Mergulhe fundo na comunicação baseada em SCP Model-D

Contents

Introdução

Este documento descreve o aprofundamento da abordagem de comunicação SCP Model-D entre o Cisco AMF/SMF e a NF de terceiros.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Funcionalidade da função de gerenciamento de acesso e mobilidade (AMF)
- Funcionalidade da Função de Gerenciamento de Sessão (SMF)
- Funcionalidade do Proxy de Comunicação de Serviço (SCP)

Componentes Utilizados

Este documento não se restringe a versões de software e hardware específicas.

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Informações de Apoio

Operadores em todo o mundo podem escolher entre vários modelos de comunicação usando

SCP para a descoberta da Função de Rede (NF) e comunicações subsequentes de NF para NF. Este tópico aborda os conceitos sobre vários modelos de comunicação e as alterações de fluxo/configuração de chamadas necessárias na Infraestrutura de Microsserviços de Assinante (SMI - Subscriber Microservices Infrastructure), AMF/SMF para ter comunicação baseada em SCP Model-D.

Visão geral da arquitetura e da solução

Na arquitetura baseada em serviço (SBA), o SCP atua como um intermediário, facilitando a comunicação indireta entre NFs ao lidar com roteamento, balanceamento de carga e descoberta de serviços, simplificando, em última análise, a arquitetura baseada em serviço.

O 3GPP 23.501 Anexo E detalha os quatro modelos de comunicação entre NF em uma implantação 5GC.

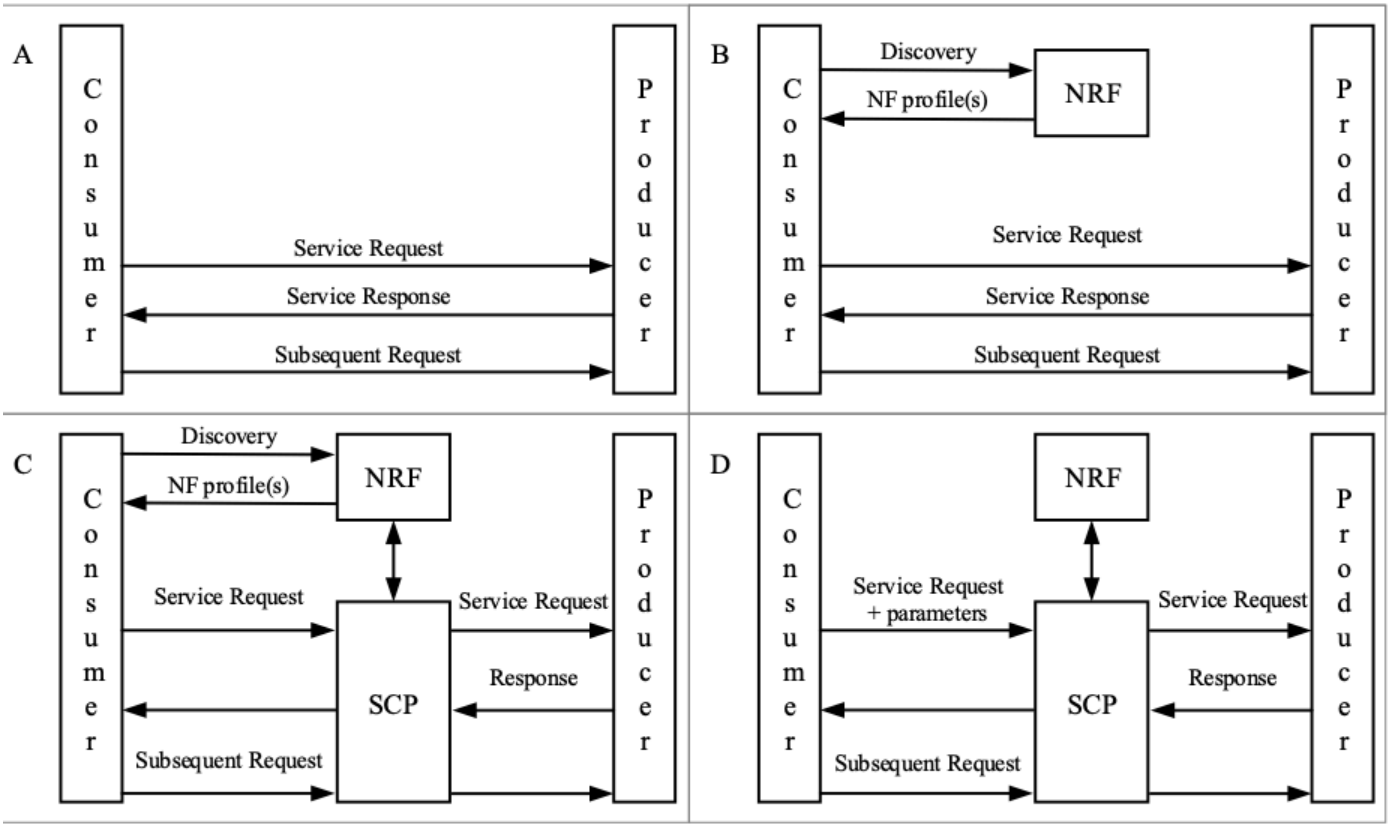


Figura A: (Modelos de comunicação diferentes envolvendo SCP)

Modelo A - Comunicação direta sem interação da Network Repository Function (NRF): Os consumidores são configurados com os "perfis de NF" dos produtores e se comunicam diretamente com um produtor de sua escolha. Esse é o tipo de seleção estática e nem NRF nem SCP são usados.

Modelo B - Comunicação direta com interação NRF: Os clientes realizam a descoberta consultando o NRF. Com base no resultado da descoberta, o consumidor faz a seleção. O consumidor envia o pedido ao produtor selecionado.

Modelo C - Comunicação indireta sem descoberta delegada: Os consumidores descobrem

consultando o NRF. Com base no resultado da descoberta, o consumidor faz a seleção de um conjunto de NF ou de uma instância específica de NF do conjunto de NF. O consumidor envia a solicitação ao SCP contendo o endereço do produtor de serviço escolhido apontando para uma instância de serviço de NF ou um conjunto de instâncias de serviço de NF. Neste último caso, o SCP escolhe uma instância do Serviço de NF. Se possível, o SCP interage com o NRF para obter parâmetros de seleção como localização, capacidade e assim por diante. O SCP roteia a solicitação para a instância do produtor do serviço de NF escolhida.

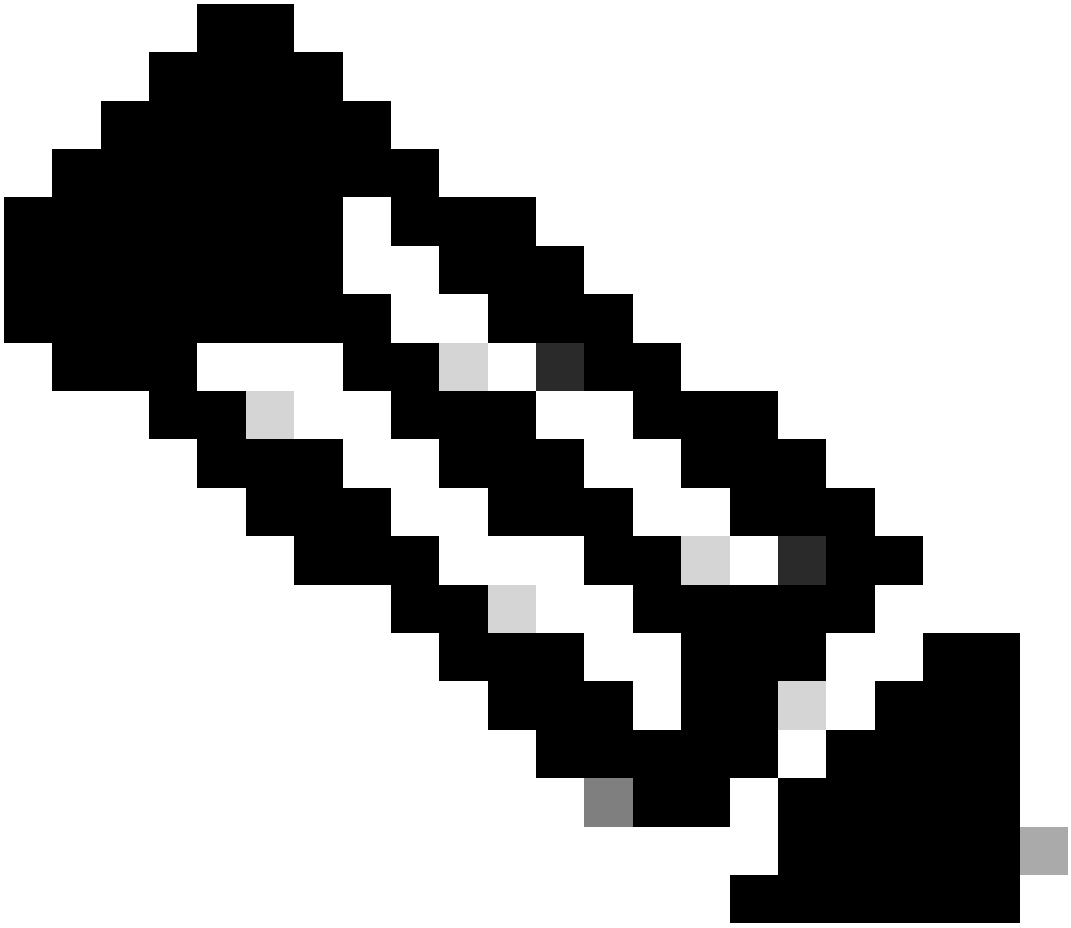
Modelo D - Comunicação indireta com descoberta delegada: Os consumidores não estão envolvidos na descoberta ou na seleção. O consumidor adiciona à solicitação de serviço todos os parâmetros de descoberta e seleção necessários para encontrar um produtor adequado. O SCP usa o endereço de solicitação e os parâmetros de descoberta e seleção na mensagem de solicitação para rotear a solicitação para uma instância de produtor adequada. O SCP pode executar a descoberta com um NRF e obter um resultado de descoberta.

Mergulhe fundo na comunicação baseada em Modelo D: Quando Call Model-D é usado, o consumidor de NF não envia diretamente uma solicitação ao NRF, mas delega essa descoberta ao SCP. O cliente NF envia uma mensagem ao SCP e concatena para cada um desses fatores de descoberta a sequência '3gpp-sbi-discovery' com o nome do fator de Descoberta que será usado se a descoberta da NF for feita através do NRF.

Em um cenário em que o SMF procurará o Unified Data Management (UDM) com nomes de serviço nudm-sdm, os fatores de descoberta serão passados para o SCP:

- Cabeçalho da autoridade: a autoridade transporta o FQDN (Fully Qualified Domain Name, Nome de domínio totalmente qualificado) ou o endereço IP, com prioridade para a configuração do endereço IP.
- 3gpp-sbi-discovery-requester-nf-type: SMF
- 3gpp-sbi-discovery-target-nf-type: UDM
- 3gpp-Sbi-discovery-service-name: nudm-sdm

Figura B: (Comunicação SMF-UDM via SCP modelo D)



Note: O formato de nome 3gpp-sbi-discovery-service está em um formato de cadeia de caracteres simples e não em um formato de matriz de acordo com 3gpp 29.510 e definições de API aberta (estilo 4.7.12.4). No 29.510, 3gpp-sbi-discovery-service-name é mencionado como um formato de matriz.

```
- name: service-names
  in: query
  description: Names of the services offered by the NF
  schema:
    type: array
    items:
      $ref: 'TS29510_Nnrf_NFManagement.yaml#/components/schemas/ServiceName'
    minItems: 1
    uniqueItems: true
  style: form
  explode: false
```

Figura C: (Instantâneo das especificações 29.510)

No entanto, style:form e explode:false convertem a matriz em uma string simples, que é explicada tomando-se um exemplo de OpenAPI.

Assume a parameter named `color` has one of the following values:

```
string -> "blue"
array -> ["blue","black","brown"]
object -> { "R": 100, "G": 200, "B": 150 }
```

The following table shows examples of rendering differences for each value.

style	explode	empty	string	array	object
matrix	false	;color	;color=blue	;color=blue,black,brown	;color=R,1(
matrix	true	;color	;color=blue	;color=blue;color=black;color=brown	;R=100;G=
label	false	.	.blue	.blue.black.brown	.R.100.G.2
label	true	.	.blue	.blue.black.brown	.R=100.G=
form	false	color=	color=blue	color=blue,black,brown	color=R,10
form	true	color=	color=blue	color=blue&color=black&color=brown	R=100&G=
simple	false	n/a	blue	blue,black,brown	R,100,G,2(
simple	true	n/a	blue	blue,black,brown	R=100,G=2
spaceDelimited	false	n/a	n/a	blue%20black%20brown	R%20100%
pipeDelimited	false	n/a	n/a	blue|black|brown	R|100|G|2(
deepObject	true	n/a	n/a	n/a	color[R]=1(

Figura D: (Instantâneo da API aberta: (4.7.12.4 Exemplos de estilo)

Você tem o controle CLI no AMF e no SMF para enviar o parâmetro 3gpp-sbi-discovery-service, pois ele é opcional (pode ser feito dependendo do ambiente de implantação).

No caso do Modelo B, se você pegar o exemplo de comunicação AMF e Authentication Server Function (AUSF), quando o AUSF for descoberto, o AMF enviará o POST para o AUSF com AUSF IP/FQDN e Port.

POST http://<ausf-fqdn>:<port>/nausf-auth/v1/ue-authentications.

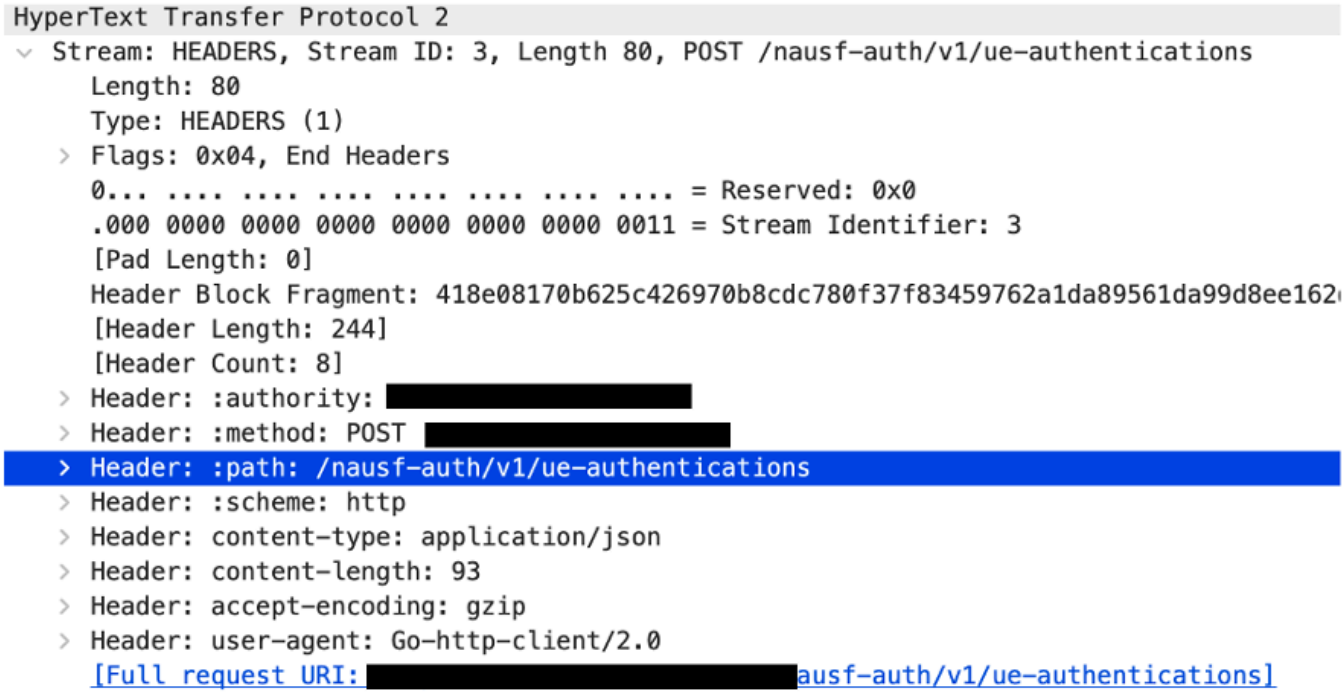


Figura E: (Comunicação AMF-AUSF através do Modelo B)

No Modelo-D, conforme a descoberta é realizada pelo SCP, em vez de POST http(s)://<ausf-fqdn>:<ausf-port>/nausf-auth/v1/ue-authentications, o AMF envia a solicitação de POST modificada que é:

POST http(s)://<scp-fqdn>:<scp-port>/nausf-auth/v1/ue-authentication

Ou

POST http(s)://<scp-fqdn>:<scp-port>/nscp-route/nausf-auth/v1/ue-authentications(if apiroot=nscp-route)

Com

3gpp-Sbi-Discovery-target-nf-type: AUSF

3gpp-Sbi-Discovery-Localidade-preferida: LOC1

3gpp-Sbi-Discovery-service-name

Onde você pode ver que o AMF substituiu a raiz da API (<ausf-fqdn>:<ausf-port>) do AUSF pela raiz da API do SCP.

```
> Header: :authority: [redacted]
> Header: :method: POST
> Header: :path: /nscp-route/nausf-auth/v1/ue-authentications
> Header: :scheme: http
> Header: 3gpp-sbi-discovery-service-names: ["nausf-auth"]
> Header: 3gpp-sbi-discovery-target-nf-type: AUSF
> Header: 3gpp-sbi-discovery-requester-nf-type: AMF
> Header: user-agent: AMF-SLICE-EMBB
> Header: 3gpp-sbi-discovery-target-plmn-list: [{"mcc":[redacted],"mnc":[redacted]]
> Header: content-type: application/json
> Header: content-length: 183
> Header: accept-encoding: gzip
  [Full request URI: http://[redacted]/nscp-route/nausf-auth/v1/ue-authentications]
```

Figura F: (Comunicação AMF-AUSF via SCP-Modelo D)

Os parâmetros 3gpp-sbi-discovery permitem que o SCP recupere a melhor NF e, em seguida, encaminhe a solicitação POST, onde ela substitui a raiz da api do SCP pela raiz da api recebida do NRF após ter recebido a resposta à sua solicitação de descoberta.

Configurações necessárias em AMF/SMF

Para indicar para cada NF (por exemplo, UDM) qual modelo de chamada deve ser usado, a configuração do modelo de seleção de nf é usada dentro do "elemento de rede de perfil" associado.

<#root>

profile network-element udm prf-udm-scp

[...]

nf-selection-model priority <>[local | nrf-query | nrf-query-peer-input | nrf-query-and-scp | scp]

exit

Depois que Model-D é escolhido, os parâmetros de consulta configurados para o elemento de rede associado ainda são usados e passados para o SCP no formato '3gpp-Sbi-Discovery-

<query-param>'.

<#root>

```
[smf] smf(config)# profile network-element udm prf-udm-scp


[smf] smf(config-udm-udm1)# query-params


Possible completions:


 [  chf-supported-plmn  dnn  requester-snssais  tai  target-nf-instance-id  target-plmn ]
```

Eventualmente, o elemento de rede do perfil é mapeado para o perfil Data Network Name (dnn).

<#root>

```
profile dnn ims


 network-element-profiles udm prf-udm-scp


 network-element-profiles scp prf-scp


exit
```

SCP(s) são definidos como elemento de rede.

o perfil de cliente nf e o perfil de tratamento de falhas são mapeados com o elemento de rede.

<#root>

```
profile network-element scp <>
```

```
 nf-client-profile        <>

 failure-handling-profile <>

exit
```

O nf-client-profile do tipo scp-profile detalha as características do endpoint SCP.

Aqui nscp-route pode ser adicionado em api-root.

<#root>

```
profile nf-client nf-type scp

 scp-profile <>

  locality LOC1

   priority 30

   service name type <>

    responsetimeout 4000

    endpoint-profile EP1

     capacity   30
```

```
     api-root   nscp-route

     priority   10

     uri-scheme http

     endpoint-name scp-customer.com

      priority 10

      capacity 50

      primary ip-address ipv4

      primary ip-address port

      fqdn name <>

      fqdn port <>

exit
```

O FQDN SMF é configurado na interface descendente do ponto de extremidade (SBI).

<#root>

```
endpoint sbi
```

relicas 2

nodes 2

fqdn <>

Exemplo de snap de pacotes

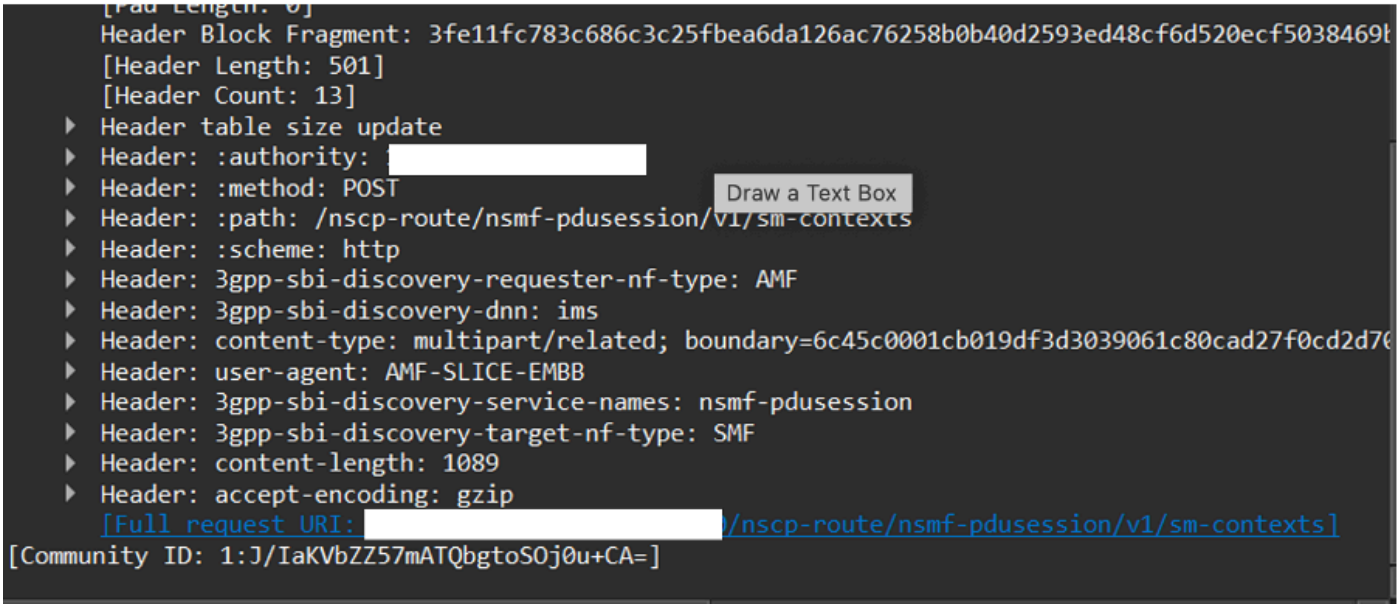


Figura G: (AMF - comunicação nsmf-pdusession SMF via SCP Modelo D)

Você precisa do dnn do perfil para consultar o elemento de rede SCP que acabou de ser configurado.

<#root>

profile dnn <>

network-element-profiles udm <>

network-element-profiles scp <>

exit

Se o tratamento de falha SCP for configurado com ação como repetição, o SMF tentará alternar o SCP com base na configuração SCP e na contagem de repetição.

Se o tratamento de falhas SCP for configurado com ação como repetição e fallback para um nome de serviço e tipo de mensagem específicos, ocorrerá o fallback para o Modelo A.

Esse Perfil de Tratamento de Falhas para SCP (FHSCP) será usado se o erro for disparado do SCP (cabeçalho do servidor indicando SCP) e a configuração do cliente de NF para o peer estiver presente.

<#root>

```
profile nf-client-failure nf-type scp


   profile failure-handling <>


   service name type npcf-smpolicycontrol


    responsetimeout 1800


    message type PcfSmpolicycontrolCreate


    status-code httpv2 0,307,429,500,503-504


     retry  1


     action retry-and-fallback


     exit
```

Exemplo do perfil de cliente nf para a Policy Control Function (PCF) para o cenário em que a ação retry and falback está configurada para o tipo de mensagem PcfSmpolicycontrolCreate:

<#root>

```
profile nf-client nf-type pcf


 pcf-profile <>


  locality LOC1


   priority 1


   service name type npcf-smpolicycontrol


    endpoint-profile epprof


     capacity   10


     priority   1


     uri-scheme http


     endpoint-name ep1


      priority 1


      capacity 10


      primary ip-address ipv4 <>
```

```
 primary ip-address port <>

exit

endpoint-name ep2

 priority 1

 capacity 10

 primary ip-address ipv4 <>

 primary ip-address port <>

exit
```

PODs DNS principais e configuração necessária na camada SMI

Os pods CoreDNS, que fazem parte do namespace do sistema kube, são implantados como um conjunto de réplicas de 2 pods. Esses pods podem ser programados em qualquer um dos dois nós mestre/de controle e não dependem de onde o IP do servidor de nome está configurado no gerenciador de cluster.

No entanto, é recomendável configurar o IP do servidor de nome em todos os nós de controle/mestre, pois você não tem um controle de rotulação para girar os pods CoreDNS conforme seu desejo. Se a rota para servidores de nome não estiver presente em nenhum dos servidores mestre onde o CoreDNS está implantado, a sincronização do cluster SMF/AMF falhará.

Atualmente, o CoreDNS encaminha solicitações DNS para o servidor de nome especificado no arquivo resolv.conf do nó.

'kubectl edit configmap coredns -n kube-system' você tem:

<#root>

```
 {


forward ./etc/resolv.conf{


     max_concurrent 1000


  }
```

Ao verificar /etc/resolv.conf no nó mestre onde o serviço é iniciado, ele deve conter:

<#root>

```
 name server <>


     name server <>
```

Exemplo de configuração de servidor de nome no nó mestre/controle:

<#root>

```
nodes <>


initial-boot netplan vlans <>


dhcp4 false
```

dhcp6 false

addresses [<>]

nameserver addresses [<>]

id <>

link <>

exit

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.