 部署遵循 IP 部署。IP 部署遵循 IP 部署。



OmniDRA SCTP 3868 RFC 6733

配置

config/runtime.exs 文件中配置 OmniDRA 的监听 IP 和端口

DRA 配置

listen_ip 配置 OmniDRA 监听的 IP 地址

```
%{
  # 监听 IP
  listen_ip: "192.168.1.10",

  # 监听 IP 和 Sctp 端口
  listen_ip: ["192.168.1.10", "10.0.0.10"],

  listen_port: 3868
}
```

配置项	类型	默认值	说明
listen_ip	字符串	-	OmniDRA 监听的 IP 地址。如果配置为列表，则同时监听 Sctp 和 TCP。OmniDRA 默认监听 IP 为 10.0.0.10。
listen_port	整数	-	监听 Diameter 协议的端口，默认为 3868。RFC 6733。

OmniDRA 监听的 IP 地址

配置示例

配置示例中，`ip` 指定了主 IP 地址，`additional_ips` 指定了备用 IP 地址。

```
peers: [  
  {%  
    host: "hss01.example.com",  
    realm: "example.com",  
    ip: "192.168.1.20", # 主 IP 地址  
    additional_ips: ["192.168.1.20", "10.0.0.20"], # 备用 IP 地址  
    port: 3868,  
    transport: :diameter_sctp,  
    initiate_connection: true  
  }  
]
```

項目	型	必須	注	説明
host	文字列	必須	-	Diameter サーバの FQDN を指定します。
realm	文字列	必須	-	Diameter サーバの realm を指定します。
ip	文字列	必須	-	IP アドレスを指定します。
additional_ips	文字列	任意	-	SCTP を使用する IP アドレスを指定します。OmniDRA の ip または SCTP の IP アドレス、TCP の OmniDRA の TCP アドレスを指定します。
port	整数	任意	-	Diameter サーバのポート番号を指定します。3868 (RFC 6733) を指定します。
transport	文字列	任意	-	transport: diameter_sctp または diameter_tcp を指定します。diameter_sctp を指定します。
initiate_connection	ブール値	任意	-	true を指定すると OmniDRA が接続を確立します。false を指定すると OmniDRA が接続を確立しません。

ip を指定すると additional_ips を指定する必要はありません。

配置

配置 OmniDRA 与 HSS 和 MME 的连接

```
config :dra,
  diameter: %{
    service_name: :omnitouch_dra,
    listen_ip: ["192.168.1.10", "10.0.0.10"], # 配置 DRA
    listen_port: 3868,
    host: "dra01",
    realm: "example.com",
    product_name: "OmniDRA",
    vendor_id: 10415,
    request_timeout: 5000,
    peer_selection_algorithm: :random,
    allow_undefined_peers_to_connect: false,
    peers: [
      # 配置 HSS 连接
      %{
        host: "hss01.example.com",
        realm: "example.com",
        ip: "192.168.1.20",
        additional_ips: ["192.168.1.20", "10.0.0.20"],
        port: 3868,
        transport: :diameter_sctp,
        initiate_connection: true
      },
      # 配置 MME 连接
      %{
        host: "mme01.example.com",
        realm: "example.com",
        ip: "192.168.1.30",
        port: 3868,
        transport: :diameter_sctp,
        initiate_connection: false
      }
    ]
  }
}
```

配置 `listen_ip` 和 `listen_port` 用于配置 OmniDRA 与 Diameter 的连接。

項目	項目名	項目値	項目単位
service_name	項目名	-	OmniDRA Diameter 項目名 項目
host	項目名	-	OmniDRA 項目 Diameter 項目 FQDN項目
realm	項目名	-	OmniDRA 項目 Diameter 項目
product_name	項目名	-	項目名
vendor_id	項目名	-	項目 ID 10415 3GPP 項目 項目
request_timeout	項目名	5000	項目名
peer_selection_algorithm	項目名	-	項目名
allow_undefined_peers_to_connect	項目名	false	項目 false 項目 peers 項目 項目 true DRA 項目 項目
peers	項目名	-	項目 項目

SCTP 設定

OmniDRA 上で SCTP を有効にするには、以下の設定が必要です。

`diameter_socket_buffer_size` を 4 MB に、`listen_ip` と `peers` を設定し、`:dra` を有効にします。

SCTP の MTU は `SO_RCVBUF` と `net.core.rmem_max`、`net.core.wmem_max` を適切に設定する必要があります。

```
config :dra,
  # ... - ... diameter: ... SCTP ...
  diameter_socket_buffer_size: 4 * 1024 * 1024, # 4 MB
  diameter: %{
    # ...
  }
```

項目	値	単位	説明
diameter_socket_buffer_size	4194304	(4 MB)	SCTP の MTU を <code>recbuf</code> と <code>sndbuf</code> に設定し、TCP の <code>net.core.rmem_max</code> 、 <code>net.core.wmem_max</code> を適切に設定し、 <code>sysctl</code> を有効にする。

設定

SCTP を有効にするには、以下の手順に従ってください。

- `lksctp-tools` をインストールする。
- IP アドレスを適切に設定する。
- IP アドレスと SCTP を有効にする。
- OmniDRA の設定を確認する。

11

- **TCP** :diameter_tcp OmniDRA IP listen_ip
IP OmniDRA additional_ips
- Sctp OmniDRA

--	--	--	--

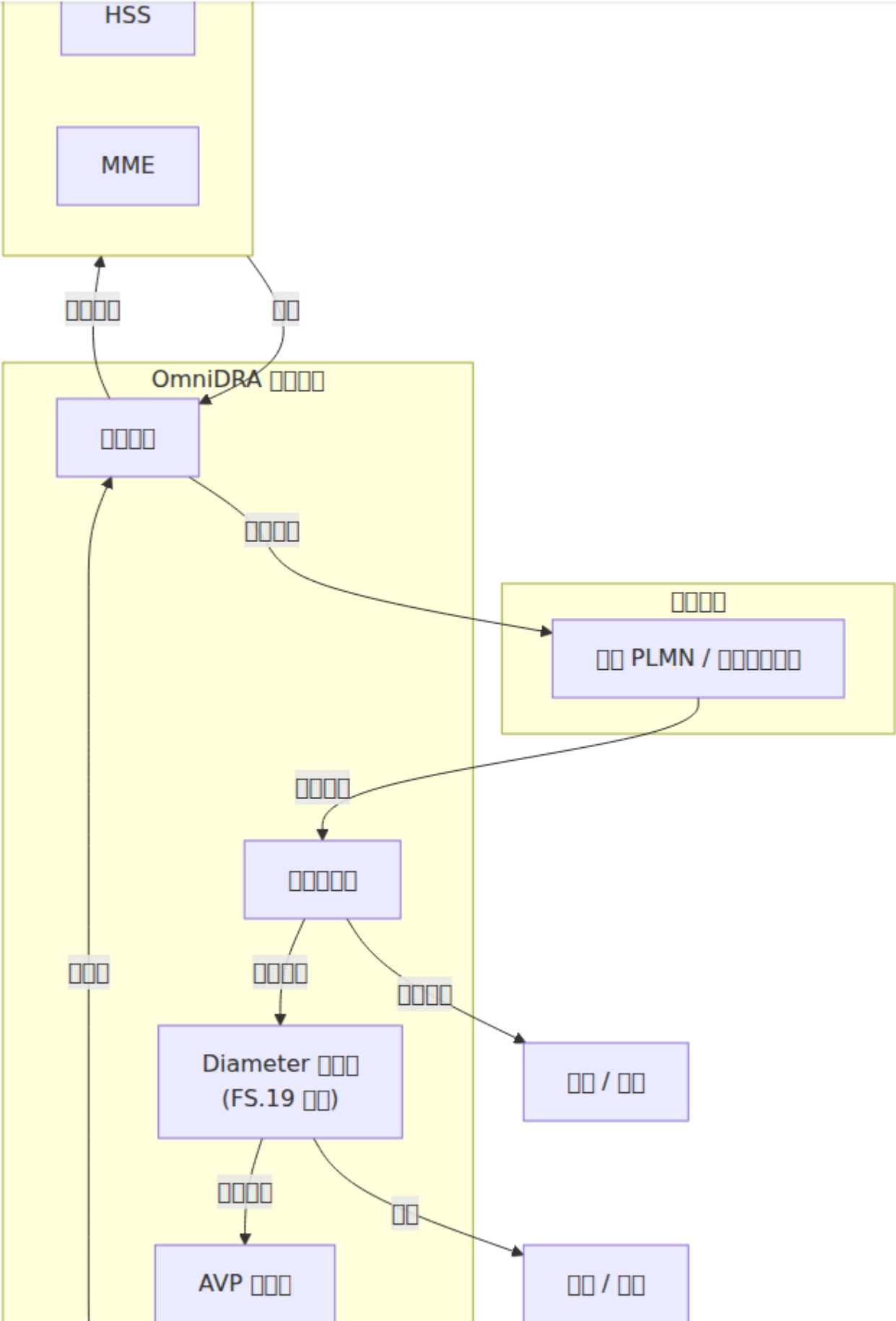
- RFC 4960 - 传输控制协议 (SCTP)
- RFC 6733 - Diameter 协议
- 其他

Diameter

OmniDRA  Diameter  **GSMA FS.19**  Diameter  v10.0 
 **GSMA FS.21**  v12.0  Diameter 





消息	消息	Diameter 消息
<code>{:error, 3002}</code>	向 Diameter 服务器 失败	DIAMETER_UNABLE_TO_DELIVER
<code>{:error, 3004}</code>	向 Diameter 服务器 失败	DIAMETER_TOO_BUSY
<code>{:error, 3007}</code>	向 Diameter 服务器 失败	DIAMETER_APPLICATION_UNSUPPORTED
<code>:drop</code>	丢弃	丢弃
<code>:log_only</code>	只记录	只记录

Diameter 消息

Diameter 消息规范 **GSMA FS.19** 中 **3.3** 中定义了 Diameter 消息规范 **FS.21** 中 **7** 中定义了 Diameter 消息规范



消息

GSMA 消息 FS.19 中 3.3.4 消息 FS.21 中 7.3.1 中

消息规范

配置

配置	FS.19 配置	配置
AVP 配置	3.3.4 4.8.1	配置 AVP 配置 Origin-Host, Origin-Realm 配置 AVP 配置
Session-Id 配置	3.3.4	配置 Session-Id AVP 263 配置 AVP RFC 6733 8.8
配置 Destination-Host	3.3.4	配置 Destination-Host AVP 配置
配置 AVP 配置	3.3.5	配置 AVP 配置 ULR 配置 User-Name 配置 Visited-PLMN-Id 3GPP TS 29.272 7.2.3

```
low_layer: %{\n  enabled: true,\n  action: {:error, 3002},\n  # AVP 配置FS.19 3.3.4\n  # 264 = Origin-Host, 296 = Origin-Realm, 283 = Destination-Realm, 293 = Destination-Host\n  single_instance_avps: [264, 296, 283, 293],\n  # Session-Id 配置 AVPRFC 6733 8.8\n  enforce_session_id_first: true,\n  # 配置 Destination-HostFS.19 3.3.4\n  reject_destination_host_in_answers: true,\n  # 配置 AVP3GPP TS 29.272\n  # 1 = User-Name (IMSI), 1407 = Visited-PLMN-Id, 264 = Origin-Host, 296 = Origin-Realm\n  mandatory_avps: %{\n    316 => [1, 1407, 264, 296],\n    318 => [1, 1407, 264, 296]\n  }\n}
```

S6a/S6d

Sequence	Interface	Direction	Reference
316	Uu/3GPP	MME → HSS	3GPP TS 29.272 §7.2.3
317	Uu/3GPP	HSS → MME	3GPP TS 29.272 §7.2.7
318	Uu/3GPP	MME → HSS	3GPP TS 29.272 §7.2.5
319	Uu/3GPP	HSS → MME	3GPP TS 29.272 §7.2.9
320	Uu/3GPP	HSS → MME	3GPP TS 29.272 §7.2.11
321	Uu/3GPP UE	MME → HSS	3GPP TS 29.272 §7.2.13
323	Uu/3GPP	MME → HSS	3GPP TS 29.272 §7.2.15

Sequence ID FS.19 3.3.5

Sequence ID	Interface	Reference
16777251	S6a/S6d	3GPP TS 29.272
16777252	S13	3GPP TS 29.272
16777312	S6c	3GPP TS 29.338
16777313	SGd	3GPP TS 29.338
16777255	SLg	3GPP TS 29.172
16777267	S9	3GPP TS 29.215

2 - Sequence

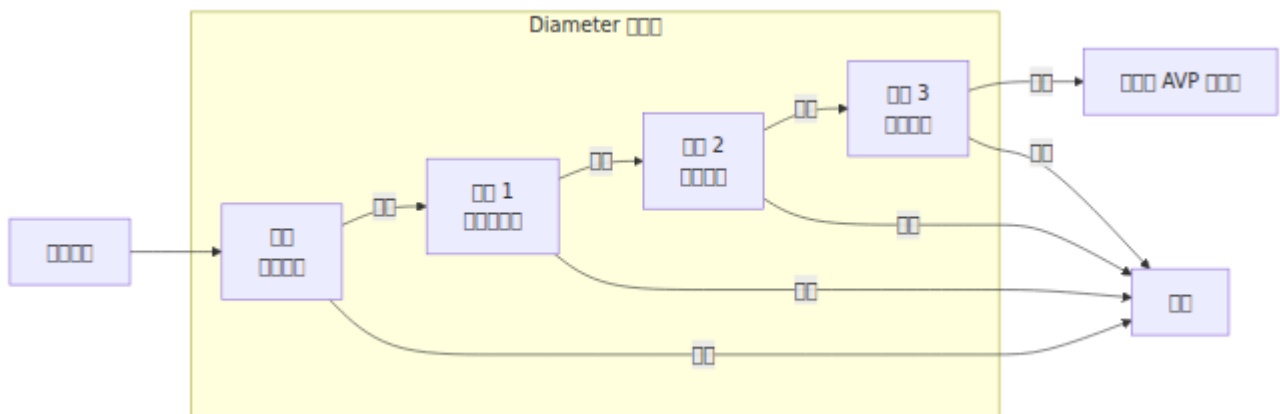
GSMA FS.19 3.3.6 FS.21 7.2.2

Step 2: The first step is to configure the Diameter client to send the IMSI and User-Name AVP. The MSISDN AVP (AVP 701) is also sent in the first step.

The second step is to configure the Diameter client to send the MSISDN AVP.

```
# Configuration
home_imsi_prefixes: ["31338", "31339"],
home_msisdn_prefixes: ["+1313"],

category_2: %{
  enabled: true,
  action: {:error, 3002},
  # S6a FS.19 3.3.6 12
  protected_command_codes: [316, 317, 318, 319, 320, 321, 323]
}
```



Step 3 - Configuration

GSMA FS.19 3.3.7 FS.21 7.2.3

Step 3: The third step is to configure the Diameter client to send the IMSI and User-Name AVP. The MSISDN AVP (AVP 701) is also sent in the first step.

The ETSI standard defines the IMSI as a 24-bit number.

The

項目	FS.19 項目	説明
設定項目	3.3.7.1 項目	ULR の Origin-Realm 項目 IMSI の Origin-Realm
項目/項目 項目	3.3.7.2 項目	

```
category_3: %{
  enabled: true,
  # 項目 log_only 項目FS.19 3.3.7 項目
  action: :log_only,
  # 項目/項目FS.19 3.3.7.2 項目
  max_velocity_kmh: 1200,
  # 項目 IMSI 項目
  min_time_between_updates_seconds: 2
}
```

項目

項目 Diameter 項目
module_diameter_firewall 項目

```
config :dra,
  module_diameter_firewall: %{
    enabled: true,
    home_imsi_prefixes: ["31338", "31339"],
    home_msisdn_prefixes: ["+1313"],
    low_layer: %{ ... }, # 項目
    category_1: %{ ... }, # 項目 1
    category_2: %{ ... }, # 項目 2
    category_3: %{ ... } # 項目 3
  }
```

配置

配置	是否	是否	配置	配置
<code>enabled</code>	是否	是否	<code>false</code>	是否启用 Diameter 配置 <code>false</code>
<code>children</code>	是否	是否	<code>[]</code>	DRA 配置项 <code>[DRA.Router.Modules.DiameterF</code> 配置 3 个配置项
<code>home_imsi_prefixes</code>	是否	是否	<code>[]</code>	配置 IMSI 配置 2 个配置
<code>home_msisdn_prefixes</code>	是否	是否	<code>[]</code>	配置 MSISDN 配置 2 个配置

配置

配置	是否必填	是否必选	默认值	说明
<code>enabled</code>	是	否	<code>true</code>	是否启用该策略
<code>action</code>	是	否	<code>{:error, 3002}</code>	策略执行失败时的错误码
<code>single_instance_avps</code>	否	否	<code>[264, 296, 283, 293]</code>	该策略支持的 AVP 列表 Origin-Host (264) Origin-Realm (296) Destination-Realm (283) Destination-Host (293)
<code>enforce_session_id_first</code>	是	否	<code>true</code>	是否强制 Session-Id 在 AVP 列表第一位
<code>reject_destination_host_in_answers</code>	否	否	<code>true</code>	是否拒绝 Destination-Host AVP 在答案中
<code>mandatory_avps</code>	否	否	<code>%{}</code>	强制 AVP 列表 格式： <code>%{316 => [1, 1407, 264, 296]}</code>

URI	URI Scheme	URI Host	URI Path	URI Query
				URI ULR URI URI User- NameURI Visited- PLMN-IdURI Origin-Host URI Origin- RealmURI

URI 1 URI

URI	URI Scheme	URI Host	URI Path	URI Query
enabled	URI	URI	true	URI 1 URI
action	URI	URI	{:error, 3007}	URI 1 URI DIAMETER_APPLICATION_UNSUPPORTEDURI
whitelists	URI	URI	-	URI :allURI ID / URI URI :all URI

Table 2

Item	Item	Item	Item
enabled	Item	Item	Item 2 Item
action	Item	Item	Item Item
protected_command_codes	Item	Item	Item Item S6a Item Item

3

enabled			false	LocationTracker
action			:log_only	:log_only
max_velocity_kmh			1200	
min_time_between_updates_seconds			2	IMSI

配置

配置	说明
<code>[:diameter, :firewall, :low_layer, :block]</code>	低层防火墙配置
<code>[:diameter, :firewall, :category_1, :block]</code>	第一类防火墙配置
<code>[:diameter, :firewall, :category_2, :block]</code>	第二类防火墙配置
<code>[:diameter, :firewall, :category_3, :block]</code>	第三类防火墙配置
<code>[:diameter, :firewall, :pass, :count]</code>	防火墙通过次数

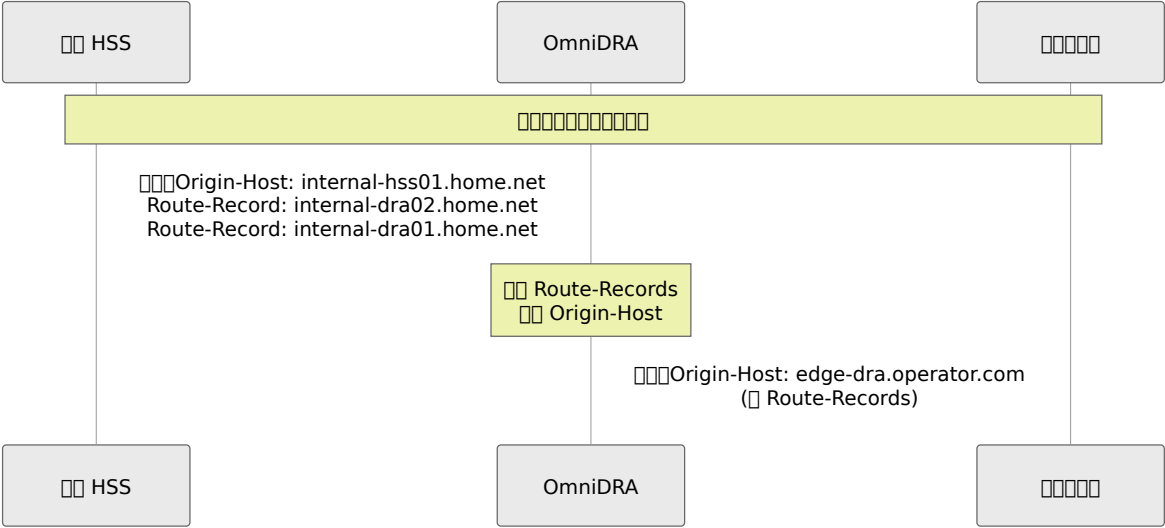
配置项包括：`origin_host`、`application_id`、`command_code`、`application_name`、`command_name`、`reason`

配置

GSMA 配置 FS.19 2.4 3.4 FS.21 3.6

配置项包括：`Diameter`、`Route-Record`、`AVP`、`Origin-Host/Origin-Realm`

配置项包括：`DRA`



Table

Item	FS.19 Item	Item
Route-Record Item	Item 2.4	Route-Record AVP Item 282
Origin-Host Item	Item 3.4	Origin-Host AVP Item DRA Item
Origin-Realm Item	Item 3.4	Origin-Realm AVP Item
Item	-	Item

```
module_topology_hiding: %{
  enabled: true,
  # 删除 Route-Record AVPFS.19 2.4
  strip_route_records: true,
  # 删除 Origin-Host 属性FS.19 3.4
  rewrite_origin_host: %{enabled: true, replacement: :self},
  # 删除 Origin-Realm
  rewrite_origin_realm: %{enabled: false, replacement: :self},
  # 删除所有外部对等体
  external_peers: :all
}
```

表

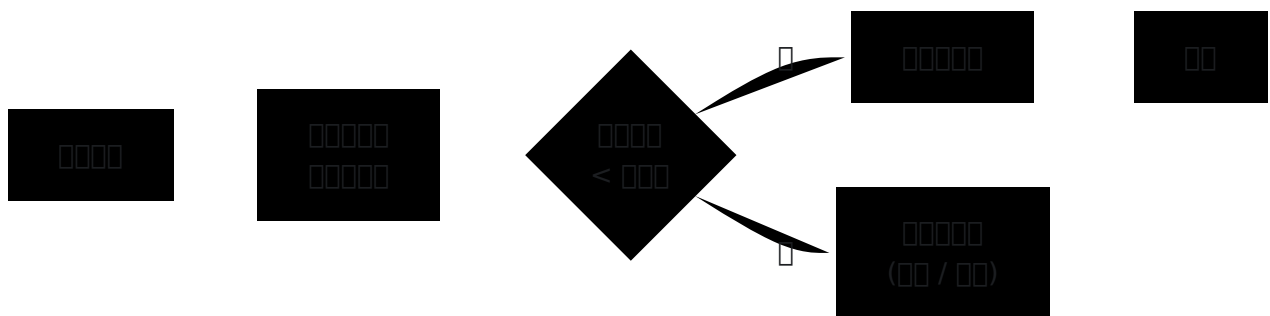
属性	默认值	是否强制	替换值	说明
enabled	是	否	false	是否启用
strip_route_records	是	否	true	是否删除 DRA 属性 Route-Record AVP 282
rewrite_origin_host	是	否	自身	是否删除 Origin-Host 属性
rewrite_origin_realm	是	否	自身	是否删除 Origin-Realm 属性
external_peers	:all 所有	否	:all	删除所有外部对等体 :all 删除所有外部对等体

删除 rewrite_origin_host 和 rewrite_origin_realm 属性


```

module_rate_limiter: %{
  enabled: true,
  # 配置FS.19 3.4
  default_max_requests_per_second: 1000,
  default_action: {:error, 3004},
  # 配置
  peer_limits: %{
    "high-volume-partner.roaming.com" => %
    {max_requests_per_second: 5000, action: {:error, 3004}},
    "restricted-peer.roaming.com" => %{max_requests_per_second:
    100, action: :drop}
  }
}

```



enabled			false	
children			[]	DRA [DRA.Router.Modu]
default_max_requests_per_second			1000	
default_action			{:error, 3004}	DIAMETER_TOC
peer_limits			%{}	

max_requests_per_second				
action				

--	--	--	--

入参	出参
[:diameter, :rate_limiter, :throttled]	返回当前速率限制器 <code>current_rate</code> 和 <code>limit</code>
[:diameter, :rate_limiter, :allowed]	返回是否允许

AVP ☐ ☐ ☐

GSMA □□□ FS.19 □ 3.3.4 □□4.8.1□4.8.2□FS.21 □ 3.6 □

AVP 000000000000 Diameter AVP00000 FS.21 3.6 000000000000 FS.19 00000
000000000000

11

項目	GSMA 項目	備考
AVP 項目	FS.21 3.6 項目	AVP項目は、AVP項目は、AVP項目
AVP 項目	FS.21 3.6 項目	AVP 項目は、AVP 項目は、AVP 項目

```
module_avp_sanitizer: %{\n  enabled: true,\n  action: {:error, 3002},\n  # 0 = IETF, 10415 = 3GPP, 13019 = ETSI, 5535 = 3GPP2\n  allowed_vendor_ids: [0, 10415, 13019, 5535],\n  strip_unknown_vendor_avps: true,\n  # 0 = IETF, 10415 = 3GPP, 13019 = ETSI, 5535 = 3GPP2\n  max_avp_nesting_depth: 10\n}
```

Table

Field	Default	Value	Description
enabled	false	false	Whether to enable AVP sanitization
action	{:error, 3002}	{:error, 3002}	Action to take when an AVP is not in the allowed_vendor_ids list
allowed_vendor_ids	[0, 10415, 13019, 5535]	[0, 10415, 13019, 5535]	List of vendor IDs allowed for AVP processing
strip_unknown_vendor_avps	true	true	Whether to strip unknown vendor AVPs from the allowed_vendor_ids list
max_avp_nesting_depth	10	10	Maximum nesting depth for AVP processing

Vendor ID

Vendor ID	Vendor	Notes
0	IETF	RFC 4074 Diameter AVP
10415	3GPP	3GPP TS 29.271 AVP S6a Gx Rx
13019	ETSI	ETSI TS 102 229 AVP
5535	3GPP2	3GPP2 IS-832 AVP CDMA

Example

Log Entry	Field
<code>[{:diameter, :avp_sanitizer, :unknown_vendor, :stripped}]</code>	Unknown vendor AVP
<code>[{:diameter, :avp_sanitizer, :nesting_depth, :violation}]</code>	AVP nesting violation
<code>[{:diameter, :avp_sanitizer, :pass, :count}]</code>	AVP sanitizer pass count

Example

Example

Example log entry:

- Example - Unknown vendor AVP
- AVP** type - Unknown vendor AVP
- Diameter** message **1** - Diameter message ID / Diameter message
- Diameter** message - Diameter message
- Diameter** message **2** - Diameter IMSI/MSISDN

6. 配置 - 配置 Route-Record 配置 Origin-Host

7. **Diameter** 配置 3 - 配置 :log_only 配置

配置

配置 FS.19 3.4 配置 配置 配置

配置	配置	配置
配置 DoS	配置	Diameter 配置
配置	配置 1	AVP 配置
配置	配置 2	配置 1配置
配置	配置 3	配置 2配置
配置	配置	-
AVP 配置	AVP 配置	配置
配置AVP 配置	配置	AVP 配置

GSMA 000000

00 / 00	FS.19 v10.0	FS.21 v12.0
000000	0 3.3.4 0	0 7.3.1 0
00 1 00	0 3.3.5 0000 B.3.3	0 7.2.1 0
00 2 00	0 3.3.6 0000 B.3.4	0 7.2.2 000 16 0
00 3 0???	0 3.3.7 0000 B.3.5	0 7.2.3 0
0000	0 2.4 000 3.4 0	0 3.6 0
0000	0 3.4 0	-
AVP 00	0 4.8.1 004.8.2 0	0 3.6 0
0000	0 3.4 0	0 3.15 0
0000000000	00 A	0 7 0

DRACON (RFC 6733)
AVP DRACON

DRA[Destination-Host]

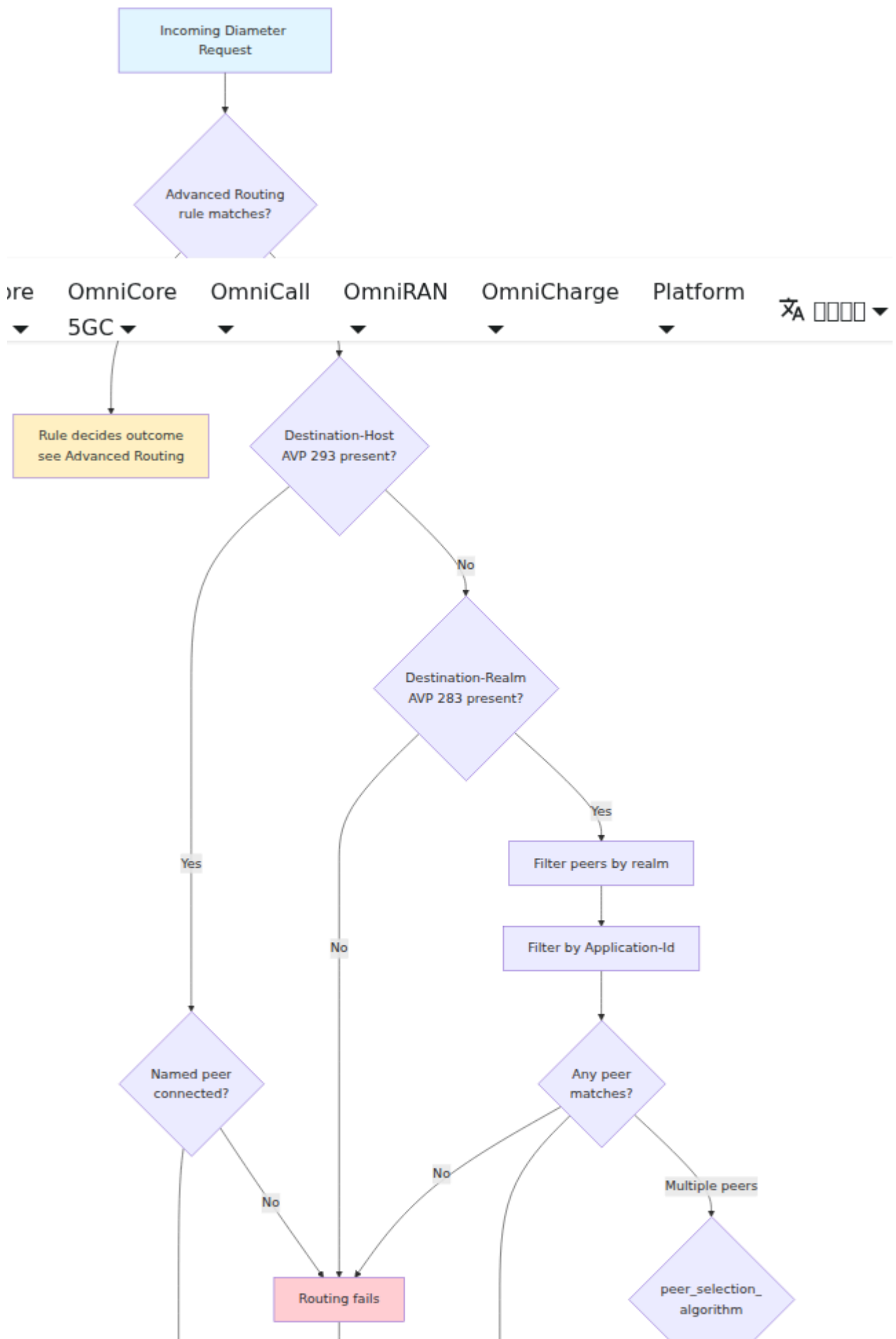
`Rx[RxPCRF]:drop`

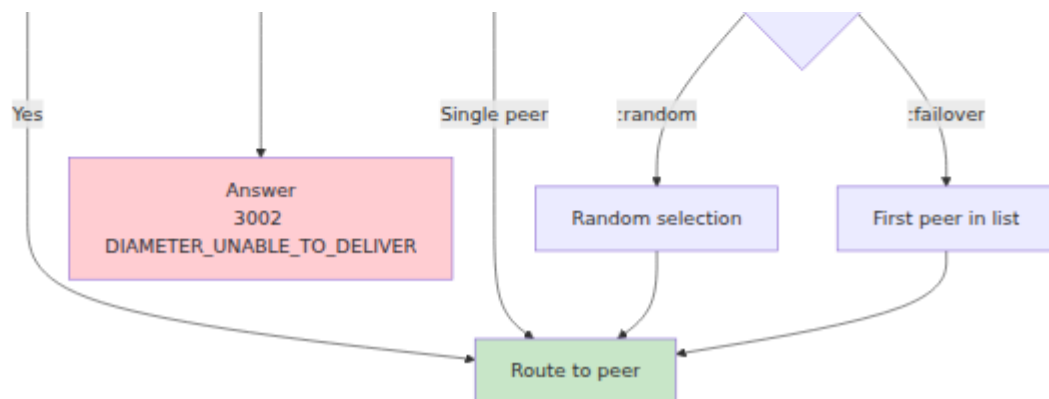
`Destination-Host`

Destination-Host AVP IMS!

1. **Destination-Host AVP (293)** - **Destination-Host**
AVP DRA
2. **Destination-Realm AVP (283)** + **Application-Id** - **Destination-Host** DRA **Application-Id**
 DRA
3. / - **Destination-Host** DRA DRA
 3002 DIAMETER_UNABLE_TO_DELIVER







1. DRA가 Diameter_UNable_to_Deliver 메시지를 수신하면
 2. Destination-Host

3.

1. Application-Id에 peer_selection_algorithm이 설정되어 있는지 확인
 2.

```

config :dra, :service, %{
  # ... DRA identity and network settings ...
  peer_selection_algorithm: :random
}
    
```

1.	2.	3.	4.	5.
peer_selection_algorithm	Atom	1.	:random	1. peer_selection_algorithm이 설정되어 있는지 확인 2. :random이 설정되어 있는지 확인 3. :failover이 설정되어 있는지 확인 4.

- :random -
- :failover -

1.

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

Destination-Host
Application-Id AVP DRA

Result-Code 3002 DIAMETER_UNABLE_TO_DELIVER RFC 6733

\$7.1.5 DRA

11

--	--	--	--

番号	名前	説明	参照
2001	DIAMETER_SUCCESS	Diameterメッセージの送信が成功した。	RFC 6733 §7.1.1
3002	DIAMETER_UNABLE_TO_DELIVER	Diameterメッセージの送信に失敗した。 失敗の原因は不明である。	RFC 6733 §7.1.3

AVP

	AVP			
1	User-Name	UTF8String	(IMSI in 3GPP)	RFC 6733 §8.14
264	Origin-Host	DiameterIdentity		RFC 6733 §6.3
268	Result-Code	Unsigned32		RFC 6733 §7.1
283	Destination-Realm	DiameterIdentity		RFC 6733 §6.6
293	Destination-Host	DiameterIdentity	()	RFC 6733 §6.5
296	Origin-Realm	DiameterIdentity		RFC 6733 §6.4
297	Experimental-Result	Grouped		RFC 6733 §7.6

AVP

번호	코드	설명	참고
257	CER/CEA	인증서/인증서	RFC 6733 §5.3
258	RAR/RAA	인증서/인증서	RFC 6733 §8.3
274	ASR/ASA	인증서/인증서	RFC 6733 §8.5
275	STR/STA	인증서/인증서	RFC 6733 §8.4
280	DWR/DWA	인증서/인증서	RFC 6733 §5.5
282	DPR/DPA	인증서/인증서	RFC 6733 §5.4
316	ULR/ULA	인증서/인증서 (S6a)	3GPP TS 29.272
317	CLR/CLA	인증서/인증서 (S6a)	3GPP TS 29.272
318	AIR/AIA	인증서/인증서 (S6a)	3GPP TS 29.272
321	PUR/PUA	인증서/인증서 (S6a)	3GPP TS 29.272

3GPP ID

Application-IdsApplication-Id

Application-Id	接口	接口描述	参考
16777251	S6a/S6d	MME/SGSN ↔ HSS 鉴权/认证	3GPP TS 29.272
16777252	S13/S13'	MME ↔ EIR 鉴权/认证	3GPP TS 29.272
16777238	Gx	PCEF ↔ PCRF 策略控制	3GPP TS 29.212
16777267	S9	PCRF ↔ PCRF 策略控制	3GPP TS 29.215
16777272	S6b	PDN-GW ↔ 3GPP AAA 鉴权/认证 3GPP	3GPP TS 29.273
16777216	Cx	I-CSCF/S-CSCF ↔ HSS IMS 鉴权/认证	3GPP TS 29.229
16777217	Sh	AS ↔ HSS IMS 鉴权/认证	3GPP TS 29.329
16777255	SLg	MME/SGSN ↔ GMLC 鉴权/认证	3GPP TS 29.172
16777291	SLh	GMLC ↔ HSS 鉴权/认证	3GPP TS 29.173
16777302	Sy	PCRF ↔ OCS 策略控制	3GPP TS 29.219
16777310	S6m	MTC-IWF ↔ HSS/HLR 鉴权/认证 M2M	3GPP TS 29.336

Application-Id	API	API Description	Reference
16777312	S6c	SMS-SC/IP-SM-GW ↔ HSS SMS API	3GPP TS 29.338
16777345	S6t	SCEF ↔ HSS API	3GPP TS 29.336
16777236	Rx	AF ↔ PCRF API	3GPP TS 29.214

API (Gy/Ro) 3GPP TS 32.299

API

- API - API
- API - API Destination-Host API

□□□□□□ (SLF)

[illegible]

SLg/SLh GMLC MME MME

3GPP TS 29.172 3GPP TS 29.173 SLF 3GPP TS 29.272

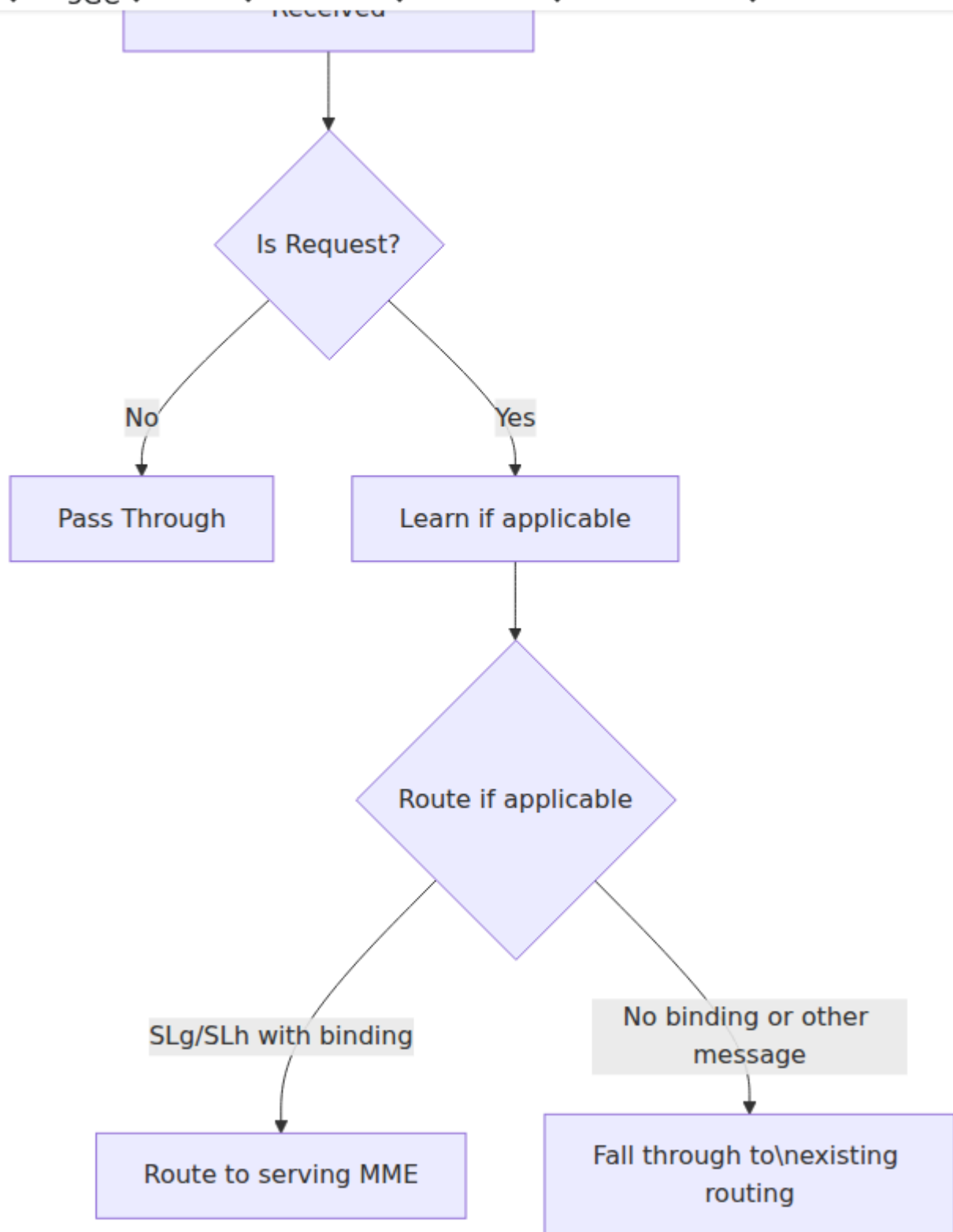
1111

□ □ □ □ □ □ □ □ □ □ □ □ □ □ □

00000000 DRA 00000000000000000000 S6a Update-Location-Request
0000000000000000000000000000

SLg Provide-Location-Request

□ □



□□□□□□



Subscriber unknown

NoBinding

Registration message
observed

Deregistration message
observed

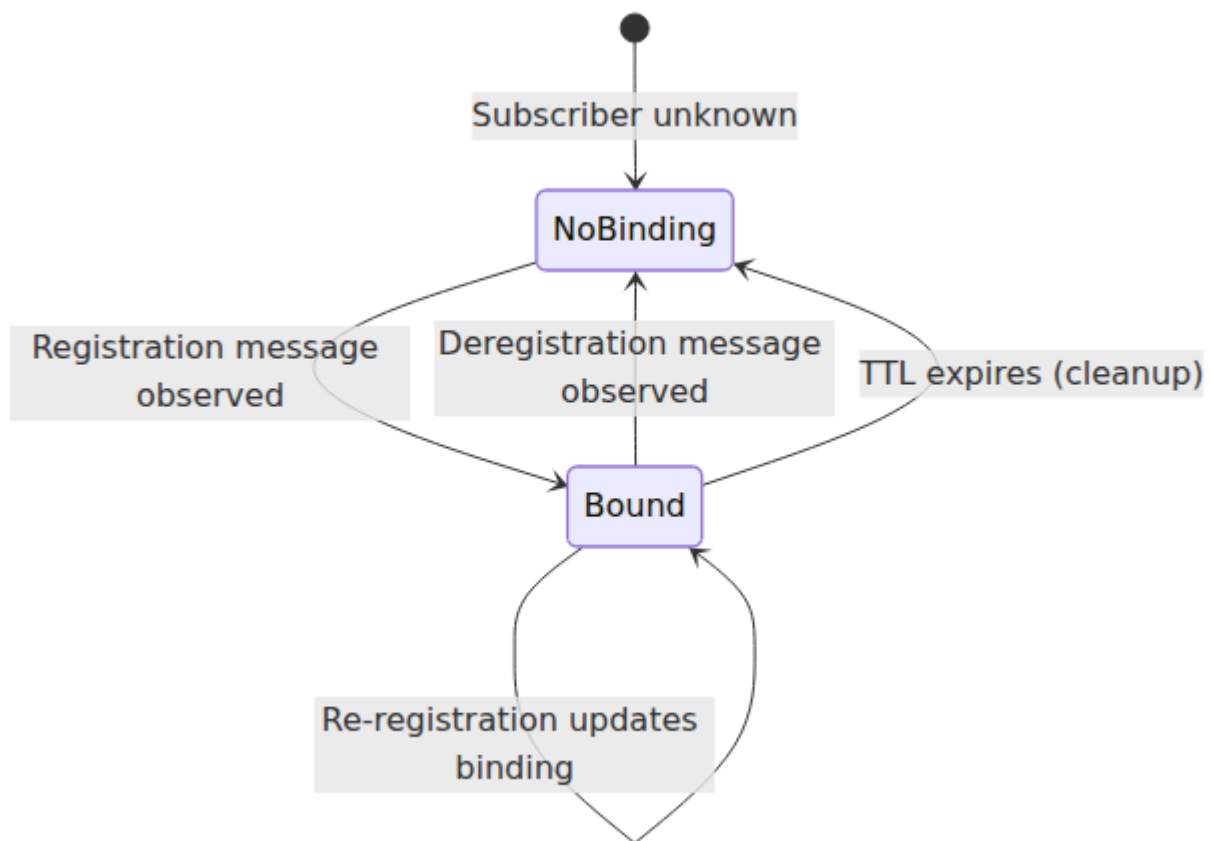
TTL expires (cleanup)

Bound

Re-registration updates
binding

□□□□□□□□

□□□□ Diameter □□□□□□□□□□□□□□



□□	□□	□□□	□□□□□	IMSI □□
S6a	Update-Location-Request (ULR)	316	serving_mme	User-Name AVP (1)
Gx	Credit-Control-Request Initial (CCR-I)	272 (CC-Request-Type=1)	serving_pgw	Subscription-Id AVP (443)
Cx	Server-Assignment-Request (SAR)	301 (Server-Assignment-Type=1)	serving_cscf	Public-Identity AVP (601)

Table 1: Diameter Messages

Interface	Message Name	Message ID	Originator
S6a	Cancel-Location-Request (CLR)	317	serving_mme
S6a	Purge-UE-Request (PUR)	321	serving_mme
Gx	Credit-Control-Request Termination (CCR-T)	272 (CC-Request-Type=3)	serving_pgw
Cx	Server-Assignment-Request (SAR)	301 (Server-Assignment-Type=4)	serving_cscf

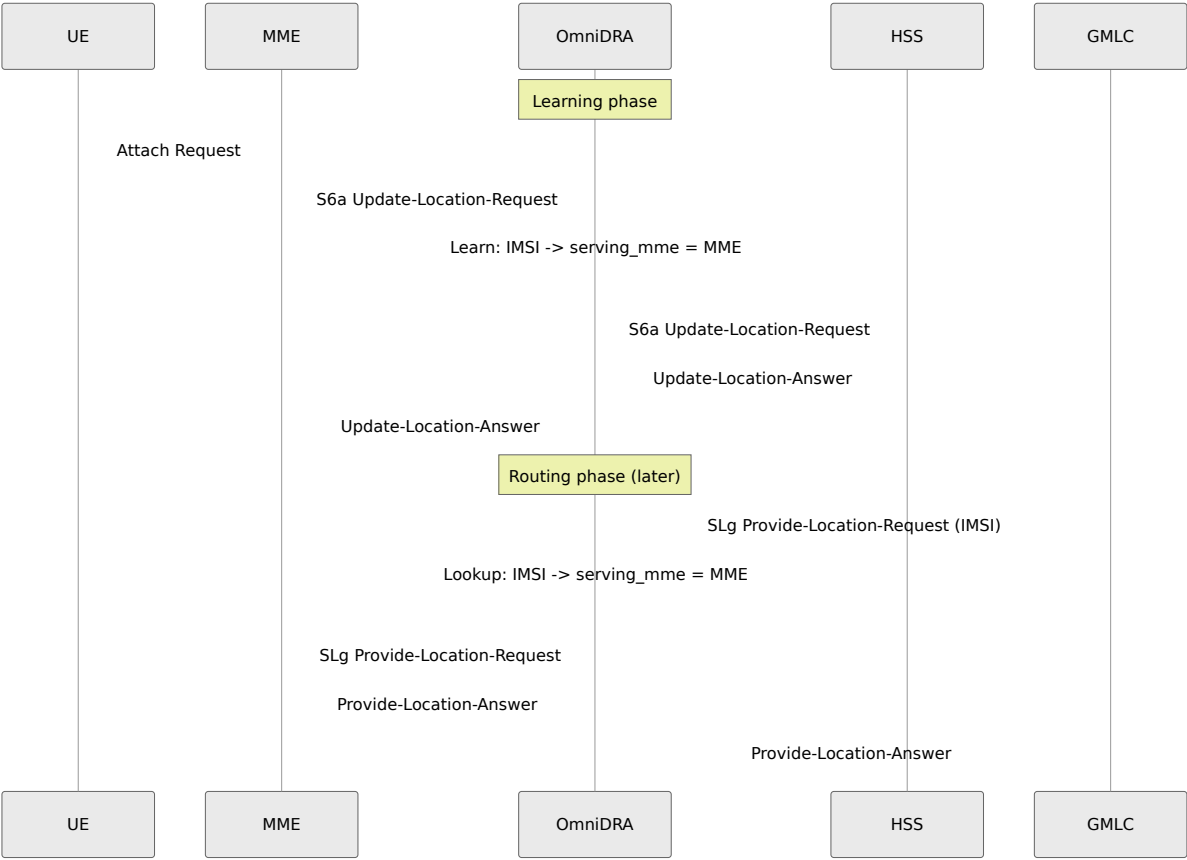
Table 2: Diameter Messages (Continued)

Table 3: Diameter Messages

Interface	Message Name	Message ID	Originator
SLg	Provide-Location-Request (PLR)	8388620	serving_mme
SLh	LCS-Routing-Info-Request (RIR)	8388622	serving_mme

DRA configuration parameters: `diameter.request_timeout` (default: 30s)
Peer Connectivity Configuration: `Peer Connectivity Configuration`

□□□□□□□□□□



□□□□

□□□□□□□□□□□□□□□□□□ SLF □□□□□□□□□□□□□□□□□□ SLF □□□□□□□□

Incoming Request

Subscriber Lookup

Advanced Routing

Peer Selection / Relay

□□

□ config/runtime.exs □□ module_subscriber_lookup □□□□□□□

```
module_subscriber_lookup: %{\n  # \n\n  enabled: false,\n\n  # \n\n  binding_ttl_seconds: 86400\n}
```

Table

Field	Type	Value	Description
enabled	bool	false	Whether the subscriber lookup is enabled. Default is false.
children	list	[]	List of DRA children. Default is [].
binding_ttl_seconds	int	86400	Binding TTL in seconds. Default is 86400 (24 hours).

Example

Example

Example configuration for 24 hours binding TTL

```
module_subscriber_lookup: %{\n  enabled: true,\n  binding_ttl_seconds: 86400\n}
```

이러한 구성을 통해 MME/PGW/CSCF는 24시간 DRA를 SLg와 SLh를 통해
MME를 통해 연결할 수 있습니다.

이러한 구성 (LCS)은 GMLC를 통해 MME를 연결할 수 있습니다.

이러한

이러한 구성을 통해 MME를 연결할 수 있습니다.

```
module_subscriber_lookup: %{  
  enabled: true,  
  binding_ttl_seconds: 3600  
}
```

이러한 구성을 통해 1개의 MMEs를 연결할 수 있습니다. 30초
이후

이러한 구성을 통해 MME를 연결할 수 있습니다.

이러한

이러한

이러한 구성을 통해 diameter.subscriber_lookup.binding.update

이러한 구성을 통해

이러한 구성을 통해 DRA를 연결할 수 있습니다.

이러한

- `imsi` - IMSI
- `binding_type` - `serving_mme`, `serving_pgw`, `serving_cscf`
- `serving_host` - Origin-Host

이러한

이러한 구성을 통해 diameter.subscriber_lookup.binding.delete

이러한 구성을 통해

DR: DRA 000000000000000000000000

DR:

- `imsi` - 0000 IMSI
- `binding_type` - 00000000

0000

DR: `diameter.subscriber_lookup.route.count`

DR: 000

DR: DRA 000000000000 SLg/SLh 000000000000

DR:

- `imsi` - 0000 IMSI
- `binding_type` - DRA 0000000000000000 `serving_mme`
- `serving_host` - DRA 0000000000 Origin-Host

0000

0000000000 **MME**

DR: DRA 000 SLg Provide-Location-Requests 0 SLh LCS-Routing-Info-Requests 00
0000 MME000000000000

00000:

- 00000000 DRA 000000000000
- 000000TTL 000
- 00 MME 00000000 DRA
- `enabled` 000 `false`

0000:

1. 000000000000
2. 000000 S6a ULR 000000 DRA000000000000000000
3. 00 `binding_ttl_seconds` 00000000000000000000

4. MME

: S6a/Gx/Cx DRA

:

-
-
- AVP IMSI

:

1. enabled: true
2. DRA
3. DRA SubscriberLookup: Learned

: DRA MME

:

- Cancel-Location-Request (CLR) Purge-UE-Request (PUR) DRA
- binding_ttl_seconds

:

1. binding_ttl_seconds
2. S6a DRA

□□

□□□□ ID

ID	□□	□□	□□
16777251	S6a/S6d	MME/SGSN □ HSS □□□□□□□□	3GPP TS 29.272
16777238	Gx	PCEF □ PCRF □□□□□□□□	3GPP TS 29.212
16777216	Cx	I-CSCF/S-CSCF □ HSS □ IMS □□	3GPP TS 29.229
16777255	SLg	GMLC □ MME □□□□□	3GPP TS 29.172
16777291	SLh	GMLC □ HSS/DRA □ LCS □□□□	3GPP TS 29.173

□□□

□□	□□	□□	□□
272	Credit-Control-Request/Answer (CCR/CCA)	Gx	□□□□□□□□□□
301	Server-Assignment-Request/Answer (SAR/SAA)	Cx	IMS □□□□□
316	Update-Location-Request/Answer (ULR/ULA)	S6a	□ MME □□□□□□ □□
317	Cancel-Location-Request/Answer (CLR/CLA)	S6a	HSS □□□□□□□
321	Purge-UE-Request/Answer (PUR/PUA)	S6a	MME □□□ UE □ □
8388620	Provide-Location-Request/Answer (PLR/PLA)	SLg	□□□ MME □□□□□ □□□
8388622	LCS-Routing-Info-Request/Answer (RIR/RIA)	SLh	LCS □□□□□□

网络

网络	网络	网络	网络
serving_mme	S6a ULR	SLg PLR, SLh RIR	网络 MME
serving_pgw	Gx CCR-I	-	网络 PGW网络
serving_cscf	Cx SAR (Registration)	-	网络 IMS 网络 S-CSCF网络



OmniDRA Metrics



目標

前提知識

ネットワークの基本的な構成と動作について理解していること。

学習目標

- ネットワークの基本的な構成と動作について理解していること。
- ネットワーク AVP と Diameter プロトコル AVP の関係
- ネットワークの基本的な構成と動作について理解していること。
- ネットワークの基本的な構成と動作について理解していること **match** プロトコル
- ネットワークの基本的な構成と動作について理解していること

学習内容

1. ネットワークの基本的な構成と動作について理解していること
2. ネットワーク AVP と Diameter プロトコル **Advanced Routing** ネットワーク AVP の関係
3. ネットワークの基本的な構成と動作について理解していること
4. ネットワークの基本的な構成と動作について理解していること **match** プロトコル
5. ネットワーク **Advanced Routing** プロトコルについて理解していること

まとめ

ネットワークの基本的な構成と動作について理解していること。DRA プロトコル

学習目標

- ネットワークの基本的な構成と動作について理解していること。
- ネットワークの基本的な構成と動作について理解していること
- ネットワーク **???** プロトコルについて理解していること
- ネットワークの基本的な構成と動作について理解していること **Standard Routing** プロトコル

学習内容

1. 配置 Advanced Routing
2. 配置 `diameter_peer_status` 为 1
3. 配置
4. 配置 Standard Routing

配置

配置 AVP 的 / 配置

配置

- 配置 AVP
- `edit` 配置 AVP `edit` 配置 AVP
- 配置
- 配置 `:request` `:answer`
- `edit` 配置

配置

1. 配置 AVP Advanced Routing 配置 AVP
2. 配置 `edit` AVP `add`
3. 配置
4. 配置 `:request` `:answer`
5. `edit` 配置

配置

配置

配置 `diameter_peer_status{origin_host="..."}` 为 0

配置

- 配置 `listen_port` 为 3868


```

config :dra, :gateway,
  # 0 = 0 CER/CEA
  peer_settle_time_ms: 2_000,

  # RFC 3539
  # watchdog_timer_ms: 30_000

```

項目	単位	値	説明
peer_settle_time_ms	ms	0	DRA 0 CER/CEA DRA 0
watchdog_timer_ms	ms	30000	RFC 3539 Tw DWR/DWA

3. `diameter_peer_status` 0
4. `peer_settle_time_ms` 0

DRA RFC 3539 DRA

0

DRA `diameter_peer_unauthorized_connection_count_total`
`origin_host` / `peer_ip`

0

- `peers` `allow_undefined_peers_to_connect` `false`

- 配置 Origin-Host 参数 host
- 配置 peers 参数

配置

1. 配置 peers 参数 host realm ip port transport
2. 配置 Origin-Host 参数 host 配置 DRA 参数
3. 配置 allow_undefined_peers_to_connect: false 配置
4. 配置 log_unauthorized_peer_connection_attempts: true 配置 DRA 配置
5. 配置 diameter_peer_unauthorized_connection_count_total 配置 peer_ip 配置 origin_host 配置

配置	配置	配置	配置
allow_undefined_peers_to_connect	配置	配置	配置 false
log_unauthorized_peer_connection_attempts	配置	配置	配置 true

SCTP 配置

このSCTP 設定は、ネットワークの信頼性を向上させるために、TCP 設定と併用して

設定します。

- ネットワークに SCTP を有効
- ネットワークに TCP と SCTP を同時に有効
- ネットワークに SCTP を有効にするための設定
- IP 設定に `additional_ips` を追加
- ネットワークに DRA を有効にする
- SCTP の設定を確認する

手順

1. SCTP を有効にするLinux 上で `lksctp-tools` をインストール
2. ネットワークに SCTP を有効にする TCP と同時に有効
3. ネットワークに IP を有効にする DRA を有効にする
4. SCTP の設定に IP を追加 `additional_ips` を追加 TCP の設定に DRA を追加 `ip` を追加 TCP を有効にする
5. ネットワークに SCTP を有効にする
6. ネットワークに SCTP を有効にする DRA を有効にする
7. SCTP Multihoming を有効にする

エラー

エラー / 3002 エラー

ネットワークに DRA を有効にする際に、3002 `DIAMETER_UNABLE_TO_DELIVER` エラーが発生する場合があります。DRA を有効にする前に、ネットワークに SCTP を有効にする必要があります。

エラー

- ネットワークに Destination-Host と Destination-Realm を有効
- ネットワークに SCTP を有効にする

- 設定するパラメータ
- 設定するパラメータ
- 設定するパラメータ 3003 DIAMETER_REALM_NOT_SERVED 3002

設定

1. 設定する Destination-Host と Destination-Realm の設定
diameter_peer_status の値 1
2. 設定する peer_settle_time_ms の設定
Peer Flapping
3. 設定する CER/CEA の設定 DRA の設定
4. 設定する DRA の設定 Advanced Routing
5. 3002 3003 3002 の DRA の設定 3003 の設定
3003 の設定
6. 3002 の設定

```
rate(diameter_peer_message_result_code_count_total{result_code="3
[5m])
```

設定

設定 DRA の設定

diameter_peer_unanswered_request_count_total の設定
diameter_peer_last_response_delay の設定

設定

- 設定する request_timeout の値 5000 の設定
- 設定する
- 設定する SCTP の設定
- 設定する

設定

1. 配置参数

```
topk(5, sum by (routed_to) (
  rate(diameter_peer_unanswered_request_count_total[5m])
))
```

2. 配置参数 `diameter_peer_last_response_delay` 配置参数

3. 配置参数 `diameter_peer_last_response_delay` DRA 配置

4. 配置 SCTP 配置参数 `SCTP Multihoming`

5. 配置参数 `request_timeout` 配置参数 TTL
配置 `request_timeout` + 5 配置

配置	配置	配置	配置	配置
<code>request_timeout</code>	配置	配置	<code>5000</code>	配置 DRA 配置/配置参数

配置 Gx / Gy 配置 UNKNOWN_SESSION_ID

配置参数 Gx 配置参数 Gy 配置参数 `5002` `DIAMETER_UNKNOWN_SESSION_ID`
配置参数 CCR-U 配置参数 CCR-T 配置参数 CCR-I 配置

配置参数

- DRA 配置参数
- 配置参数
- DRA 配置参数
- 配置参数 Session-Id 配置参数

配置参数

1. 配置参数 Session-Id 配置参数 Gx 配置 Gy 配置参数
配置 CCR-I 配置参数 `Destination-Host` 配置参数 Session-Id 配置参数
配置参数 `Standard Routing` 配置 `Advanced Routing`

2. 設定 `peer_selection_algorithm` を `:random` に変更すると、ランダムにピアを選択します。
設定 `:failover` は、ピアが利用不可になると、`Destination-Host` のピアを選択します。
3. 設定 `5002` は、ピアが利用不可になると、ランダムにピアを選択します。
4. 設定 `peer_settle_time_ms` は、ピアが DRA に参加してから `Peer Flapping` までの時間を設定します。
5. 設定 `5002` は、ピアが利用不可になると、ランダムにピアを選択します。

```
rate(diameter_peer_message_result_code_count_total{result_code="5[5m]})
```

Table 1

Code	Message	Direction	Reference
2001	DIAMETER_SUCCESS	DRA to NAS	RFC 6733 §7.1.1
3002	DIAMETER_UNABLE_TO_DELIVER	DRA to NAS NAS to DRA	RFC 6733 §7.1.3
3003	DIAMETER_REALM_NOT_SERVED	NAS to DRA	RFC 6733 §7.1.3
3004	DIAMETER_TOO_BUSY	NAS to DRA	RFC 6733 §7.1.3
5002	DIAMETER_UNKNOWN_SESSION_ID	NAS to DRA Session-Id DRA to NAS	RFC 6733 §7.1.5
5012	DIAMETER_UNABLE_TO_COMPLY	DRA to NAS	RFC 6733 §7.1.5

Table 2

- Metrics - Prometheus
- Standard Routing - RFC 6733
- Advanced Routing -
- SCTP Multihoming - SCTP

