SCP 모델-D 기반 커뮤니케이션에 대한 심층 분석

목차

[소개](#)

[사전 요구 사항](#)

- [요구 사항](#)
- [사용되는 구성 요소](#)

[배경 정보](#)

[아키텍처 및 솔루션 개요](#)

[AMF/SMF에 필요한 구성](#)

- [샘플 패킷 스냅](#)
- [SMI 레이어에 필요한 코어 DNS POD 및 컨피그레이션](#)

소개

이 문서에서는 Cisco AMF/SMF와 서드파티 NF 간의 SCP 모델-D 통신 접근 방식을 심층적으로 살펴봅니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- AMF(Access and Mobility Management Function) 기능
- SMF(Session Management Function)의 기능
- SCP(서비스 통신 프록시)의 기능

사용되는 구성 요소

이 문서는 특정 소프트웨어 및 하드웨어 버전으로 한정되지 않습니다.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

배경 정보

전 세계 통신사는 NF(Network Function) 탐색을 위해 SCP를 사용하고 NF 통신에 대한 후속 NF를 사용하는 여러 통신 모델 중에서 선택할 수 있습니다. 이 항목에서는 SCP Model-D 기반 통신을 위해 SMI(Subscriber Microservices Infrastructure), AMF/SMF에 필요한 다양한 통신 모델 및 통화 흐름/컨피그레이션 변경에 대한 개념을 다룹니다.

아키텍처 및 솔루션 개요

SBA(Service Based Architecture)에서 SCP는 중재자 역할을 하여 라우팅, 로드 밸런싱 및 서비스 검색을 처리하여 NF 간의 간접 통신을 촉진함으로써 궁극적으로 서비스 기반 아키텍처를 간소화합니다.

3GPP 23.501 Annex-E는 5GC 구축에서 NF 간의 네 가지 통신 모델을 상세히 설명하고 있다.

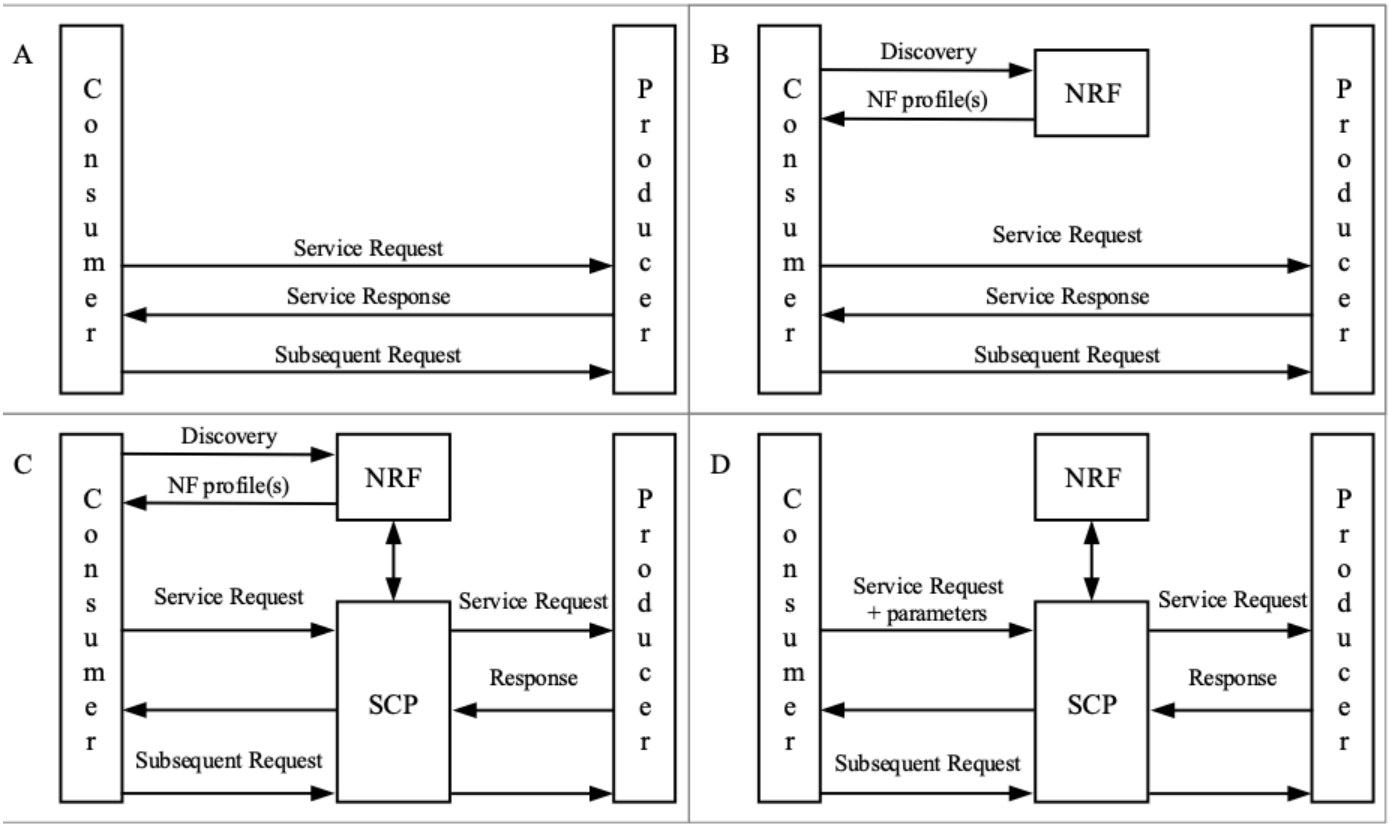


그림 A: (SCP와 관련된 다양한 커뮤니케이션 모델)

모델 A - NRF(Network Repository Function) 상호 작용 없는 직접 통신: 소비자는 생산자의 'NF 프로파일'로 구성되고 자신이 선택한 생산자와 직접 커뮤니케이션합니다. 이는 고정 선택 유형이며 NRF와 SCP는 사용되지 않습니다.

Model-B - NRF 상호 작용을 통한 직접 통신: 소비자는 NRF에 쿼리하여 검색을 수행합니다. 검색 결과에 따라 소비자는 선택을 수행합니다. 소비자가 선택한 생산자에게 요청을 보냅니다.

Model-C - 위임된 검색이 없는 간접 통신: 소비자는 NRF에 쿼리하여 이를 발견합니다. 검색 결과에 기초하여, 소비자는 NF 세트 또는 NF 세트의 특정 NF 인스턴스의 선택을 수행한다. 소비자는 NF 서비스 인스턴스 또는 NF 서비스 인스턴스 집합을 가리키는 선택된 서비스 생산자의 주소를 포함하는 SCP에 요청을 전송한다. 후자의 경우 SCP는 NF 서비스 인스턴스를 선택합니다. 가능하면 SCP는 위치, 용량 등과 같은 선택 매개변수를 가져오기 위해 NRF와 상호 작용합니다. SCP는 요청을 선택한 NF 서비스 프로듀서 인스턴스로 라우팅합니다.

Model-D - 위임된 검색을 통한 간접 통신: 소비자는 발견이나 선택에 관여하지 않는다. 소비자는 적절한 생산자를 찾는 데 필요한 모든 검색 및 선택 파라미터를 서비스 요청에 추가한다. SCP는 요청을 적절한 프로듀서 인스턴스로 라우팅하기 위해 요청 메시지에 요청 주소와 검색 및 선택 매개변

수를 사용합니다. SCP는 NRF로 검색을 수행하고 검색 결과를 얻을 수 있습니다.

Model-D 기반 커뮤니케이션에 대한 심층 분석: Call Model-D를 사용하면 NF 소비자가 NRF에 직접 요청을 보내지 않고 SCP에 이 검색을 위임합니다. NF 클라이언트는 SCP에 메시지를 전송하고 이러한 각 검색 인자에 대해 NRF를 통해 NF 검색을 수행할 경우 사용할 검색 인자의 이름으로 '3gpp-sbi-discovery' 문자열을 연결합니다.

SMF가 service-names nudm-sdm을 사용하는 UDM(Unified Data Management)을 찾는 시나리오의 경우 검색 요소가 SCP에 전달됩니다.

- 권한 헤더: 권한은 FQDN(Fully Qualified Domain Name) 또는 IP 주소를 전달하며, IP 주소 컨피그레이션에 우선 순위를 부여합니다.
- 3gpp-sbi-discovery-requester-nf-type: SMF
- 3gpp-sbi-discovery-target-nf-type: UDM
- 3gpp-Sbi-discovery-service-name: nudm-sdm

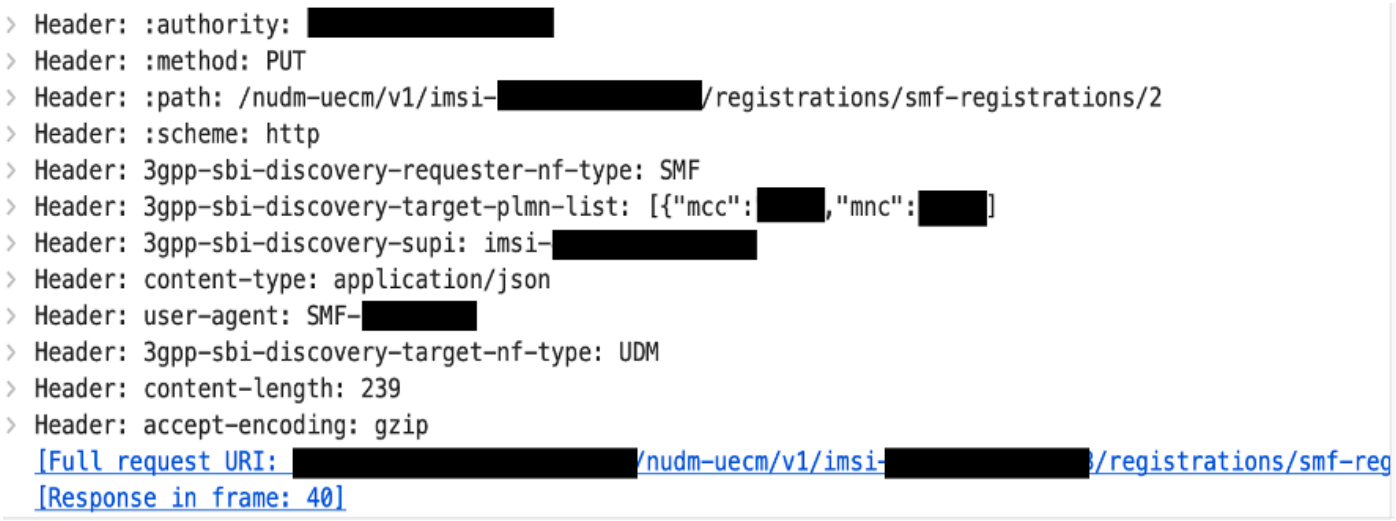
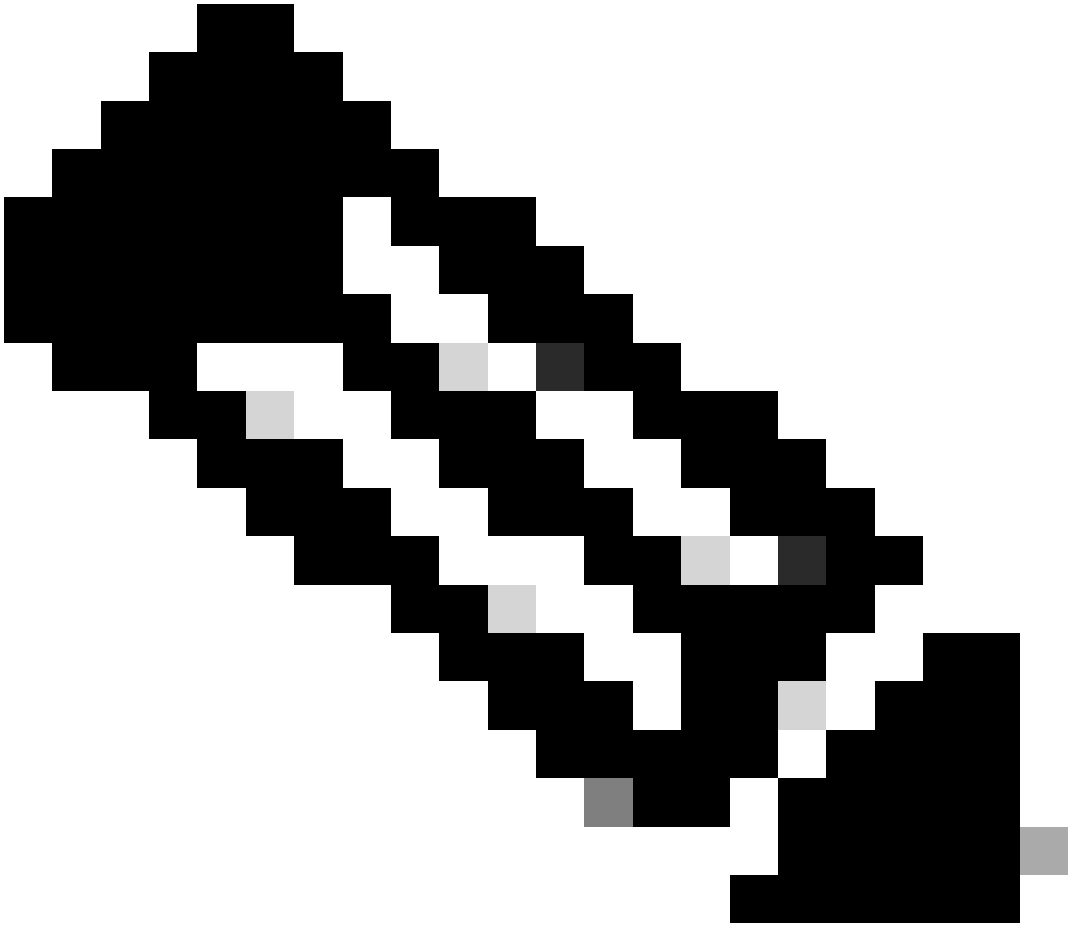


그림 B: (SCP 모델 D를 통한 SMF-UDM 통신)



참고: 3gpp-sbi-discovery-service name 형식은 3gpp 29.510 및 오픈 API 정의(4.7.12.4 스타일)에 따라 배열 형식이 아닌 일반 문자열 형식입니다. 29.510 3gpp-sbi-discovery-service-name에서는 배열 형식으로 언급됩니다.

```
- 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
```

그림 C: (29.510 사양의 스냅샷)

그러나 style:form 및 explode:false는 배열을 일반 문자열로 변환하며, 이는 OpenAPI의 예를 통해

설명됩니다.

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(

그림 D: (Open API의 스냅샷: (4.7.12.4 스타일 예)

AMF와 SMF 모두에서 CLI를 제어하여 3gpp-sbi-discovery-service 매개변수를 선택 사항으로서 전송합니다(구축 환경에 따라 수행 가능).

Model-B의 경우 AMF 및 AUSF(Authentication Server Function) 통신의 예를 들어 보면, AUSF가 검색되면 AMF는 AUSF IP/FQDN 및 Port를 사용하여 POST를 AUSF로 전송합니다.

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

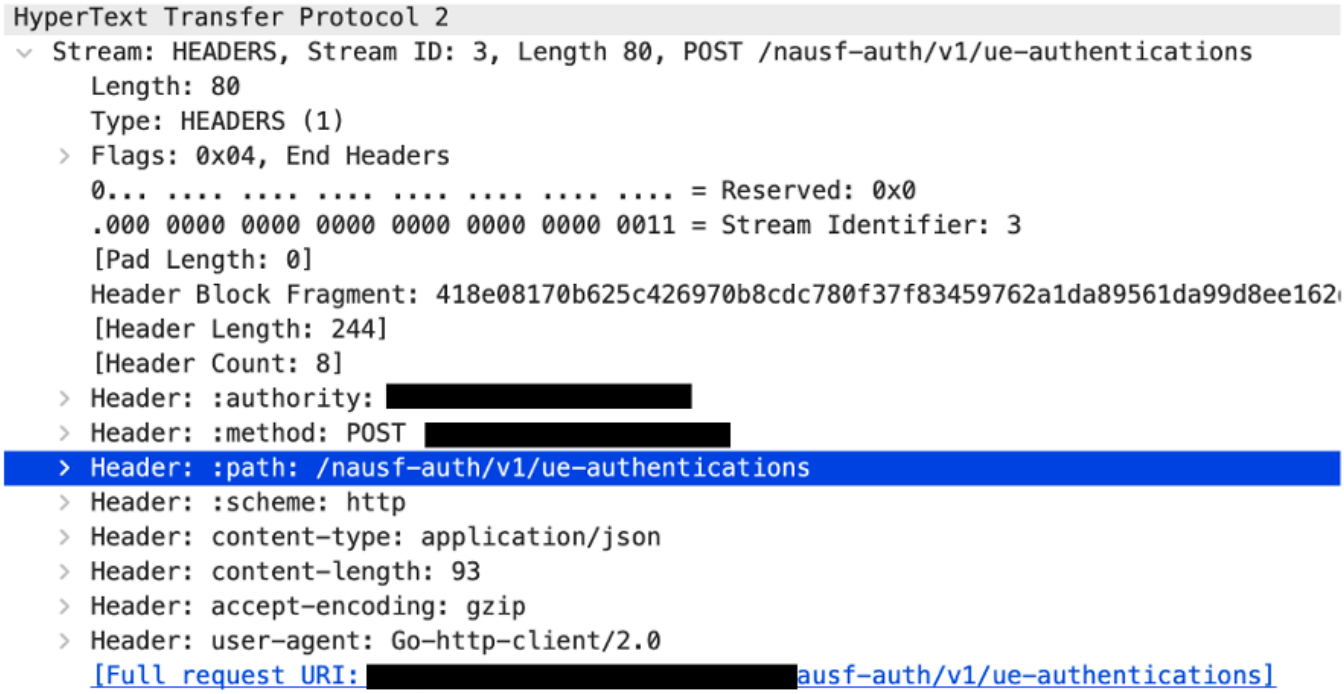


그림 E: (모델 B를 통한 AMF-AUSF 통신)

Model-D에서 SCP가 검색을 수행하므로 POST http(s)://<ausf-fqdn>:<ausf-port>/nausf-auth/v1/ue-authentication 대신 AMF가 수정된 POST 요청을 보냅니다.

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

또는

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

포함

3gpp-Sbi-Discovery-target-nf-type: 호주 연방 공화국

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

3gpp-Sbi-Discovery-service-name

여기서 AMF가 AUSF의 api-root(<ausf-fqdn>:<ausf-port>)를 SCP의 api-root로 대체했음을 확인할 수 있습니다.

```
> 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]
```

그림 F: (SCP-Model D를 통한 AMF-AUSF 통신)

3gpp-sbi-discovery 매개변수를 사용하면 SCP가 최상의 NF를 검색한 다음 POST 요청을 전달할 수 있습니다. 여기서 SCP의 api-root를 검색 요청에 대한 응답을 수신한 후 NRF에서 수신한 api-root로 교체합니다.

AMF/SMF에 필요한 구성

각 NF(예: UDM)에 대해 어떤 통화 모델을 사용해야 하는지 나타내기 위해 nf-selection-model 컨피그레이션이 연결된 'profile network-element' 내에서 사용됩니다.

<#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
```

Model-D를 선택하면 연결된 network-element에 대해 구성된 쿼리 파라미터가 계속 사용되며 '3gpp-Sbi-Discovery-<query-param>' 형식으로 SCP에 전달됩니다.

<#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 ]
```

결국 프로파일 network-element는 프로파일 dnn(Data Network Name)에 매핑됩니다.

<#root>

```
profile dnn ims


 network-element-profiles udm prf-udm-scp


 network-element-profiles scp prf-scp


exit
```

SCP는 network-element로 정의됩니다.

nf-client-profile 및 장애 처리 프로파일은 network-element와 매핑됩니다.

<#root>

```
profile network-element scp <>


 nf-client-profile        <>
```

```
 failure-handling-profile <>

exit
```

scp-profile 유형의 nf-client-profile은 SCP 엔드포인트의 특성을 세부적으로 설명합니다.

여기서 nscp-route는 api-root에 추가할 수 있습니다.

<#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 FQDN은 엔드포인트 사우스바운드 인터페이스(SBI)에서 구성됩니다.

<#root>

```
endpoint sbi

relicas 2
```

nodes 2

fqdn <>

샘플 패킷 스냅

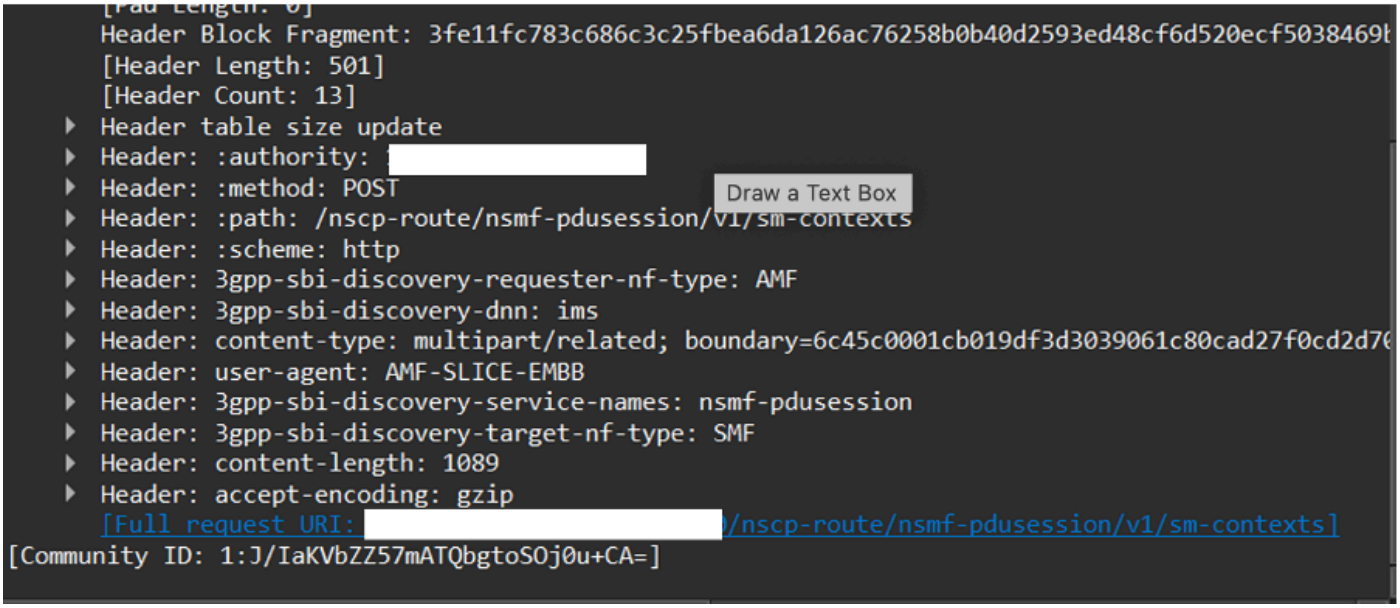


그림 G: (SCP 모델 D를 통한 AMF- SMF nsmf-pdussession 통신)

프로파일 dnn에서 방금 구성된 SCP 네트워크 요소를 참조해야 합니다.

<#root>

profile dnn <>

network-element-profiles udm <>

network-element-profiles scp <>

exit

SCP 실패 처리가 재시도로서 작업으로 구성된 경우 SMF는 SCP 컨피그레이션 및 재시도 횟수를 기반으로 대체 SCP를 시도합니다.

특정 서비스 이름 및 메시지 유형에 대한 retry-and-fallback 작업으로 SCP 실패 처리를 구성하면 Model-A로 대체됩니다.

이 SCP(FHSCP)용 오류 처리 프로필은 SCP(SCP를 나타내는 서버 헤더)에서 오류가 발생하고 피어에 대한 NF-client 컨피그레이션이 있는 경우 사용됩니다.

<#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
```

작업 재시도 및 폴백이 메시지 유형 PcfSmpolicycontrolCreate에 대해 구성된 시나리오의 PCF(Policy Control Function)에 대한 nf-client 프로파일의 예:

<#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 레이어에 필요한 코어 DNS POD 및 컨피그레이션

kube-system 네임스페이스의 일부인 CoreDNS 포드는 2포드 복제 세트로 배포됩니다. 이러한 포드는 두 마스터/제어 노드 중 하나에서 스케줄링할 수 있으며 네임서버 IP가 클러스터 관리자에서 구성되는 위치에 종속되지 않습니다.

그러나 원하는 대로 CoreDNS 포드를 스핀하는 레이블 컨트롤이 없으므로 모든 제어/마스터 노드에서 네임서버 IP를 구성하는 것이 좋습니다. 이름 서버에 대한 경로가 CoreDNS가 구축된 마스터에 없는 경우 SMF/AMF 클러스터 동기화가 실패합니다.

현재 CoreDNS는 DNS 요청을 노드 resolv.conf 파일에 지정된 네임서버로 전달합니다.

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

<#root>

```
 {

forward ./etc/resolv.conf{
```

```
    max_concurrent 1000

  }
```

서비스가 시작되는 마스터 노드에서 /etc/resolv.conf을 확인할 때 다음 사항을 포함해야 합니다.

<#root>

```
 name server <>

    name server <>
```

마스터/제어 노드의 네임서버 컨피그레이션 예:

<#root>

```
nodes <>

initial-boot netplan vlans <>

dhcp4 false

dhcp6 false

addresses [<>]

nameserver addresses [<>]
```

id <>

link <>

exit

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.