基于SCP模型D的通信深入探讨

目录

简介

本文档介绍Cisco AMF/SMF与第三方NF之间的SCP Model-D通信方法的深入探讨。

先决条件

要求

Cisco 建议您了解以下主题：

- 访问和移动管理功能(AMF)的功能
- 会话管理功能(SMF)的功能
- 服务通信代理(SCP)的功能

使用的组件

本文档不限于特定的软件和硬件版本。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

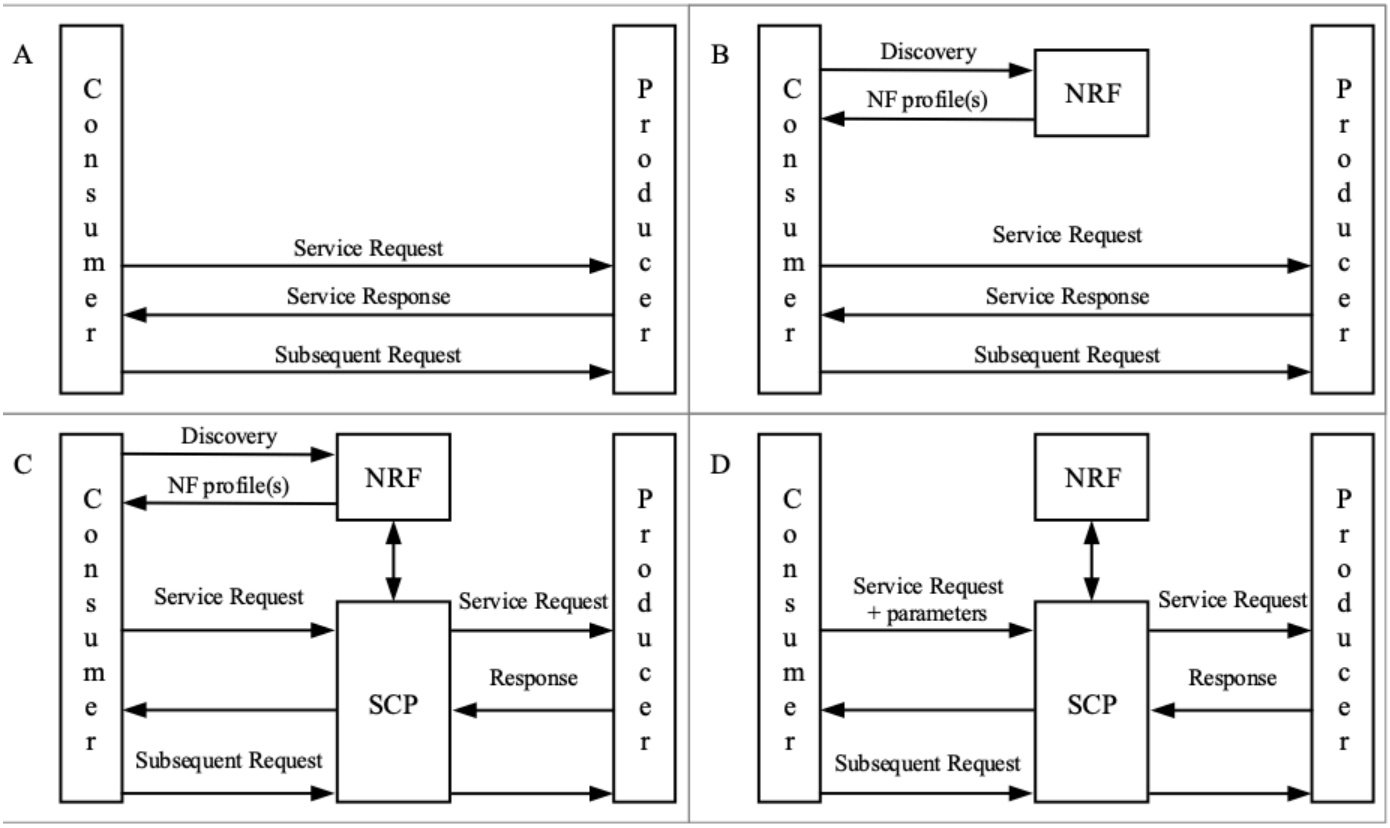
背景信息

世界各地的运营商可以在使用SCP进行网络功能(NF)发现以及随后的NF到NF通信的多种通信模式之间进行选择。本主题涉及各种通信模式的概念以及用户微服务基础设施(SMI)、AMF/SMF上实现基于SCP Model-D的通信所需的呼叫流/配置更改。

架构和解决方案概述

在基于服务的体系结构(SBA)中，SCP充当中介，通过处理路由、负载均衡和服务发现来促进NF之间的间接通信，最终简化基于服务的体系结构。

3GPP 23.501 Annex-E详细介绍了5GC部署中NF之间的四种通信模型。



图A:（涉及SCP的不同通信模式）

Model-A — 没有网络存储库功能(NRF)交互的直接通信：消费者配置有生产者的“NF配置文件”，并直接与自己选择的生产者通信。这是静态选择的类型，既不使用NRF，也不使用SCP。

Model-B — 与NRF交互的直接通信：消费者通过查询NRF执行发现。根据发现结果，消费者进行选择。使用者将请求发送到选定的生产者。

Model-C — 无授权发现的间接通信：消费者通过查询NRF发现。基于发现结果，消费者选择NF集或NF集的特定NF实例。消费者将请求发送到包含指向NF服务实例或一组NF服务实例的所选服务生成者地址的SCP。在后一种情况下，SCP会选择NF服务实例。如果可能，SCP会与NRF进行交互，以获取位置、容量等选择参数。SCP将请求路由到所选NF服务制造者实例。

Model-D — 与授权发现的间接通信：消费者不参与发现或选择。消费者向服务请求添加为找到合适的生产者所需的任何必要的发现和选择参数。SCP使用请求地址以及请求消息中的发现和选择参数，以便将请求路由到适当的生产者实例。SCP可以通过NRF执行发现并获得发现结果。

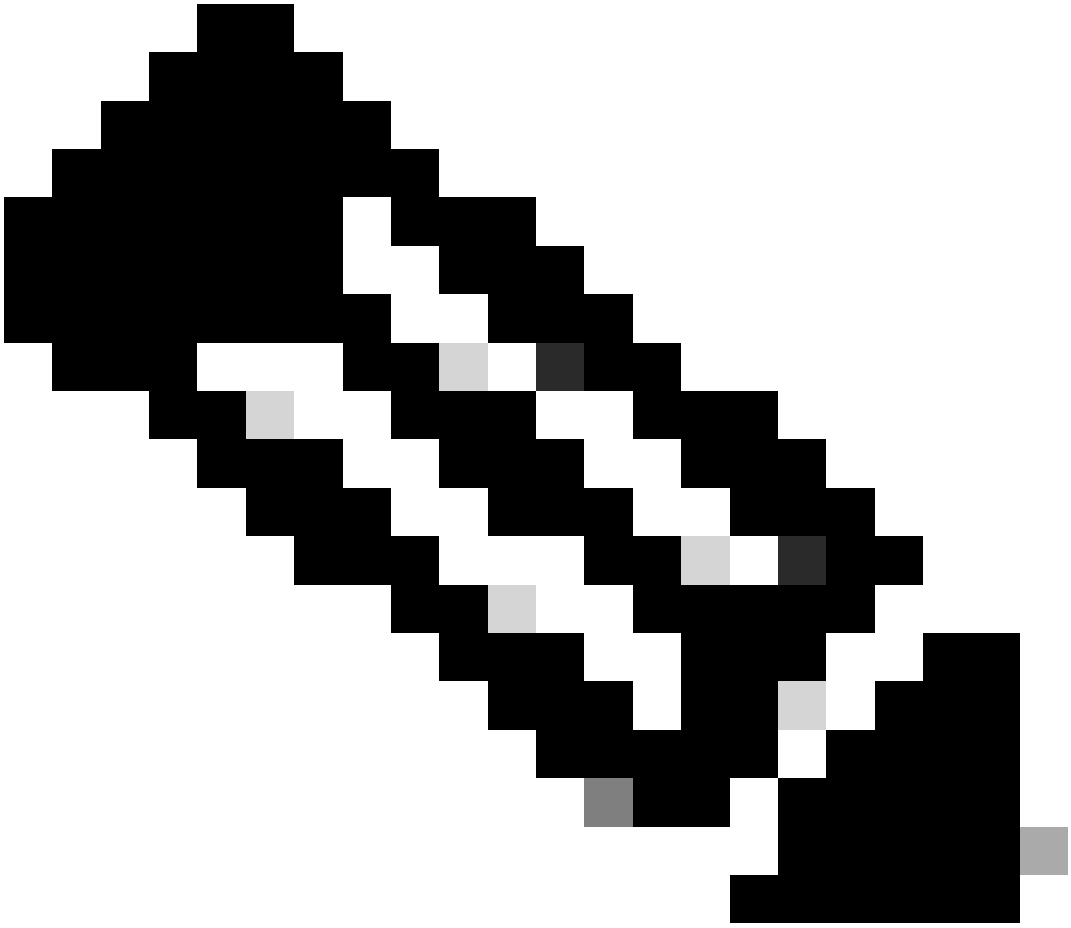
深入探讨基于Model-D的通信：使用呼叫模型D时，NF使用者不会直接向NRF发送请求，而是将此发现委托给SCP。NF客户端向SCP发送消息，并为每个发现因子连接字符串“3gpp-sbi-discovery”和发现因子的名称，如果通过NRF完成NF发现，将使用该发现因子的名称。

对于SMF将查找服务名称为nudm-sdm的统一数据管理(UDM)的场景，发现因素将传递给SCP:

- 授权信头：授权可以传送完全限定域名(FQDN)或IP地址，并为IP地址配置赋予优先级。
- 3gpp-sbi-discovery-requester-nf-type:SMF
- 3gpp-sbi-discovery-target-nf-type:UDM
- 3gpp-Sbi-discovery-service-name:nudm-sdm

```
> Header: :authority: [redacted]
> Header: :method: PUT
> Header: :path: /nudm-uecm/v1/imsi-[redacted]/registrations/smf-registrations/2
> Header: :scheme: http
> Header: 3gpp-sbi-discovery-requester-nf-type: SMF
> Header: 3gpp-sbi-discovery-target-plmn-list: [{"mcc":[redacted],"mnc":[redacted]]
> Header: 3gpp-sbi-discovery-supi: imsi-[redacted]
> Header: content-type: application/json
> Header: user-agent: SMF-[redacted]
> Header: 3gpp-sbi-discovery-target-nf-type: UDM
> Header: content-length: 239
> Header: accept-encoding: gzip
  [Full request URI: [redacted]/nudm-uecm/v1/imsi-[redacted]/registrations/smf-reg
  [Response in frame: 40]
```

图B:（通过SCP型号D的SMF-UDM通信）



注意：3gpp-sbi-discovery-service name格式为纯字符串格式，而不符合3gpp 29.510和开放式API定义（4.7.12.4样式）的数组格式。在29.510 3gpp-sbi-discovery-service-name中，称为数组格式。

```
- 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
```

图C:（29.510规范中的快照）

但是，style:form和explode:false会将数组转换为一个纯字符串，这可以通过从OpenAPI中获取示例

来解释。

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.

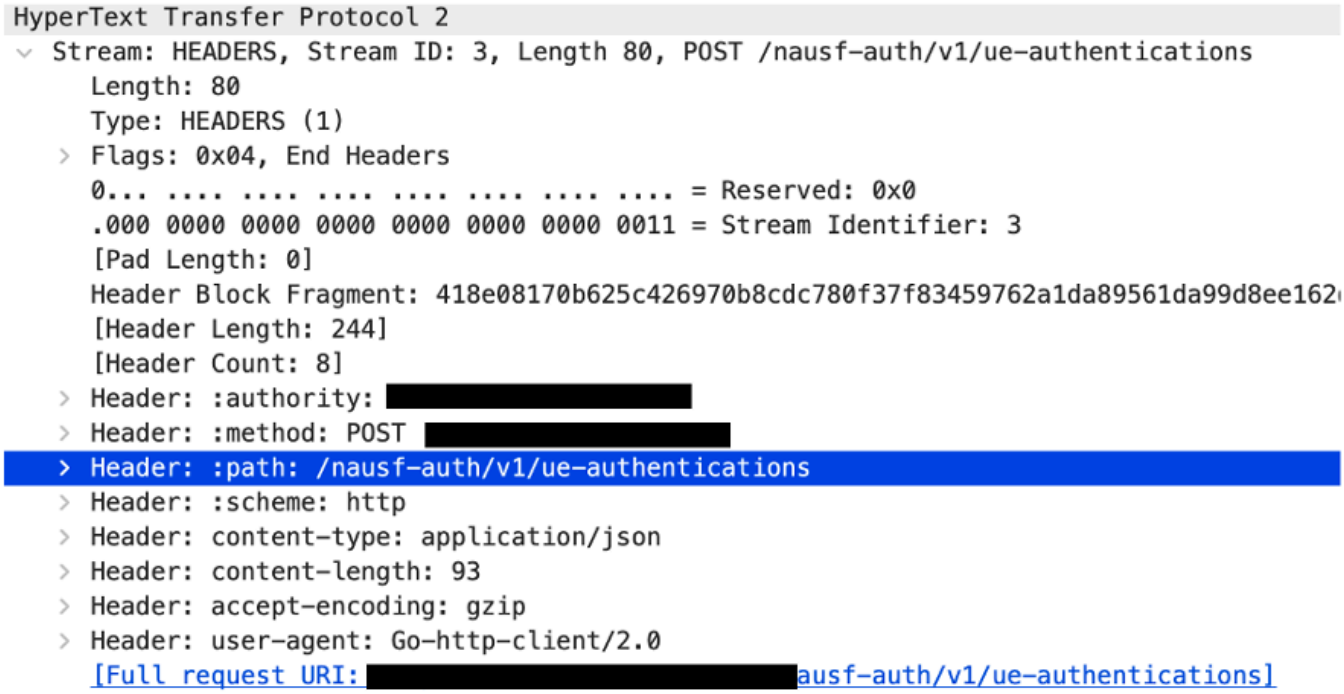
style	explode	empty	string	array	object
matrix	false	;color	;color=blue	;color=blue,black,brown	;color=R,1(
matrix	true	;color	;color=blue	;color=blue;color=black;color=brown	;R=100;G=
label	false	.	.blue	.blue.black.brown	.R.100.G.2
label	true	.	.blue	.blue.black.brown	.R=100.G=
form	false	color=	color=blue	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(

图D:(来自开放式API的快照：（4.7.12.4样式示例）

您在AMF和SMF中都具有CLI控制以发送参数3gpp-sbi-discovery-service，因为这是可选的（可根据部署环境执行）。

对于Model-B，如果您以AMF和身份验证服务器功能(AUSF)通信为例，一旦发现AUSF，AMF会使用AUSF IP/FQDN和端口将POST发送到AUSF。

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



图E:（通过模型B的AMF-AUSF通信）

在Model-D中，由于发现由SCP执行，而不是POST http(s)://<ausf-fqdn>:<ausf-port>/nausf-auth/v1/ue-authentications时，AMF发送修改后的POST请求，即：

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

或者

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

使用

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

3gpp-Sbi-Discovery-Preferred-locality:LOC1

3gpp-Sbi-Discovery-service-name

其中您可以看到AMF已将AUSF的api-root(<ausf-fqdn>:<ausf-port>)替换为SCP的api-root。

```
> 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]
```

图F:（通过SCP — 型号D的AMF-AUSF通信）

3gpp-sbi-discovery参数允许SCP检索最佳NF，然后转发POST请求，其中SCP的api-root在接收到对其发现请求的响应后用从NRF接收的api-root替换SCP的api-root。

AMF/SMF所需的配置

为了指示每个NF（例如，UDM）必须使用哪个呼叫模型，在关联的“profile network-element”中使用nf-selection-model配置。

<#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

选择Model-D后，仍使用为关联网络元素配置的查询参数，并将以“3gpp-Sbi-Discovery-<query-param>”格式传递给SCP。

<#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 ]
```

最后，配置文件网络元素映射到配置文件数据网络名称(dnn)。

```
<#root>
profile dnn ims

 network-element-profiles udm prf-udm-scp

 network-element-profiles scp prf-scp

exit
```

SCP定义为网络元素。

nf-client-profile和failure-handling profile与network-element映射。

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

 nf-client-profile        <>
```

```
 failure-handling-profile <>

exit
```

类型为scp-profile的nf-client-profile详细描述了SCP终端的特征。

此处nscp-route可以添加到api-root中。

<#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
```

SMF FQDN在终端南向接口(SBI)中配置。

<#root>

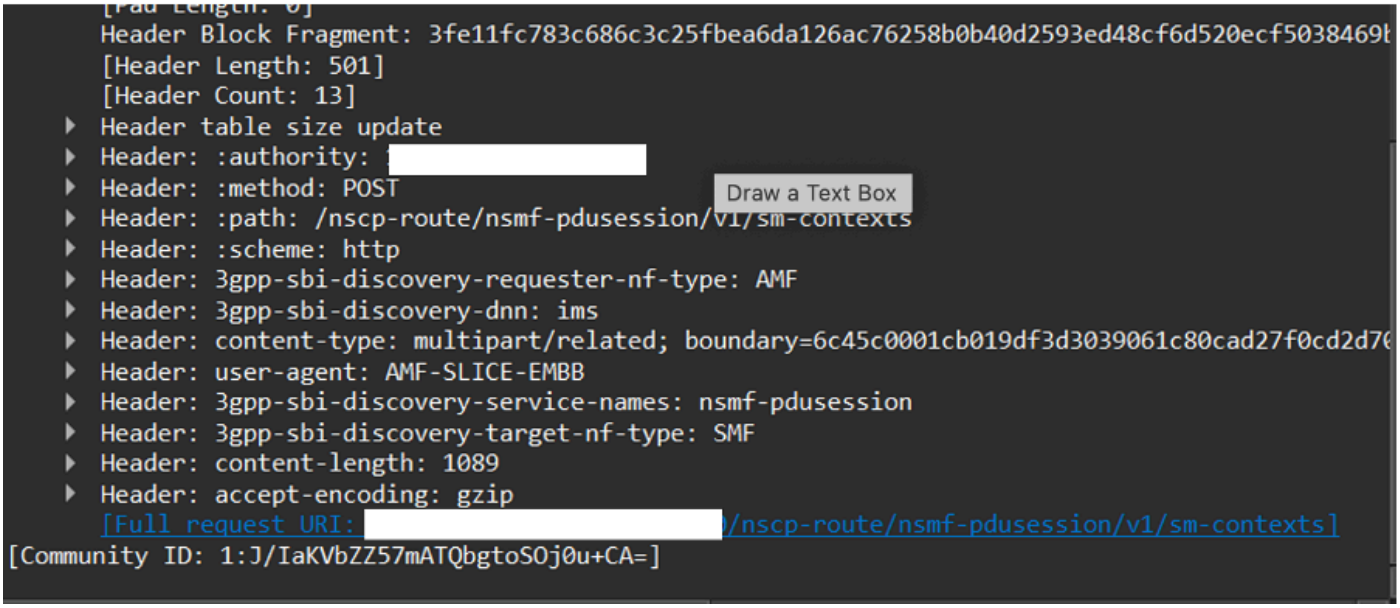
```
endpoint sbi

relicas 2

nodes 2
```

```
fqdn <>
```

数据包快照示例



图G:（通过SCP型号D进行的AMF-SMF nsmf-pdusession通信）

您需要从配置文件dnn引用刚才配置的SCP网络元素。

<#root>

```
profile dnn <>
```

```
 network-element-profiles udm <>
```

```
 network-element-profiles scp <>
```

```
exit
```

如果SCP故障处理配置为使用重试操作，则SMF将根据SCP配置和重试计数尝试备用SCP。

如果将SCP故障处理配置为对特定服务名称和消息类型执行重试和回退操作，则会回退到型号A。

如果从SCP（指示了SCP的服务器报头）触发错误且对等设备的NF-client配置存在，则使用此失败的SCP处理配置文件(FHSCP)。

<#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
```

为消息类型PcfSmpolicycontrolCreate配置操作retry和falback的方案的策略控制功能(PCF)的nf-client配置文件示例：

<#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
```

SMI层所需的核心DNS POD和配置

CoreDNS Pod是Web系统命名空间的一部分，部署为2 Pod复制集。这些Pod可在两个主/控制节点中的任意一个上安排，并且不依赖于在集群管理器中配置名称服务器IP的位置。

但是，建议您在所有控制/主节点中配置名称服务器IP，因为您没有标签控制来根据自己的意愿旋转CoreDNS Pod。如果在部署CoreDNS的任何主设备上不存在指向名称服务器的路由，则SMF/AMF集群同步将失败。

目前，CoreDNS会将DNS请求转发到节点resolv.conf文件中指定的名称服务器。

'kubectl edit configmap coredns -n kube-system'您有：

<#root>

```
 {

forward ./etc/resolv.conf{

     max_concurrent 1000

  }
```

在启动服务的主节点上检查/etc/resolv.conf时，它必须包含：

<#root>

```
 name server <>
```

```
     name server <>
```

主/控制节点中的名称服务器配置示例：

<#root>

```
nodes <>
```

```
initial-boot netplan vlans <>
```

```
dhcp4 false
```

```
dhcp6 false
```

```
addresses [<>]
```

```
nameserver addresses [<>]
```

```
id <>
```

```
link <>
```

```
exit
```

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。