Red Hat Enterprise Linux 9

ID 관리에서 DNS 작업

Red Hat Enterprise Linux 9의 Identity Management와 통합된 DNS(Domain Name Service) 관리

Last Updated: 2022-07-25

Red Hat Enterprise Linux 9 ID 관리에서 DNS 작업

Red Hat Enterprise Linux 9의 Identity Management와 통합된 DNS(Domain Name Service) 관리

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

법적 공지

초록

이 문서 컬렉션은 Red Hat Enterprise Linux 9의 ID 관리에서 DNS 구성, 영역, 위치 및 표준화를 관리하는 방법에 대한 지침을 제공합니다.

차례

보다 포괄적 수용을 위한 오픈 소스 용어 교체

Red Hat은 코드, 문서, 웹 속성에서 문제가 있는 용어를 교체하기 위해 최선을 다하고 있습니다. 먼저 마스터(master), 슬레이브(slave), 블랙리스트(blacklist), 화이트리스트(whitelist) 등 네 가지 용어를 교체하고 있습니다. 이러한 변경 작업은 작업 범위가 크므로 향후 여러 릴리스에 걸쳐 점차 구현할 예정입니다. 자세한 내용은 CTO Chris Wright의 메시지 를 참조하십시오.

ID 관리에서 계획된 용어 교체는 다음과 같습니다.

- ***블록 목록*** 대체 *블랙리스트*
- ***허용 목록*** 대체 *허용 목록*
- ***보조*** 대체 *슬레이브*
- 컨텍스트에 따라 *master* 라는 단어가 보다 정확한 언어로 교체될 것입니다.
  - ***IdM 서버*** 대체 *IdM 마스터*
  - ***CA 갱신 서버는*** *CA 갱신 마스터*로 대체
  - ***CRL 게시자 서버는*** *CRL master*를 대체합니다.
  - ***multi-supplier*** 는 *멀티 마스터*교체

RED HAT 문서에 관한 피드백 제공

문서 개선을 위한 의견을 보내 주십시오. Red Hat이 어떻게 이를 개선할 수 있는지 알려 주십시오.

- 특정 문구에 대한 간단한 의견 작성 방법은 다음과 같습니다.
  1. 문서가 *Multi-page HTML* 형식으로 표시되는지 확인합니다. 또한 문서 오른쪽 상단에 **피드백** 버튼이 있는지 확인합니다.
  2. 마우스 커서를 사용하여 주석 처리하려는 텍스트 부분을 강조 표시합니다.
  3. 강조 표시된 텍스트 아래에 표시되는 **피드백 추가** 팝업을 클릭합니다.
  4. 표시된 지침을 따릅니다.
- Bugzilla를 통해 피드백을 제출하려면 새 티켓을 생성합니다.
  1. Bugzilla 웹 사이트로 이동합니다.
  2. 구성 요소로 **문서**를 사용합니다.
  3. **설명** 필드에 문서 개선을 위한 제안 사항을 기입하십시오. 관련된 문서의 해당 부분 링크를 알려주십시오.
  4. **버그 제출**을 클릭합니다.

1장. ANSIBLE 플레이북을 사용하여 IDM의 글로벌 DNS 구성 관리

Red Hat Ansible Engine **dnsconfig** 모듈을 사용하면 IdM(Identity Management) DNS에 대한 글로벌 구성을 구성할 수 있습니다. 글로벌 DNS 구성에 정의된 설정은 모든 IdM DNS 서버에 적용됩니다. 그러나 글로벌 구성은 특정 IdM DNS 영역의 구성보다 우선 순위가 낮습니다.

dnsconfig 모듈은 다음 변수를 지원합니다.

- 글로벌 전달자, 특히 IP 주소 및 통신에 사용되는 포트입니다.
- 글로벌 전달 정책: only, first 또는 none입니다. 이러한 유형의 DNS 전달 정책에 대한 자세한 내용은 IdM의 DNS 전달 정책을 참조하십시오.
- 정방향 조회 및 역방향 조회 영역의 동기화입니다.

사전 요구 사항

- DNS 서비스가 IdM 서버에 설치되어 있습니다. 통합 DNS를 사용하여 IdM 서버를 설치하는 방법에 대한 자세한 내용은 다음 링크 중 하나를 참조하십시오.
  - IdM 서버 설치: 통합 DNS와 함께 통합 CA를 루트 CA로 설치
  - IdM 서버 설치: 루트 CA로 외부 CA를 사용하는 통합 DNS 사용
  - IdM 서버 설치: CA없이 통합 DNS 사용

이 장에는 다음 섹션이 포함됩니다.

- NetworkManager에서 /etc/resolv.conf의 글로벌 전달자가 제거되지 않도록 하는 방법
- Ansible을 사용하여 IdM에 DNS 글로벌 전달자가 있는지 확인
- Ansible을 사용하여 IdM에 DNS 글로벌 전달자가 없는지 확인
- ipadnsconfig ansible-freeipa 모듈의 **action: member** 옵션
- IdM의 DNS 전달 정책 소개
- Ansible 플레이북을 사용하여 IdM DNS 글로벌 구성에 첫 번째 정책이 설정되어 있는지 확인합니다.
- Ansible 플레이북을 사용하여 IdM DNS에서 글로벌 전달자가 비활성화되었는지 확인
- Ansible 플레이북을 사용하여 IdM DNS에서 정방향 및 역방향 조회 영역의 동기화가 비활성화되었는지 확인합니다.

1.1. NETWORKMANAGER에서 /ETC/RESOLV.CONF의 글로벌 전달자가 제거되지 않도록 하는 방법

통합 DNS를 사용하여 IdM(Identity Management)을 설치하면 **127.0.0.1** localhost 주소를 가리키도록 **/etc/resolv.conf** 파일이 설정됩니다.

```
# Generated by NetworkManager
search idm.example.com
nameserver 127.0.0.1
```

DHCP(**Dynamic Host Configuration Protocol**)를 사용하는 네트워크와 같은 특정 환경에서는 **NetworkManager** 서비스가 **/etc/resolv.conf** 파일에 대한 변경 사항을 되돌릴 수 있습니다. DNS 구성을 영구적으로 만들기 위해 IdM DNS 설치 프로세스는 다음과 같은 방식으로 **NetworkManager** 서비스를 구성합니다.

1. DNS 설치 스크립트는 검색 순서 및 DNS 서버 목록을 제어하기 위해 **/etc/NetworkManager/conf.d/zzz-ipa.conf NetworkManager** 구성 파일을 생성합니다.

```
# auto-generated by IPA installer
[main]
dns=default

[global-dns]
searches=$DOMAIN

[global-dns-domain-*]
servers=127.0.0.1
```

2. **NetworkManager** 서비스가 다시 로드됩니다. 이 파일은 항상 **/etc/NetworkManager/conf.d/** 디렉터리에 마지막 파일의 설정을 사용하여 **/etc/resolv.conf** 파일을 생성합니다. 이 경우 **zzz-ipa.conf** 파일입니다.

중요

/etc/resolv.conf 파일을 수동으로 수정하지 마십시오.

1.2. ANSIBLE을 사용하여 IDM에 DNS 글로벌 전달자가 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS 글로벌 전달자가 있는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 포트 53에 있는 IP(Internet Protocol) v4 주소가 **7.7.9.9** 인 DNS 서버로 DNS 글로벌 전달자가 있는지, 포트 **53** 에서 **2001:db8::1:0** 의 IP v6 주소가 있는지 확인합니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.
- IdM 관리자 암호를 알고 있습니다.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **forwarders-absent.yml** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp forwarders-absent.yml ensure-presence-of-a-global-forwarder.yml
```

4. 편집할 수 있도록 **ensure-presence-of-a-global-forwarder.yml** 파일을 엽니다.

5. 다음 변수를 설정하여 파일을 조정합니다.

   a. **IdM DNS에 글로벌 전달자가 있는지 확인하기 위해 플레이북의 name 변수를 Playbook로** 변경합니다.

   b. **tasks** 섹션에서 작업의 **이름을** 변경하여 **DNS 글로벌 전달자가 포트 53의 7.7.9.9 및 2001:db8::1:0으로 있는지 확인합니다**.

   c. **ipadnsconfig** 부분의 **forwarders** 섹션에서 다음을 수행합니다.

      i. 첫 번째 **ip_address** 값을 global forwarder: **7.7.9.9** 의 IPv4 주소로 변경합니다.

      ii. 두 번째 **ip_address** 값을 global forwarder: **2001:db8::1:0** 의 IPv6 주소로 변경합니다.

      iii. **포트** 값이 **53** 으로 설정되어 있는지 확인합니다.

   d. **state** 를 **present** 로 변경합니다.
   이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to ensure the presence of a global forwarder in IdM DNS
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure the presence of a DNS global forwarder to 7.7.9.9 and 2001:db8::1:0 on port
53
    ipadnsconfig:
      forwarders:
        - ip_address: 7.7.9.9
        - ip_address: 2001:db8::1:0
          port: 53
      state: present
```

6. 파일을 저장합니다.

7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file ensure-presence-of-a-global-forwarder.yml
```

추가 리소스

- **/usr/share/doc/ansible-freeipa/** 디렉토리의 **README-dnsconfig.md** 파일을 참조하십시오.

1.3. ANSIBLE을 사용하여 IDM에 DNS 글로벌 전달자가 없는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS 글로벌 전달자가 없는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 인터넷 프로토콜(IP) v4 주소가 **8.8.6.6** 이고 포트 **53** 에서 **2001:4860:4860::8800** 의 IP v6 주소가 있는 DNS 글로벌 전달자가 없는지 확인합니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.
- IdM 관리자 암호를 알고 있습니다.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **forwarders-absent.yml** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp forwarders-absent.yml ensure-absence-of-a-global-forwarder.yml
```

4. 편집할 수 있도록 **ensure-absence-of-a-global-forwarder.yml** 파일을 엽니다.
5. 다음 변수를 설정하여 파일을 조정합니다.
   a. **IdM DNS에 글로벌 전달자가 없도록 플레이북의 name 변수를 Playbook로** 변경합니다.
   b. **tasks** 섹션에서 작업의 **이름을** 변경하여 **DNS 글로벌 전달자가 8.8.6.6 및 2001:4860:4860:4860::8800이 없는 것을 확인하도록 합니다**.
   c. **ipadnsconfig** 부분의 **forwarders** 섹션에서 다음을 수행합니다.
      i. 첫 번째 **ip_address** 값을 글로벌 forwarder의 IPv4 주소로 변경합니다. **8.8.6.6**.
      ii. 두 번째 **ip_address** 값을 global forwarder의 IPv6 주소로 변경합니다. **2001:4860:4860::8800**.
      iii. **포트** 값이 **53** 으로 설정되어 있는지 확인합니다.
   d. **작업** 변수를 **member** 로 설정합니다.
   e. **상태가 absent** 로 설정되어 있는지 확인합니다.

   이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to ensure the absence of a global forwarder in IdM DNS
  hosts: ipaserver
```

```
  become: true

  tasks:
  - name: Ensure the absence of a DNS global forwarder to 8.8.6.6 and
2001:4860:4860::8800 on port 53
    ipadnsconfig:
      forwarders:
        - ip_address: 8.8.6.6
        - ip_address: 2001:4860:4860::8800
          port: 53
      action: member
      state: absent
```

중요

action: member 를 사용하지 않고 플레이북에서 **state: absent** 옵션만 사용하면 플레이북이 실패합니다.

6. 파일을 저장합니다.
7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file ensure-absence-of-a-global-forwarder.yml
```

추가 리소스

- **/usr/share/doc/ansible-freeipa/** 디렉토리의 **README-dnsconfig.md** 파일
- ipadnsconfig ansible-freeipa 모듈의 **action: member** 옵션

1.4. IPADNSCONFIG ANSIBLE-FREEIPA 모듈의 ACTION: MEMBER 옵션

ansible-freeipa ipadnsconfig 모듈을 사용하여 IdM(Identity Management)에서 글로벌 전달자를 제외하려면 **state: absent** 옵션 외에 **action: member** 옵션을 사용해야 합니다. **action: member** 를 사용하지 않고 플레이북에서 **state: absent** 만 사용하면 플레이북이 실패합니다. 결과적으로 모든 글로벌 전달자를 제거하려면 플레이북에서 모두 개별적으로 지정해야 합니다. 반면 **state: 현재** 옵션에는 **action: member** 가 필요하지 않습니다.

다음 표에서 는 action: member 옵션의 올바른 사용을 보여주는 DNS 글로벌 전달자를 추가 및 제거하는 구성 예제를 제공합니다. 표는 각 줄에 표시됩니다.

- 플레이북을 실행하기 전에 구성된 글로벌 전달자
- 플레이북의 발췌 내용
- 플레이북을 실행한 후 구성된 글로벌 전달자

표 1.1. 글로벌 전달자의 ipadnsconfig 관리

이전 버전과의 전달자	플레이북 발췌 내용	이후의 forwarders

이전 버전과의 전달자	플레이북 발췌 내용	이후의 forwarders
8.8.6.6	[...] tasks: - name: Ensure the presence of DNS global forwarder 8.8.6.7 ipadnsconfig: forwarders: - ip_address: 8.8.6.7 state: present	8.8.6.7
8.8.6.6	[...] tasks: - name: Ensure the presence of DNS global forwarder 8.8.6.7 ipadnsconfig: forwarders: - ip_address: 8.8.6.7 action: member state: present	8.8.6.6, 8.8.6.7
8.8.6.6, 8.8.6.7	[...] tasks: - name: Ensure the absence of DNS global forwarder 8.8.6.7 ipadnsconfig: forwarders: - ip_address: 8.8.6.7 state: absent	플레이북을 실행하면 오류가 발생합니다. 원래 설정 - 8.8.6.6, 8.8.6.7은 변경되지 않습니다.
8.8.6.6, 8.8.6.7	[...] tasks: - name: Ensure the absence of DNS global forwarder 8.8.6.7 ipadnsconfig: forwarders: - ip_address: 8.8.6.7 action: member state: absent	8.8.6.6

1.5. IDM의 DNS 전달 정책

IdM은 **첫** 번째 및 **유일한** 표준 BIND 전달 정책은 물론 **none** IdM 관련 전달 정책을 지원합니다.

forward first *(기본값)*

IdM BIND 서비스는 DNS 쿼리를 구성된 전달자에게 전달합니다. 서버 오류 또는 시간 초과로 인해 쿼리가 실패하면 BIND는 인터넷의 서버를 사용하여 재귀 해결으로 돌아갑니다. **첫 번째** 정책은 기본 정책이며 DNS 트래픽을 최적화하는 데 적합합니다.

forward only

IdM BIND 서비스는 DNS 쿼리를 구성된 전달자에게 전달합니다. 서버 오류 또는 시간 초과로 인해 쿼리가 실패하면 BIND에서 클라이언트에 오류를 반환합니다. 정 **방향 전용** 정책은 분할 DNS 구성이 있는 환경에 권장됩니다.

none *(forwarding disabled)*

DNS 쿼리는 **none** 전달 정책으로 전달되지 않습니다. 전달 비활성화는 글로벌 전달 구성에 대한 영역별 재정의로만 유용합니다. 이 옵션은 BIND 구성에서 빈 전달자 목록을 지정하는 것과 동일한 IdM입니다.

참고

전달을 사용하여 IdM의 데이터를 다른 DNS 서버의 데이터와 결합할 수 없습니다. IdM DNS에서 기본 영역의 특정 하위 영역에 대한 쿼리만 전달할 수 있습니다.

기본적으로 BIND 서비스는 쿼리된 DNS 이름이 IdM 서버에 권한이 있는 영역에 속하는 경우 쿼리를 다른 서버로 전달하지 않습니다. 이러한 상황에서 IdM 데이터베이스에서 쿼리된 DNS 이름을 찾을 수 없으면 **NXDOMAIN** 응답이 반환됩니다. 포워딩은 사용되지 않습니다.

예 1.1. 예를 들면 Scenario

IdM 서버는 **test.example에 대한 권한이 있습니다.** DNS 영역. BIND는 **192.0.2.254** IP 주소가 있는 DNS 서버로 쿼리를 전달하도록 구성됩니다.

클라이언트가 **존재하지 않는.test.example**에 대한 쿼리를 전송하는 경우 DNS 이름인 BIND에서는 IdM 서버가 **test.example.** 영역에 대한 권한이 있고 쿼리를 **192.0.2.254.** 서버로 전달하지 않음을 탐지합니다. 결과적으로 DNS 클라이언트는 **NXDomain** 오류 메시지를 수신하여 사용자에게 쿼리된 도메인이 존재하지 않음을 알립니다.

1.6. ANSIBLE 플레이북을 사용하여 IDM DNS 글로벌 구성에 첫 번째 정책이 설정되어 있는지 확인합니다.

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM DNS의 글로벌 전달 정책이 **먼저** 전달되도록 설정하는 방법을 설명합니다.

전달 **첫 번째** DNS 전달 정책을 사용하는 경우 DNS 쿼리가 구성된 전달자에게 전달됩니다. 서버 오류 또는 시간 초과로 인해 쿼리가 실패하면 BIND는 인터넷의 서버를 사용하여 재귀 해결으로 돌아갑니다. forward first policy는 기본 정책입니다. 이는 트래픽 최적화에 적합합니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다.
- IdM 관리자 암호를 알고 있습니다.
- IdM 환경에는 통합 DNS 서버가 포함되어 있습니다.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **set-configuration.yml** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp set-configuration.yml set-forward-policy-to-first.yml
```

4. 편집을 위해 **set-forward-policy-to-first.yml** 파일을 엽니다.

5. **ipadnsconfig** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.

   - **ipaadmin_password** 변수를 IdM 관리자 암호로 설정합니다.
   - **forward_policy** 변수를 **먼저** 로 설정합니다.
     관련이 없는 원래 플레이북의 다른 모든 행을 삭제합니다. 이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to set global forwarding policy to first
  hosts: ipaserver
  become: true

  tasks:
  - name: Set global forwarding policy to first.
    ipadnsconfig:
      ipaadmin_password: Secret123
      forward_policy: first
```

6. 파일을 저장합니다.

7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file set-forward-policy-to-first.yml
```

추가 리소스

- IdM의 DNS 전달 정책을 참조하십시오.
- **/usr/share/doc/ansible-freeipa/** 디렉토리의 **README-dnsconfig.md** 파일을 참조하십시오.
- 자세한 샘플 플레이북은 **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리를 참조하십시오.

1.7. ANSIBLE 플레이북을 사용하여 IDM DNS에서 글로벌 전달자가 비활성화되었는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM DNS에서 글로벌 전달자가 비활성화되었는지 확인하는 방법을 설명합니다. disable는 **forward_policy** 변수를 **none** 으로 설정하여 수행됩니다.

글로벌 전달자를 비활성화하면 DNS 쿼리가 전달되지 않습니다. 전달 비활성화는 글로벌 전달 구성에 대한 영역별 재정의로만 유용합니다. 이 옵션은 BIND 구성에서 빈 전달자 목록을 지정하는 것과 동일한 IdM입니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다.
- IdM 관리자 암호를 알고 있습니다.
- IdM 환경에는 통합 DNS 서버가 포함되어 있습니다.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **disable-global-forwarders.yml** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp disable-global-forwarders.yml disable-global-forwarders-copy.yml
```

4. 편집할 **disable-global-forwarders-copy.yml** 파일을 엽니다.
5. **ipadnsconfig** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.
   - **ipaadmin_password** 변수를 IdM 관리자 암호로 설정합니다.
   - **forward_policy** 변수를 **none** 으로 설정합니다.
     이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to disable global DNS forwarders
  hosts: ipaserver
  become: true

  tasks:
  - name: Disable global forwarders.
    ipadnsconfig:
      ipaadmin_password: Secret123
      forward_policy: none
```

6. 파일을 저장합니다.

7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file disable-global-forwarders-copy.yml
```

추가 리소스

- IdM의 DNS 전달 정책을 참조하십시오.
- **/usr/share/doc/ansible-freeipa/** 디렉토리의 **README-dnsconfig.md** 파일을 참조하십시오.
- **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리에서 더 많은 샘플 플레이북을 참조하십시오.

1.8. ANSIBLE 플레이북을 사용하여 IDM DNS에서 정방향 및 역방향 조회 영역의 동기화가 비활성화되었는지 확인합니다.

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM DNS에서 정방향 및 역방향 조회 영역이 동기화되지 않도록 하는 방법을 설명합니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다.
- IdM 관리자 암호를 알고 있습니다.
- IdM 환경에는 통합 DNS 서버가 포함되어 있습니다.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **disallow-reverse-sync.yml** Ansible 플레이북 파일을 복사합니다. 예를 들어 다음과 같습니다.

```
$ cp disallow-reverse-sync.yml disallow-reverse-sync-copy.yml
```

4. 편집을 위해 **disallow-reverse-sync-copy.yml** 파일을 엽니다.
5. **ipadnsconfig** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.
   - **ipaadmin_password** 변수를 IdM 관리자 암호로 설정합니다.
   - **allow_sync_ptr** 변수를 **no** 로 설정합니다.
     이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
```

```
- name: Playbook to disallow reverse record synchronization
  hosts: ipaserver
  become: true

  tasks:
  - name: Disallow reverse record synchronization.
    ipadnsconfig:
      ipaadmin_password: Secret123
      allow_sync_ptr: no
```

6. 파일을 저장합니다.

7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file disallow-reverse-sync-copy.yml
```

추가 리소스

- **/usr/share/doc/ansible-freeipa/** 디렉토리의 **README-dnsconfig.md** 파일을 참조하십시오.
- 자세한 샘플 플레이북은 **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리를 참조하십시오.

2장. IDM의 DNS 영역 관리

IdM(Identity Management) 관리자는 IdM DNS 영역의 작동 방식을 관리할 수 있습니다. 이 장에서는 다음 주제 및 절차에 대해 설명합니다.

- IdM에서 지원되는 DNS 영역 유형
  - IdM 웹 UI를 사용하여 기본 IdM DNS 영역을 추가하는 방법
  - IdM CLI를 사용하여 기본 IdM DNS 영역을 추가하는 방법
  - IdM 웹 UI를 사용하여 기본 IdM DNS 영역을 제거하는 방법
  - IdM CLI를 사용하여 기본 IdM DNS 영역을 제거하는 방법
- IdM에서 구성할 수 있는 DNS 속성
  - IdM 웹 UI에서 이러한 속성을 구성하는 방법
  - IdM CLI에서 이러한 속성을 구성하는 방법
- IdM에서 영역 전송 작업 방법
  - IdM 웹 UI에서 영역 전송을 허용하는 방법
  - IdM CLI에서 영역 전송을 허용하는 방법

사전 요구 사항

- DNS 서비스가 IdM 서버에 설치되어 있습니다. 통합 DNS를 사용하여 IdM 서버를 설치하는 방법에 대한 자세한 내용은 다음 링크 중 하나를 참조하십시오.
  - IdM 서버 설치: 통합 DNS와 함께 통합 CA를 루트 CA로 설치
  - IdM 서버 설치: 루트 CA로 외부 CA를 사용하는 통합 DNS 사용
  - IdM 서버 설치: CA없이 통합 DNS 사용

2.1. 지원되는 DNS 영역 유형

IdM(Identity Management)은 *기본* 및 *전달* 영역의 두 가지 유형의 DNS 영역을 지원합니다. 이 섹션에서는 이러한 두 가지 유형의 영역에 대해 설명하고 DNS 전달을 위한 시나리오 예제를 포함합니다.

참고

이 가이드에서는 Microsoft Windows DNS에 사용되는 용어와 다른 영역 유형에 대해 BIND 용어를 사용합니다. BIND의 기본 영역은 Microsoft Windows DNS의 정방향 *조회 영역* 및 *역방향 조회 영역*과 동일한 용도로 사용됩니다. BIND의 전달 영역은 Microsoft Windows DNS *의 조건부 전달자* 와 동일한 용도로 사용됩니다.

기본 DNS 영역

기본 DNS 영역에는 권한 있는 DNS 데이터가 포함되어 있으며 동적 DNS 업데이트를 허용할 수 있습니다. 이 동작은 표준 BIND 구성의 **유형 master** 설정과 동일합니다. **ipa dnszone-*** 명령을 사용하여 기본 영역을 관리할 수 있습니다.

표준 DNS 규칙에 따라 모든 기본 영역에는 **권한 시작** (SOA) 및 **네임서버** (NS) 레코드가 포함되어야 합니다. IdM은 DNS 영역을 생성할 때 이러한 레코드를 자동으로 생성하지만 적절한 위임을 생성하려면 NS 레코드를 상위 영역에 수동으로 복사해야 합니다.

표준 BIND 동작에 따라 서버에 권한이 없는 이름에 대한 쿼리가 다른 DNS 서버로 전달됩니다. 이러한 DNS 서버는 전달자라고 하며 쿼리에 대해 권한이 없을 수도 있습니다.

> **예 2.1. DNS 전달 시나리오 예**
>
> IdM 서버에는 **test.example.** 기본 영역이 포함되어 있습니다. 이 영역에는 **sub.test.example.** 이름에 대한 NS 위임 레코드가 포함되어 있습니다. 또한 **test.example.** 영역은 **sub.text.example** 하위 영역에 대한 **192.0.2.254** forwarder IP 주소로 구성됩니다.
>
> **존재하지 않는** 이름을 쿼리하는 클라이언트는 **NXDomain** 응답을 수신하며, IdM 서버가 이 이름에 대해 권한이 있기 때문에 전달이 수행되지 않습니다.
>
> 반면, IdM 서버가 이 이름에 대해 권한이 없으므로 **host1.sub.test.example.** 이름에 대한 쿼리가 구성된 forwarder **192.0.2.254** 로 전달됩니다.

DNS 영역 전달

IdM의 관점에서는 전달 DNS 영역에 권한 있는 데이터가 포함되어 있지 않습니다. 사실, 앞으로 "zone"은 일반적으로 다음 두 가지 정보만 포함합니다.

- 도메인 이름
- 도메인과 연결된 DNS 서버의 IP 주소

정의된 도메인에 속하는 이름에 대한 모든 쿼리는 지정된 IP 주소로 전달됩니다. 이 동작은 표준 BIND 구성의 **type forward** 설정과 동일합니다. **ipa dnsforwardzone-*** 명령을 사용하여 전달 영역을 관리할 수 있습니다.

전달 DNS 영역은 IdM-Active Directory (AD) 신뢰의 컨텍스트에서 특히 유용합니다. IdM DNS 서버에서 **idm.example.com** 영역에 대한 권한이 있고 AD DNS 서버에 **ad.example.com** 영역에 대한 권한이 있는 경우 **ad.example.com** 은 **idm.example.com** 기본 영역의 DNS 전달 영역입니다. 즉, **somehost.ad.example.com** 의 IP 주소에 대한 쿼리가 IdM 클라이언트에서 제공되면 쿼리가 **ad.example.com** IdM DNS 전달 영역에 지정된 AD 도메인 컨트롤러로 전달됩니다.

2.2. IDM 웹 UI에서 기본 DNS 영역 추가

이 섹션에서는 IdM(Identity Management) 웹 UI를 사용하여 기본 DNS 영역을 추가하는 방법을 설명합니다.

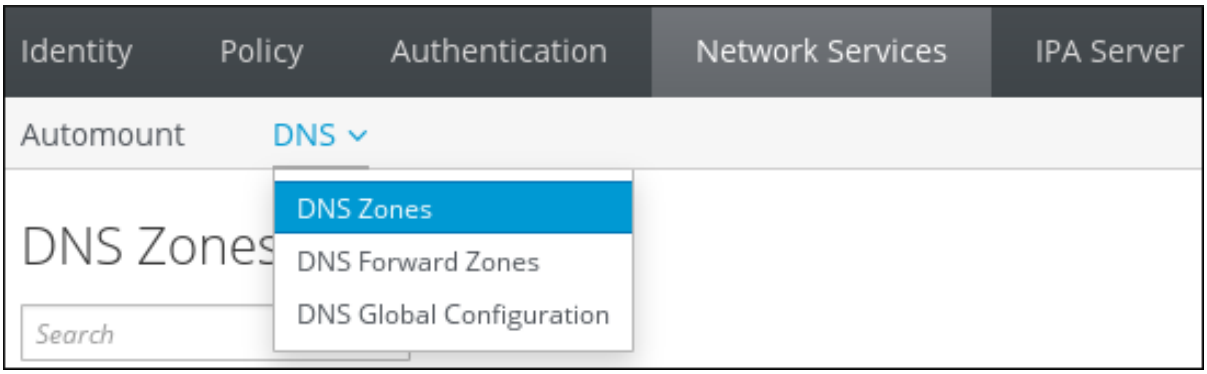
사전 요구 사항

- IdM 관리자로 로그인되어 있습니다.

절차

1. IdM 웹 UI에서 **네트워크 서비스 → DNS → DNS 영역**을 클릭합니다.

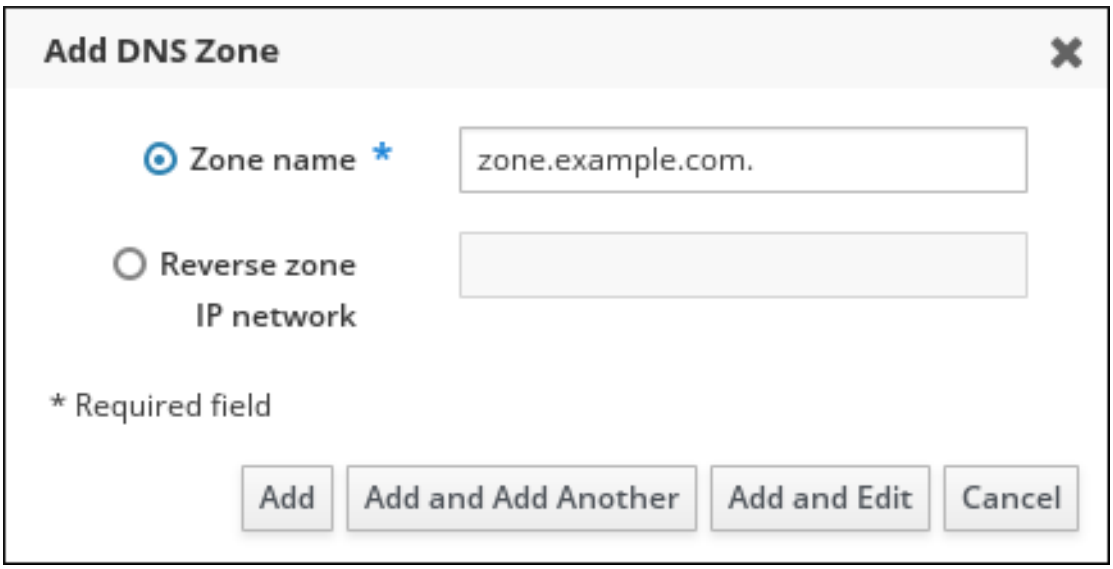
그림 2.1. IdM DNS 기본 영역 관리



2. 모든 영역 목록 상단에서 **Add** (추가)를 클릭합니다.

3. 영역 이름을 제공합니다.

그림 2.2. 새로운 IdM 기본 영역 입력



4. **추가**를 클릭합니다.

2.3. IDM CLI에서 기본 DNS 영역 추가

이 섹션에서는 IdM(Identity Management) CLI(명령줄 인터페이스)에 기본 DNS 영역을 추가하는 방법을 설명합니다.

사전 요구 사항

- IdM 관리자로 로그인되어 있습니다.

절차

- **ipa dnszone-add** 명령은 DNS 도메인에 새 영역을 추가합니다. 새 영역을 추가하려면 새 하위 도메인의 이름을 지정해야 합니다. 명령을 사용하여 하위 도메인 이름을 직접 전달할 수 있습니다.

```
$ ipa dnszone-add newzone.idm.example.com
```

ipa dnszone-add 에 이름을 전달하지 않으면 스크립트에서 자동으로 메시지를 표시합니다.

추가 리소스

- **ipa dnszone-add --help** 를 참조하십시오.

2.4. IDM 웹 UI에서 기본 DNS 영역 제거

이 섹션에서는 IdM 웹 UI를 사용하여 IdM(Identity Management)에서 기본 DNS 영역을 제거하는 방법을 설명합니다.

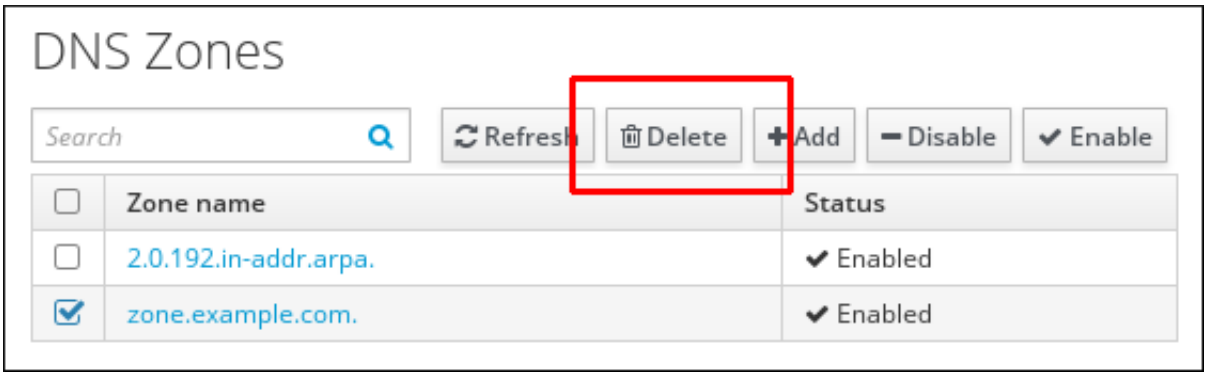
사전 요구 사항

- IdM 관리자로 로그인되어 있습니다.

절차

1. IdM 웹 UI에서 **네트워크 서비스 → DNS → DNS 영역**을 클릭합니다.
2. 영역 이름으로 확인란을 선택하고 **삭제** 를 클릭합니다.

그림 2.3. 기본 DNS 영역 제거



3. **Remove DNS zones** (DNS 영역 제거) 대화 상자에서 선택한 영역을 삭제하도록 확인합니다.

2.5. IDM CLI에서 기본 DNS 영역 제거

이 섹션에서는 IdM CLI(명령줄 인터페이스)를 사용하여 IdM(Identity Management)에서 기본 DNS 영역을 제거하는 방법을 설명합니다.

사전 요구 사항

- IdM 관리자로 로그인되어 있습니다.

절차

- 기본 DNS 영역을 제거하려면 **ipa dnszone-del** 명령을 입력하고 제거할 영역의 이름을 입력합니다. 예를 들어 다음과 같습니다.

```
$ ipa dnszone-del idm.example.com
```

2.6. DNS 구성 우선 순위

다음 수준에서 많은 DNS 구성 옵션을 구성할 수 있습니다. 각 수준은 다른 우선 순위를 갖습니다.

영역별 구성

IdM에 정의된 특정 영역에 고유한 구성 수준은 우선 순위가 높습니다. **ipa dnszone-*** 및 **ipa dnsforwardzone-*** 명령을 사용하여 영역별 구성을 관리할 수 있습니다.

서버 단위 구성

IdM 서버를 설치하는 동안 서버별 전달자를 정의해야 합니다. **ipa dnsserver-*** 명령을 사용하여 서버별 전달자를 관리할 수 있습니다. 복제본을 설치할 때 서버별 전달자를 설정하지 않으려면 **--no-forwarder** 옵션을 사용할 수 있습니다.

글로벌 DNS 구성

영역별 구성이 정의되지 않은 경우 IdM은 LDAP에 저장된 글로벌 DNS 구성을 사용합니다. **ipa dnsconfig-*** 명령을 사용하여 글로벌 DNS 구성을 관리할 수 있습니다. 글로벌 DNS 구성에 정의된 설정은 모든 IdM DNS 서버에 적용됩니다.

/etc/named.conf의 구성

각 IdM DNS 서버의 **/etc/named.conf** 파일에 정의된 구성에 우선순위가 가장 낮습니다. 이는 각 서버에 고유하며 수동으로 편집해야 합니다.
/etc/named.conf 파일은 일반적으로 로컬 DNS 캐시로 DNS 전달을 지정하는 데만 사용됩니다. 기타 옵션은 위에 언급된 영역별 및 글로벌 DNS 구성에 대한 명령을 사용하여 관리됩니다.

동시에 여러 수준에서 DNS 옵션을 구성할 수 있습니다. 이러한 경우 우선 순위가 가장 높은 구성이 하위 수준에서 정의된 구성보다 우선합니다.

추가 리소스

- LDAP 의 서버 구성의 우선 순위 설정 순서

2.7. 기본 IDM DNS 영역의 구성 속성

IdM(Identity Management)은 새로 고침 기간, 전송 설정 또는 캐시 설정과 같은 특정 기본 구성으로 새 영역을 생성합니다. IdM DNS 영역 속성에서 다음 옵션 중 하나를 사용하여 수정할 수 있는 기본 영역 구성의 속성을 찾을 수 있습니다.

- CLI(명령줄 인터페이스)의 **dnszone-mod** 명령. 자세한 내용은 IdM CLI에서 기본 DNS 영역의 구성 편집 을 참조하십시오.
- IdM 웹 UI. 자세한 내용은 IdM 웹 UI에서 기본 DNS 영역의 구성 편집 을 참조하십시오.
- **ipadnszone** 모듈을 사용하는 Ansible 플레이북입니다. 자세한 내용은 IdM의 DNS 영역 관리를 참조하십시오.

영역에 대한 실제 정보를 설정하는 것과 함께 설정은 DNS 서버가 *권한 시작* (SOA) 레코드 항목을 처리하는 방법과 DNS 이름 서버에서 해당 레코드를 업데이트하는 방법을 정의합니다.

표 2.1. IdM DNS 영역 속성

속성	명령줄 옵션	설명
권한 있는 이름 서버	**--name-server**	SOA MNAME이라고도 하는 기본 DNS 이름 서버의 도메인 이름을 설정합니다. 기본적으로 각 IdM 서버는 SOA MNAME 필드에 자신을 알립니다. 결과적으로 **--name-server** 를 사용하여 LDAP에 저장된 값이 무시됩니다.

속성	명령줄 옵션	설명
관리자 이메일 주소	**--admin-email**	영역 관리자에게 사용할 이메일 주소를 설정합니다. 이 기본값은 호스트의 root 계정입니다.
SOA 직렬	**--serial**	SOA 레코드의 일련 번호를 설정합니다. IdM은 버전 번호를 자동으로 설정하며 사용자는 수정할 필요가 없습니다.
SOA 새로 고침	**--refresh**	기본 DNS 서버에서 업데이트를 요청하기 전에 대기할 간격(초)을 설정합니다.
SOA 재시도	**--retry**	실패한 새로 고침 작업을 다시 시도하기 전에 대기하는 시간(초)을 설정합니다.
SOA 만료	**--expire**	작업을 종료하기 전에 보조 DNS 서버에서 새로 고침 업데이트를 수행하려고 하는 시간(초)을 설정합니다.
SOA 최소	**--minimum**	RFC 2308 에 따라 음수 캐시의 시간(TTL) 값을 초 단위로 설정합니다.
SOA 지원 시간	**--ttl**	영역 apex에서 레코드의 TTL을 초 단위로 설정합니다. **example.com** 영역의 경우 example.com이라는 이름의 모든 레코드(A, NS 또는 SOA)가 구성되어 있지만 **test.example.com** 과 같은 다른 도메인 이름은 영향을 받지 않습니다.
기본 라이브 시간	**--default-ttl**	이전에 개별 TTL 값이 설정되지 않은 영역의 모든 값에 대한 음수 캐싱을 위해 기본 시간을 TTL(TTL)으로 설정합니다. 변경 사항을 적용하려면 모든 IdM DNS 서버에서 **named-pkcs11** 서비스를 다시 시작해야 합니다.
BIND 업데이트 정책	**--update-policy**	DNS 영역의 클라이언트에 허용되는 권한을 설정합니다.
동적 업데이트	**--dynamic-update**=TRUE|FALSE	클라이언트의 DNS 레코드에 대한 동적 업데이트를 활성화합니다. false로 설정하면 IdM 클라이언트 시스템에서 IP 주소를 추가하거나 업데이트할 수 없습니다.
전송 허용	**--allow-transfer**=*string*	지정된 영역을 부분(;)으로 구분하여 전송할 수 있는 IP 주소 또는 네트워크 이름 목록을 제공합니다. 영역 전송은 기본적으로 비활성화되어 있습니다. 기본값 **--allow-transfer** 값은 **none** 입니다.
쿼리 허용	**--allow-query**	Semicolons (;)로 구분된 DNS 쿼리를 실행할 수 있는 IP 주소 또는 네트워크 이름 목록을 제공합니다.
PTR 동기화 허용	**--allow-sync-ptr**=1|0	영역의 A 또는 AAAA 레코드(전달 레코드)가 PTR 레코드(reverse) 레코드와 자동으로 동기화되는지 여부를 설정합니다.

속성	명령줄 옵션	설명
영역 전달자	**--forwarder**=*IP_address*	DNS 영역을 위해 특별히 구성된 전달자를 지정합니다. IdM 도메인에 사용된 글로벌 전달자와는 별개입니다. 여러 전달자를 지정하려면 옵션을 여러 번 사용합니다.
forward policy	**--forward-policy**=none|only|first	전달 정책을 지정합니다. 지원되는 정책에 대한 자세한 내용은 IdM의 DNS 전달 정책을 참조하십시오.

2.8. IDM 웹 UI에서 기본 DNS 영역 구성 편집

이 섹션에서는 IdM 웹 UI를 사용하여 기본 IdM(Identity Management) DNS의 구성 속성을 편집하는 방법을 설명합니다.

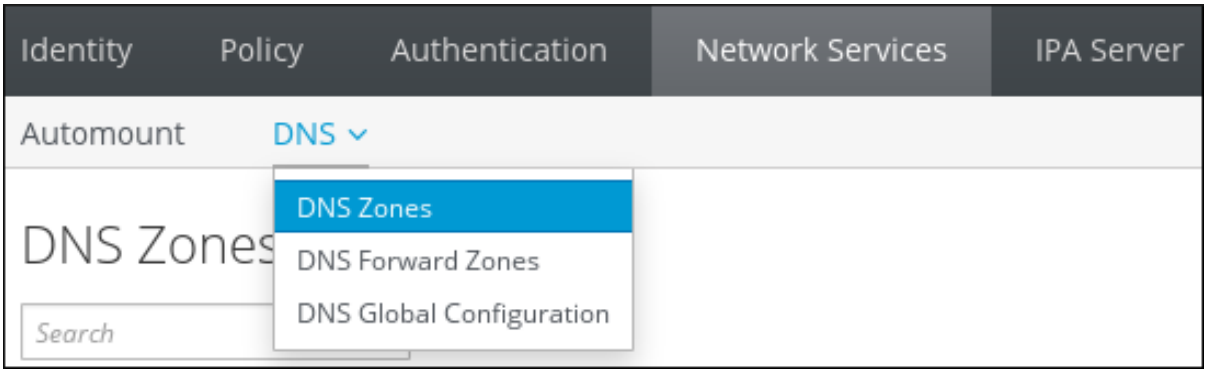
사전 요구 사항

- IdM 관리자로 로그인되어 있습니다.

절차

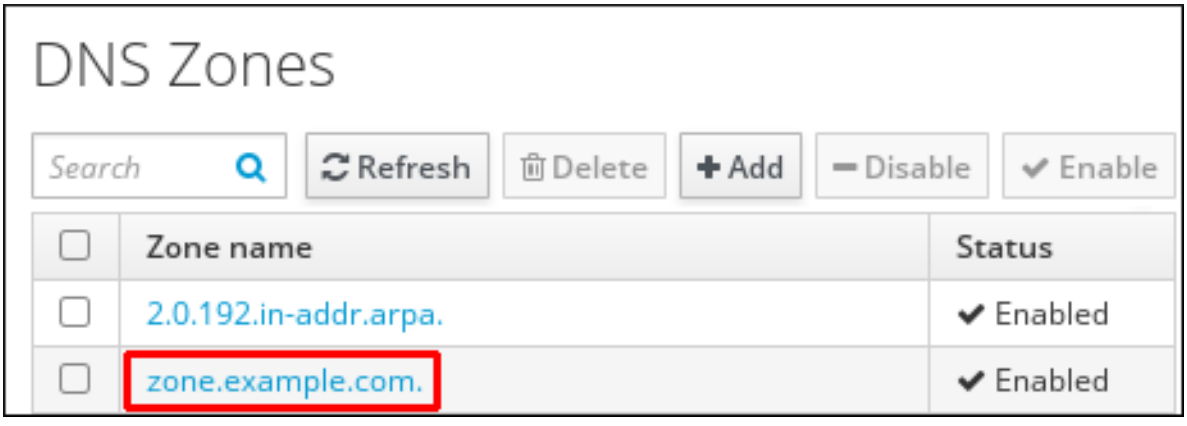
1. IdM 웹 UI에서 **네트워크 서비스 → DNS → DNS 영역**을 클릭합니다.

그림 2.4. DNS 기본 영역 관리



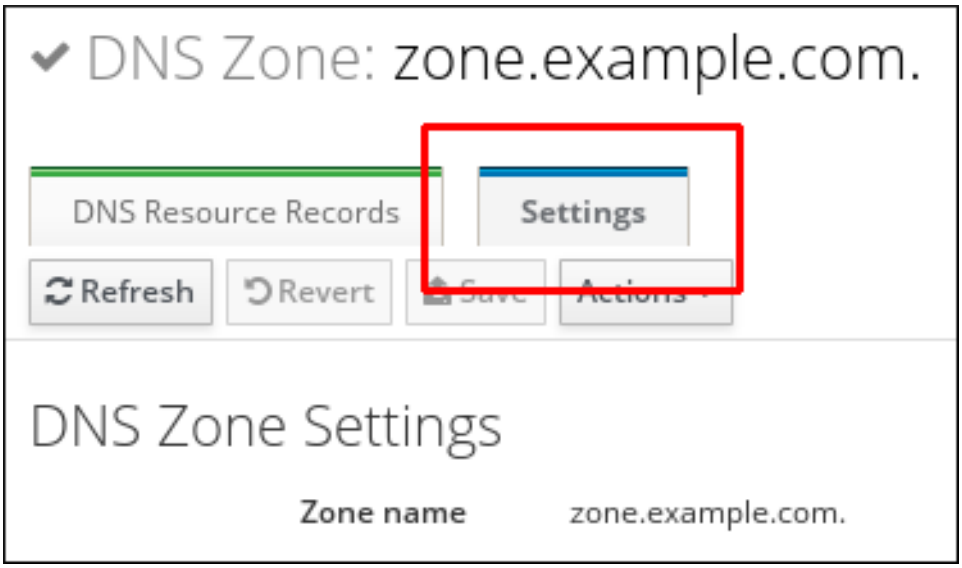
2. **DNS 영역** 섹션의 모든 영역 목록에서 영역 이름을 클릭하여 DNS 영역 페이지를 엽니다.

그림 2.5. 기본 영역 편집



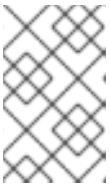
3. **설정**을 클릭합니다.

그림 2.6. 기본 영역 편집 페이지의 설정 탭



4. 필요에 따라 영역 구성을 변경합니다.
사용 가능한 설정에 대한 자세한 내용은 IdM DNS 영역 특성을 참조하십시오.

5. **저장을** 클릭하여 새 구성을 확인합니다.



참고

영역의 기본 시간을 실시간(TTL)으로 변경하는 경우 모든 IdM DNS 서버에서 **named-pkcs11** 서비스를 다시 시작하여 변경 사항을 적용합니다. 다른 모든 설정은 즉시 활성화됩니다.

2.9. IDM CLI에서 기본 DNS 영역 구성 편집

이 섹션에서는 IdM(Identity Management) 명령줄 인터페이스(CLI)를 사용하여 기본 DNS 영역의 구성을 편집하는 방법을 설명합니다.

사전 요구 사항

- IdM 관리자로 로그인되어 있습니다.

절차

- 기존 기본 DNS 영역을 수정하려면 **ipa dnszone-mod** 명령을 사용합니다. 예를 들어 실패한 새로 고침 작업을 1800초로 다시 시도하기 전에 대기할 시간을 설정하려면 다음을 수행합니다.

```
$ ipa dnszone-mod --retry 1800
```

사용 가능한 설정 및 해당 CLI 옵션에 대한 자세한 내용은 IdM DNS 영역 속성 을 참조하십시오.

특정 설정에 수정 중인 DNS 영역 항목에 값이 없는 경우 **ipa dnszone-mod** 명령에서 값을 추가합니다. 설정에 값이 없는 경우 명령에서 현재 값을 지정된 값으로 덮어씁니다.

참고

영역의 기본 시간을 실시간(TTL)으로 변경하는 경우 모든 IdM DNS 서버에서 **named-pkcs11** 서비스를 다시 시작하여 변경 사항을 적용합니다. 다른 모든 설정은 즉시 활성화됩니다.

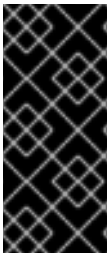
추가 리소스

- **ipa dnszone-mod --help** 를 참조하십시오.

2.10. IDM의 영역 전송

이 섹션에서는 DNS 통합 DNS가 있는 IdM(Identity Management) 배포에서 영역 전송이 작동하는 방법을 설명합니다.

이름 서버는 영역에 대한 권한 있는 데이터를 유지합니다. 영역 A DNS 영역에 대한 권한이 있는 DNS 서버의 영역을 변경하는 경우, *영역 A* 외부에 있는 IdM DNS 도메인의 다른 이름 서버 간에 변경 사항을 배포해야 합니다. *영역 전송 은 한 이름 서버의 모든 리소스 레코드를 다른 이름으로 복사합니다.*



중요

IdM 통합 DNS는 다른 서버에서 동시에 작성할 수 있습니다. IdM 영역의 권한 시작(SOA) 일련 번호는 개별 IdM DNS 서버 간에 동기화되지 않습니다. 이러한 이유로 to-be-transferred 영역 외부에 DNS 서버를 구성하여 to-be-transferred 영역 내에서 하나의 특정 DNS 서버만 사용합니다. 따라서 동기화되지 않은 SOA 일련 번호로 인한 영역 전송 오류를 방지할 수 있습니다.

IdM은 RFC 5936 (AXFR) 및 RFC jenkinsfile(IXFR) 표준에 따라 영역 전송을 지원합니다.

추가 리소스

- *IdM 웹 UI에서 영역 전송 활성화를 참조하십시오.*
- *IdM CLI에서 영역 전송 활성화를 참조하십시오.*

2.11. IDM 웹 UI에서 영역 전송 활성화

이 섹션에서는 IdM 웹 UI를 사용하여 IdM(Identity Management)에서 영역 전송을 활성화하는 방법을 설명합니다.

사전 요구 사항

- *IdM 관리자로 로그인되어 있습니다.*

절차

1. *IdM 웹 UI에서 **네트워크 서비스 → DNS → DNS 영역을** 클릭합니다.*
2. ***설정을** 클릭합니다.*
3. ***Allow transfer** 에서 영역 레코드를 전송할 이름 서버를 지정합니다.*

그림 2.7. 영역 전송 활성화

4. *DNS 영역 페이지 상단에 있는* ***저장을*** *클릭하여 새 구성을 확인합니다.*

2.12. IDM CLI에서 영역 전송 활성화

이 섹션에서는 IdM CLI(명령줄 인터페이스)를 사용하여 IdM(Identity Management)에서 영역 전송을 활성화하는 방법을 설명합니다.

사전 요구 사항

- *IdM 관리자로 로그인되어 있습니다.*
- *보조 DNS 서버에 대한 루트 액세스 권한이 있습니다.*

절차

- ***BIND*** *서비스에서 영역 전송을 활성화하려면* ***ipa dnszone-mod*** *명령을 입력하고* ***--allow-transfer*** *ferferferer 옵션을 사용하여 영역 레코드를 전송할 to-be-transferred 영역 외부에 있는 이름 서버 목록을 지정합니다. 예를 들어 다음과 같습니다.*

```
$ ipa dnszone-mod --allow-transfer=192.0.2.1;198.51.100.1;203.0.113.1
idm.example.com
```

검증 단계

1. *영역 전송을 활성화할 DNS 서버 중 하나에 SSH를 수행합니다.*

```
$ ssh 192.0.2.1
```

2. ***dig*** *유틸리티와 같은 툴을 사용하여 IdM DNS 영역을 전송합니다.*

```
# dig @ipa-server zone_name AXFR
```

명령에서 오류를 반환하지 않으면 zone_name 에 대해 영역 전송을 성공적으로 활성화했습니다.

2.13. 추가 리소스

- *Ansible 플레이북을 사용하여 IdM DNS 영역 관리를 참조하십시오.*

3장. ANSIBLE 플레이북을 사용하여 IDM DNS 영역 관리

IdM(Identity Management) 관리자는 **ansible-freeipa** 패키지에서 사용할 수 있는 **dnszone** 모듈을 사용하여 IdM DNS 영역이 작동하는 방식을 관리할 수 있습니다. 이 장에서는 다음 주제 및 절차에 대해 설명합니다.

- IdM에서 지원되는 DNS 영역 유형
- IdM에서 구성할 수 있는 DNS 속성
- Ansible 플레이북을 사용하여 IdM DNS에 기본 영역을 생성하는 방법
- Ansible 플레이북을 사용하여 여러 변수가 있는 기본 IdM DNS 영역이 있는지 확인하는 방법
- Ansible 플레이북을 사용하여 IP 주소가 지정될 때 역방향 DNS 조회 영역이 있는지 확인하는 방법

사전 요구 사항

- DNS 서비스가 IdM 서버에 설치되어 있습니다. Red Hat Ansible Engine을 사용하여 통합 DNS로 IdM 서버를 설치하는 방법에 대한 자세한 내용은 Ansible 플레이북을 사용하여 ID 관리 서버 설치를 참조하십시오.

3.1. 지원되는 DNS 영역 유형

IdM(Identity Management)은 기본 및 전달 영역의 두 가지 유형의 DNS 영역을 지원합니다. 이 섹션에서는 이러한 두 가지 유형의 영역에 대해 설명하고 DNS 전달을 위한 시나리오 예제를 포함합니다.

참고

이 가이드에서는 Microsoft Windows DNS에 사용되는 용어와 다른 영역 유형에 대해 BIND 용어를 사용합니다. BIND의 기본 영역은 Microsoft Windows DNS의 정방향 조회 영역 및 역방향 조회 영역과 동일한 용도로 사용됩니다. BIND의 전달 영역은 Microsoft Windows DNS의 조건부 전달자 와 동일한 용도로 사용됩니다.

기본 DNS 영역

기본 DNS 영역에는 권한 있는 DNS 데이터가 포함되어 있으며 동적 DNS 업데이트를 허용할 수 있습니다. 이 동작은 표준 BIND 구성의 **유형 master** 설정과 동일합니다. **ipa dnszone-*** 명령을 사용하여 기본 영역을 관리할 수 있습니다.
표준 DNS 규칙에 따라 모든 기본 영역에는 **권한 시작** (SOA) 및 **네임서버** (NS) 레코드가 포함되어야 합니다. IdM은 DNS 영역을 생성할 때 이러한 레코드를 자동으로 생성하지만 적절한 위임을 생성하려면 NS 레코드를 상위 영역에 수동으로 복사해야 합니다.

표준 BIND 동작에 따라 서버에 권한이 없는 이름에 대한 쿼리가 다른 DNS 서버로 전달됩니다. 이러한 DNS 서버는 전달자라고 하며 쿼리에 대해 권한이 없을 수도 있습니다.

> **예 3.1. DNS 전달 시나리오 예**
>
> IdM 서버에는 **test.example.** 기본 영역이 포함되어 있습니다. 이 영역에는 **sub.test.example.** 이름에 대한 NS 위임 레코드가 포함되어 있습니다. 또한 **test.example.** 영역은 **sub.text.example** 하위 영역에 대한 **192.0.2.254** forwarder IP 주소로 구성됩니다.
>
> **존재하지 않는** 이름을 쿼리하는 클라이언트는 **NXDomain** 응답을 수신하며, IdM 서버가 이 이름에 대해 권한이 있기 때문에 전달이 수행되지 않습니다.

> 반면, IdM 서버가 이 이름에 대해 권한이 없으므로 **host1.sub.test.example.** 이름에 대한 쿼리가 구성된 forwarder **192.0.2.254** 로 전달됩니다.

DNS 영역 전달

IdM의 관점에서는 전달 DNS 영역에 권한 있는 데이터가 포함되어 있지 않습니다. 사실, 앞으로 "zone"은 일반적으로 다음 두 가지 정보만 포함합니다.

- 도메인 이름
- 도메인과 연결된 DNS 서버의 IP 주소

정의된 도메인에 속하는 이름에 대한 모든 쿼리는 지정된 IP 주소로 전달됩니다. 이 동작은 표준 BIND 구성의 **type forward** 설정과 동일합니다. **ipa dnsforwardzone-*** 명령을 사용하여 전달 영역을 관리할 수 있습니다.

전달 DNS 영역은 IdM-Active Directory (AD) 신뢰의 컨텍스트에서 특히 유용합니다. IdM DNS 서버에서 **idm.example.com** 영역에 대한 권한이 있고 AD DNS 서버에 **ad.example.com** 영역에 대한 권한이 있는 경우 **ad.example.com** 은 **idm.example.com** 기본 영역의 DNS 전달 영역입니다. 즉, **somehost.ad.example.com** 의 IP 주소에 대한 쿼리가 IdM 클라이언트에서 제공되면 쿼리가 **ad.example.com** IdM DNS 전달 영역에 지정된 AD 도메인 컨트롤러로 전달됩니다.

3.2. 기본 IDM DNS 영역의 구성 속성

IdM(Identity Management)은 새로 고침 기간, 전송 설정 또는 캐시 설정과 같은 특정 기본 구성으로 새 영역을 생성합니다. IdM DNS 영역 속성에서 다음 옵션 중 하나를 사용하여 수정할 수 있는 기본 영역 구성의 속성을 찾을 수 있습니다.

- CLI(명령줄 인터페이스)의 **dnszone-mod** 명령. 자세한 내용은 IdM CLI에서 기본 DNS 영역의 구성 편집 을 참조하십시오.
- IdM 웹 UI. 자세한 내용은 IdM 웹 UI에서 기본 DNS 영역의 구성 편집 을 참조하십시오.
- **ipadnszone** 모듈을 사용하는 Ansible 플레이북입니다. 자세한 내용은 IdM의 DNS 영역 관리를 참조하십시오.

영역에 대한 실제 정보를 설정하는 것과 함께 설정은 DNS 서버가 권한 시작 (SOA) 레코드 항목을 처리하는 방법과 DNS 이름 서버에서 해당 레코드를 업데이트하는 방법을 정의합니다.

표 3.1. IdM DNS 영역 속성

속성	Ansible-freeipa 변수	설명
권한 있는 이름 서버	**name_server**	SOA MNAME이라고도 하는 기본 DNS 이름 서버의 도메인 이름을 설정합니다. 기본적으로 각 IdM 서버는 SOA MNAME 필드에 자신을 알립니다. 결과적으로 **--name-server** 를 사용하여 LDAP에 저장된 값이 무시됩니다.
관리자 이메일 주소	**admin_email**	영역 관리자에게 사용할 이메일 주소를 설정합니다. 이 기본값은 호스트의 root 계정입니다.

속성	Ansible-freeipa 변수	설명
SOA 직렬	**serial**	SOA 레코드의 일련 번호를 설정합니다. IdM은 버전 번호를 자동으로 설정하며 사용자는 수정할 필요가 없습니다.
SOA 새로 고침	**새로 고침**	기본 DNS 서버에서 업데이트를 요청하기 전에 대기할 간격(초)을 설정합니다.
SOA 재시도	**Retry**	실패한 새로 고침 작업을 다시 시도하기 전에 대기하는 시간(초)을 설정합니다.
SOA 만료	**expire**	작업을 종료하기 전에 보조 DNS 서버에서 새로 고침 업데이트를 수행하려고 하는 시간(초)을 설정합니다.
SOA 최소	**minimum**	RFC 2308 에 따라 음수 캐시의 시간(TTL) 값을 초 단위로 설정합니다.
SOA 지원 시간	**ttl**	영역 apex에서 레코드의 TTL을 초 단위로 설정합니다. **example.com** 영역의 경우 example.com이라는 이름의 모든 레코드(A, NS 또는 SOA)가 구성되어 있지만 **test.example.com** 과 같은 다른 도메인 이름은 영향을 받지 않습니다.
기본 라이브 시간	**default_ttl**	이전에 개별 TTL 값이 설정되지 않은 영역의 모든 값에 대한 음수 캐싱을 위해 기본 시간을 TTL(TTL)으로 설정합니다. 변경 사항을 적용하려면 모든 IdM DNS 서버에서 **named-pkcs11** 서비스를 다시 시작해야 합니다.
BIND 업데이트 정책	**update_policy**	DNS 영역의 클라이언트에 허용되는 권한을 설정합니다.
동적 업데이트	**dynamic_update**=TRUE|FALSE	클라이언트의 DNS 레코드에 대한 동적 업데이트를 활성화합니다. false로 설정하면 IdM 클라이언트 시스템에서 IP 주소를 추가하거나 업데이트할 수 없습니다.
전송 허용	**allow_transfer**=*string*	지정된 영역을 부분(;)으로 구분하여 전송할 수 있는 IP 주소 또는 네트워크 이름 목록을 제공합니다. 영역 전송은 기본적으로 비활성화되어 있습니다. 기본 **allow_transfer** 값은 **none** 입니다.
쿼리 허용	**allow_query**	Semicolons (;)로 구분된 DNS 쿼리를 실행할 수 있는 IP 주소 또는 네트워크 이름 목록을 제공합니다.
PTR 동기화 허용	**allow_sync_ptr**=1|0	영역의 A 또는 AAAA 레코드(전달 레코드)가 PTR 레코드(reverse) 레코드와 자동으로 동기화되는지 여부를 설정합니다.

속성	Ansible-freeipa 변수	설명
영역 전달자	**forwarder**=*IP_address*	DNS 영역을 위해 특별히 구성된 전달자를 지정합니다. IdM 도메인에 사용된 글로벌 전달자와는 별개입니다. 여러 전달자를 지정하려면 옵션을 여러 번 사용합니다.
forward policy	**forward_policy**=none|only|first	전달 정책을 지정합니다. 지원되는 정책에 대한 자세한 내용은 IdM의 DNS 전달 정책을 참조하십시오.

추가 리소스

- */usr/share/doc/ansible-freeipa/* 디렉토리의 ***README-dnszone.md*** 파일을 참조하십시오.

3.3. ANSIBLE을 사용하여 IDM DNS에 기본 영역 생성

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 기본 DNS 영역이 있는지 확인하는 방법을 보여줍니다. 아래 절차에 사용된 예에서 IdM 관리자는 ***zone.idm.example.com*** *DNS 영역이 있는지 확인합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*
- *IdM 관리자 암호를 알고 있습니다.*

절차

1. ***/usr/share/doc/ansible-freeipa/playbooks/dnszone*** *디렉터리로 이동합니다.*

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnszone
```

2. *인벤토리 파일을 열고 구성하려는 IdM 서버가* ***[ipaserver]*** *섹션에 나열되어 있는지 확인합니다. 예를 들어,* ***server.idm.example.com*** *을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.*

```
[ipaserver]
server.idm.example.com
```

3. ***dnszone-present.yml*** *Ansible 플레이북 파일을 복사합니다. 예를 들어 다음과 같습니다.*

```
$ cp dnszone-present.yml dnszone-present-copy.yml
```

4. *편집할* ***dnszone-present-copy.yml*** *파일을 엽니다.*
5. ***ipadnszone*** *작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.*
   - ***ipaadmin_password*** *변수를 IdM 관리자 암호로 설정합니다.*
   - ***zone_name*** *변수를* ***zone.idm.example.com*** *으로 설정합니다.*
     이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Ensure dnszone present
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure zone is present.
    ipadnszone:
      ipaadmin_password: Secret123
      zone_name: zone.idm.example.com
      state: present
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file dnszone-present-copy.yml
```

추가 리소스

- *지원되는 DNS 영역 유형을 참조하십시오.*
- ***/usr/share/doc/ansible-freeipa/*** *디렉토리의* ***README-dnszone.md*** *파일을 참조하십시오.*
- ***/usr/share/doc/ansible-freeipa/playbooks/dnszone*** *디렉터리에서 샘플 Ansible 플레이북을 참조하십시오.*

3.4. ANSIBLE 플레이북을 사용하여 IDM에 여러 변수가 있는 기본 DNS 영역이 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 기본 DNS 영역이 있는지 확인하는 방법을 보여줍니다. 아래 절차에 사용된 예에서 IdM 관리자는 ***zone.idm.example.com*** *DNS 영역이 있는지 확인합니다. Ansible 플레이북은 영역의 여러 매개 변수를 구성합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다.*
- *IdM 관리자 암호를 알고 있습니다.*

절차

1. ***/usr/share/doc/ansible-freeipa/playbooks/dnszone*** *디렉터리로 이동합니다.*

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnszone
```

2. *인벤토리 파일을 열고 구성하려는 IdM 서버가* ***[ipaserver]*** *섹션에 나열되어 있는지 확인합니다. 예를 들어,* ***server.idm.example.com*** *을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.*

```
[ipaserver]
server.idm.example.com
```

3. ***dnszone-all-params.yml*** *Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.*

```
$ cp dnszone-all-params.yml dnszone-all-params-copy.yml
```

4. 편집할 **dnszone-all-params-copy.yml** 파일을 엽니다.

5. ***ipadnszone*** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.

   - ***ipaadmin_password*** 변수를 IdM 관리자 암호로 설정합니다.
   - ***zone_name*** 변수를 **zone.idm.example.com** 으로 설정합니다.
   - A 및 AAAA 레코드의 동기화를 PTR 레코드와 AAAA 레코드의 동기화를 허용하려면 ***allow_sync_ptr*** 변수를 true로 설정합니다.
   - IdM 클라이언트 시스템에서 IP 주소를 추가하거나 업데이트하는 데 ***dynamic_update*** 변수를 true로 설정합니다.
   - 영역에 레코드의 인라인 DNSSEC 서명을 허용하려면 ***dnssec*** 변수를 true로 설정합니다.
   - ***allow_transfer*** 변수를 영역에 있는 보조 이름 서버의 IP 주소로 설정합니다.
   - ***allow_query*** 변수를 쿼리를 실행할 수 있는 IP 주소 또는 네트워크로 설정합니다.
   - ***전달자*** 변수를 글로벌 전달자의 IP 주소로 설정합니다.
   - ***serial*** 변수를 SOA 레코드 일련 번호로 설정합니다.
   - 영역의 DNS 레코드에 대한 ***새로 고침,재시도,만료,최소,ttl*** 및 ***default_ttl*** 값을 정의합니다.
   - ***nsec3param_rec*** 변수를 사용하여 영역에 대한 NSEC3PARAM 레코드를 정의합니다.
   - 기존 영역과 겹치는 경우에도 ***skip_overlap_check*** 변수를 true로 설정합니다.
   - 이름 서버가 확인할 수 없는 경우에도 DNS 영역 생성을 강제 적용하려면 ***skip_nameserver_check*** 를 true로 설정합니다.
     이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Ensure dnszone present
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure zone is present.
    ipadnszone:
      ipaadmin_password: Secret123
      zone_name: zone.idm.example.com
      allow_sync_ptr: true
      dynamic_update: true
      dnssec: true
      allow_transfer:
        - 1.1.1.1
        - 2.2.2.2
      allow_query:
        - 1.1.1.1
        - 2.2.2.2
      forwarders:
```

```
    - ip_address: 8.8.8.8
    - ip_address: 8.8.4.4
      port: 52
  serial: 1234
  refresh: 3600
  retry: 900
  expire: 1209600
  minimum: 3600
  ttl: 60
  default_ttl: 90
  name_server: server.idm.example.com.
  admin_email: admin.admin@idm.example.com
  nsec3param_rec: "1 7 100 0123456789abcdef"
  skip_overlap_check: true
  skip_nameserver_check: true
  state: present
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file dnszone-all-params-copy.yml
```

추가 리소스

- *지원되는 DNS 영역 유형을 참조하십시오.*
- *기본 IdM DNS 영역의 구성 속성 을 참조하십시오.*
- ***/usr/share/doc/ansible-freeipa/** 디렉토리의 **README-dnszone.md** 파일을 참조하십시오.*
- ***/usr/share/doc/ansible-freeipa/playbooks/dnszone** 디렉터리에서 샘플 Ansible 플레이북을 참조하십시오.*

3.5. ANSIBLE 플레이북을 사용하여 IP 주소가 지정될 때 역방향 DNS 조회 영역이 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 역방향 DNS 영역이 있는지 확인하는 방법을 보여줍니다. 아래 절차에 사용된 예에서 IdM 관리자는 IdM 호스트의 IP 주소 및 접두사 길이를 사용하여 역방향 DNS 조회 영역이 있는지 확인합니다.

***name_from_ip** 변수를 사용하여 DNS 서버의 IP 주소 접두사 길이를 제공하면 영역 이름을 제어할 수 있습니다. 접두사 길이를 지정하지 않으면 시스템은 DNS 서버를 영역을 쿼리하고 **name_from_ip** 값 **192.168.1.2** 를 기반으로 쿼리는 다음 DNS 영역 중 하나를 반환할 수 있습니다.*

- *1.168.192.in-addr.arpa.*
- *168.192.in-addr.arpa.*
- *192.in-addr.arpa.*

*쿼리에서 반환된 영역이 예상하지 않을 수 있기 때문에 **name_from_ip** 는 실수로 영역 제거를 방지하기 위해 **present** 로 설정된 **state** 옵션에서만 사용할 수 있습니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*
- *IdM 관리자 암호를 알고 있습니다.*

절차

1. ***/usr/share/doc/ansible-freeipa/playbooks/dnszone** 디렉터리로 이동합니다.*

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnszone
```

2. *인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.*

```
[ipaserver]
server.idm.example.com
```

3. ***dnszone-reverse-from-ip.yml** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.*

```
$ cp dnszone-reverse-from-ip.yml dnszone-reverse-from-ip-copy.yml
```

4. *편집할 **dnszone-reverse-from-ip-copy.yml** 파일을 엽니다.*
5. ***ipadnszone** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.*
   - ***ipaadmin_password** 변수를 IdM 관리자 암호로 설정합니다.*
   - ***name_from_ip** 변수를 IdM 이름 서버의 IP로 설정하고 접두사 길이를 제공합니다. 이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.*

```
---
- name: Ensure dnszone present
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure zone for reverse DNS lookup is present.
    ipadnszone:
      ipaadmin_password: Secret123
      name_from_ip: 192.168.1.2/24
      state: present
    register: result
  - name: Display inferred zone name.
    debug:
      msg: "Zone name: {{ result.dnszone.name }}"
```

   *Playbook은 **192.168.1.2** IP 주소 및 해당 접두사 길이 24에서 역방향 DNS 조회 영역을 생성합니다. 다음으로 플레이북은 결과 영역 이름을 표시합니다.*

6. *파일을 저장합니다.*
7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file dnszone-reverse-from-ip-copy.yml
```

추가 리소스

- *지원되는 DNS 영역 유형을 참조하십시오.*
- ***/usr/share/doc/ansible-freeipa/** 디렉토리의 **README-dnszone.md** 파일을 참조하십시오.*
- ***/usr/share/doc/ansible-freeipa/playbooks/dnszone** 디렉터리에서 샘플 Ansible 플레이북을 참조하십시오.*

4 장. IDM 의 DNS 위치 관리

IdM(Identity Management) 관리자는 IdM 웹 UI 및 IdM CLI(명령줄 인터페이스) 를 사용하여 IdM(Identity Management) DNS 위치를 관리할 수 있습니다. 이 장에서는 다음 주제 및 절차에 대해 설명합니다.

- DNS 기반 서비스 검색
- DNS 위치에 대한 배포 고려 사항
- DNS 시간 (TTL)
- IdM 웹 UI 를 사용하여 DNS 위치 생성
- IdM CLI 를 사용하여 DNS 위치 생성
- IdM 웹 UI 를 사용하여 DNS 위치에 IdM 서버 할당
- IdM 웹 UI 를 사용하여 DNS 위치에 IdM 서버 할당
- IdM 서버를 동일한 위치에서 사용하도록 IdM 클라이언트 구성

4.1. DNS 기반 서비스 검색

DNS 기반 서비스 검색은 클라이언트가 DNS 프로토콜을 사용하여 **LDAP** 또는 **Kerberos** 와 같은 특정 서비스를 제공하는 네트워크의 서버를 찾는 프로세스입니다. 일반적인 작업 유형 중 하나는 클라이언트가 더 높은 처리량과 네트워크 대기 시간을 줄이며 전체 비용을 절감하므로 가장 가까운 네트워크 인프라 내에서 인증 서버를 찾을 수 있도록 하는 것입니다.

서비스 검색의 주요 장점은 다음과 같습니다.

- 가까운 서버의 이름으로 명시적으로 구성할 필요는 없습니다.
- DNS 서버는 정책의 중앙 공급자로 사용됩니다. 동일한 DNS 서버를 사용하는 클라이언트는 서비스 공급자 및 기본 순서에 대한 동일한 정책에 액세스할 수 있습니다.

IdM(Identity Management) 도메인에서 **LDAP**,**Kerberos** 및 기타 서비스에 대한 DNS 서비스 레코드(SRV 레코드) 가 있습니다. 예를 들어 다음 명령은 IdM DNS 도메인에 TCP 기반 **Kerberos** 서비스를 제공하는 호스트의 DNS 서버를 쿼리합니다.

예 4.1. DNS 위치 독립 결과

```
$ dig -t SRV +short _kerberos._tcp.idm.example.com
0 100 88 idmserver-01.idm.example.com.
0 100 88 idmserver-02.idm.example.com.
```

출력에는 다음 정보가 포함됩니다.

- **0** (우선 순위): 대상 호스트의 우선 순위입니다. 더 낮은 값이 선호됩니다.
- **100** (가장). 동일한 우선순위가 있는 항목의 상대적 가중치를 지정합니다. 자세한 내용은 RFC 2782, 섹션 3 에서 참조하십시오.
- **88** (포트 번호): 서비스의 포트 번호.
- 서비스를 제공하는 호스트의 정식 이름입니다.

이 예제에서 반환된 두 호스트 이름은 동일한 우선 순위와 weight를 갖습니다. 이 경우 클라이언트는 결과 목록의 임의의 항목을 사용합니다.

*클라이언트가 DNS 위치에 구성된 DNS 서버를 쿼리하도록 구성된 경우 출력이 달라집니다. 위치에 할당된 IdM 서버의 경우 맞춤형 값이 반환됩니다. 아래 예제에서 클라이언트는 위치 **germany** 에서 DNS 서버를 쿼리하도록 구성되어 있습니다.*

> ***예 4.2. DNS 위치 기반 결과***
>
> ```
> $ dig -t SRV +short _kerberos._tcp.idm.example.com
> _kerberos._tcp.germany._locations.idm.example.com.
> 0 100 88 idmserver-01.idm.example.com.
> 50 100 88 idmserver-02.idm.example.com.
> ```

*IdM DNS 서버는 로컬 서버를 선호하는 DNS 위치별 SRV 레코드를 가리키는 DNS 별칭(CNAME)을 자동으로 반환합니다. 이 CNAME 레코드는 출력의 첫 번째 줄에 표시됩니다. 이 예제에서 호스트 **idmserver-01.idm.example.com** 은 우선 순위가 가장 낮은 값이 있으므로 우선합니다. **idmserver-02.idm.example.com** 은 우선 순위가 높습니다. 따라서 기본 호스트를 사용할 수 없는 경우 백업으로만 사용됩니다.*

4.2. DNS 위치에 대한 배포 고려 사항

IdM(Identity Management)은 통합 DNS를 사용할 때 위치별 서비스(SRV) 레코드를 생성할 수 있습니다. IdM DNS 서버는 위치별 SRV 레코드를 생성하기 때문에 각 DNS 위치에 적어도 하나의 IdM DNS 서버를 설치해야 합니다.

DNS 위치에 대한 클라이언트의 선호도는 클라이언트에서 수신한 DNS 레코드에서만 정의됩니다. 이러한 이유로 IdM DNS 서버를 비IdM DNS 소비자 서버와 결합하고 DNS 서비스 검색을 수행하는 클라이언트가 IdM DNS 서버에서 위치별 레코드를 확인하면 재구성할 수 있습니다.

IdM 및 비IdM DNS 서비스가 혼합된 대부분의 배포에서 DNS 재귀자는 왕복 시간 메트릭을 사용하여 가장 가까운 IdM DNS 서버를 자동으로 선택합니다. 일반적으로 IdM이 아닌 DNS 서버를 사용하는 클라이언트가 가장 가까운 DNS 위치에 대한 레코드를 수신하므로 최적의 IdM 서버 세트를 사용할 수 있습니다.

4.3. DNS 시간(TTL)

*클라이언트는 영역의 구성에 설정된 시간 동안 DNS 리소스 레코드를 캐시할 수 있습니다. 이러한 캐싱으로 인해 클라이언트는 TTL(Time to live) 값이 만료될 때까지 변경 사항을 수신하지 못할 수 있습니다. IdM(Identity Management)의 기본 TTL 값은 **1일입니다**.*

클라이언트 컴퓨터가 사이트 간에 로밍하는 경우 IdM DNS 영역의 TTL 값을 조정해야 합니다. 클라이언트가 사이트 간에 로밍하는 데 필요한 시간보다 값을 더 낮은 값으로 설정합니다. 이렇게 하면 다른 사이트에 다시 연결하기 전에 클라이언트의 캐시된 DNS 항목이 만료되므로 DNS 서버를 쿼리하여 위치별 SRV 레코드를 새로 고칩니다.

추가 리소스

- *기본 IdM DNS 영역의 구성 속성 을 참조하십시오.*

4.4. IDM 웹 UI를 사용하여 DNS 위치 생성

DNS 위치를 사용하여 IdM(Identity Management) 클라이언트 및 서버 간 통신 속도를 높일 수 있습니다. 이 섹션에서는 IdM 웹 UI를 사용하여 DNS 위치를 생성하는 방법을 설명합니다.

사전 요구 사항

- IdM 배포에 DNS가 통합되어 있습니다.
- IdM에서 DNS 위치를 생성할 수 있는 권한이 있습니다. 예를 들어 IdM 관리자로 로그인되어 있습니다.

절차

1. **IPA 서버** 탭을 엽니다.
2. **Topology** (토폴로지) 하위 탭을 선택합니다.
3. 네비게이션 바에서 **IPA Locations** (IPA 위치)를 클릭합니다.
4. 위치 목록 상단에서 **추가를** 클릭합니다.
5. 위치 이름을 입력합니다.
6. **추가** 버튼을 클릭하여 위치를 저장합니다.
7. 선택 사항: 단계를 반복하여 추가 위치를 추가합니다.

추가 리소스

- IdM 웹 UI를 사용하여 DNS 위치에 IdM 서버 할당을 참조하십시오.
- IdM 위치가 있는지 확인하려면 Ansible 사용을 참조하십시오.

4.5. IDM CLI를 사용하여 DNS 위치 생성

DNS 위치를 사용하여 IdM(Identity Management) 클라이언트 및 서버 간 통신 속도를 높일 수 있습니다. 이 섹션에서는 IdM CLI(명령줄 인터페이스)에서 **ipa location-add** 명령을 사용하여 DNS 위치를 생성하는 방법을 설명합니다.

사전 요구 사항

- IdM 배포에 DNS가 통합되어 있습니다.
- IdM에서 DNS 위치를 생성할 수 있는 권한이 있습니다. 예를 들어 IdM 관리자로 로그인되어 있습니다.

절차

1. 예를 들어 새 위치 **germany** 를 만들려면 다음을 입력합니다.

```
$ ipa location-add germany
---------------------------
Added IPA location "germany"
---------------------------
  Location name: germany
```

2. 선택 사항: 단계를 반복하여 추가 위치를 추가합니다.

추가 리소스

- IdM CLI를 사용하여 DNS 위치에 IdM 서버 할당을 참조하십시오.
- IdM 위치가 있는지 확인하려면 Ansible 사용을 참조하십시오.

4.6. IDM 웹 UI를 사용하여 DNS 위치에 IDM 서버 할당

IdM(Identity Management) DNS 위치를 사용하여 IdM 클라이언트와 서버 간 통신 속도를 높일 수 있습니다. 이 섹션에서는 IdM 웹 UI를 사용하여 DNS 위치에 IdM 서버를 할당하는 방법을 설명합니다.

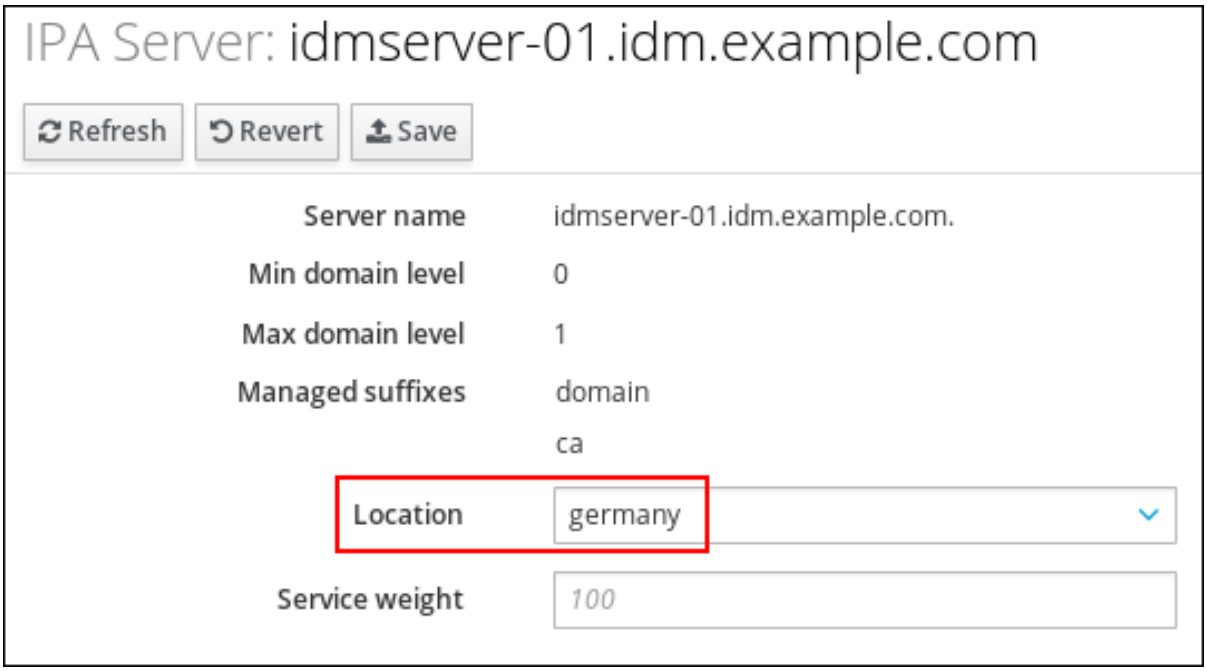
사전 요구 사항

- IdM 배포에 DNS가 통합되어 있습니다.
- DNS 위치(예: IdM admin 사용자)에 서버를 할당할 수 있는 권한이 있는 사용자로 로그인되어 있습니다.
- DNS 위치를 할당하려는 호스트에 대한 **루트** 액세스 권한이 있습니다.
- 서버를 할당할 IdM DNS 위치를 생성 했습니다.

절차

1. **IPA 서버** 탭을 엽니다.
2. **Topology**(토폴로지) 하위 탭을 선택합니다.
3. 탐색에서 **IPA 서버**를 클릭합니다.
4. IdM 서버 이름을 클릭합니다.
5. DNS 위치를 선택하고 선택적으로 서비스 가중치를 설정합니다.

그림 4.1. DNS 위치에 서버 할당



6. *저장을 클릭합니다.*

7. *이전 단계에서 할당한 호스트의 CLI(명령줄 인터페이스)에서 **named-pkcs11** 서비스를 다시 시작합니다.*

```
[root@idmserver-01 ~]# systemctl restart named-pkcs11
```

8. *선택 사항: 단계를 반복하여 추가 IdM 서버에 DNS 위치를 할당합니다.*

추가 리소스

- *동일한 위치에서 IdM 서버를 사용하도록 IdM 클라이언트 구성을 참조하십시오.*

4.7. IDM CLI를 사용하여 DNS 위치에 IDM 서버 할당

IdM(Identity Management) DNS 위치를 사용하여 IdM 클라이언트와 서버 간 통신 속도를 높일 수 있습니다. 이 섹션에서는 IdM CLI(명령줄 인터페이스)를 사용하여 DNS 위치에 IdM 서버를 할당하는 방법을 설명합니다.

사전 요구 사항

- *IdM 배포에 DNS가 통합되어 있습니다.*
- *DNS 위치(예: IdM admin 사용자)에 서버를 할당할 수 있는 권한이 있는 사용자로 로그인되어 있습니다.*
- *DNS 위치를 할당하려는 호스트에 대한 **루트** 액세스 권한이 있습니다.*
- *서버를 할당할 IdM DNS 위치를 생성 했습니다.*

절차

1. *선택 사항: 구성된 모든 DNS 위치를 나열합니다.*

```
[root@server ~]# ipa location-find
----------------------
2 IPA locations matched
----------------------
Location name: australia
Location name: germany
----------------------------
Number of entries returned: 2
----------------------------
```

2. 서버를 DNS 위치에 할당합니다. 예를 들어 위치 **germany** 를 서버 **idmserver-01.idm.example.com** 에 할당하려면 다음을 실행합니다.

```
# ipa server-mod idmserver-01.idm.example.com --location=germany
ipa: WARNING: Service named-pkcs11.service requires restart on IPA server
idmserver-01.idm.example.com to apply configuration changes.
-----------------------------------------------
Modified IPA server "idmserver-01.idm.example.com"
-----------------------------------------------
Servername: idmserver-01.idm.example.com
Min domain level: 0
Max domain level: 1
Location: germany
Enabled server roles: DNS server, NTP server
```

3. 이전 단계에서 할당한 호스트에서 **named-pkcs11** 서비스를 다시 시작합니다. DNS 위치는 다음을 수행합니다.

```
# systemctl restart named-pkcs11
```

4. 선택 사항: 단계를 반복하여 추가 IdM 서버에 DNS 위치를 할당합니다.

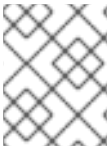
추가 리소스

- 동일한 위치에서 IdM 서버를 사용하도록 IdM 클라이언트 구성을 참조하십시오.

4.8. IDM 서버를 동일한 위치에서 사용하도록 IDM 클라이언트 구성

IdM(Identity Management) 서버는 IdM 웹 UI를 사용하여 DNS 위치에 IdM 서버 할당에 설명된 대로 DNS 위치에 할당 됩니다. 이제 IdM 서버와 동일한 위치에 있는 DNS 서버를 사용하도록 클라이언트를 구성할 수 있습니다.

- **DHCP** 서버가 DNS 서버 IP 주소를 클라이언트에 할당하는 경우 **DHCP** 서비스를 구성합니다. **DHCP** 서비스에서 DNS 서버를 할당하는 방법에 대한 자세한 내용은 **DHCP** 서비스 설명서를 참조하십시오.
- 클라이언트에서 **DHCP** 서버에서 DNS 서버 IP 주소를 수신하지 못하면 클라이언트의 네트워크 구성에 IP를 수동으로 설정합니다. Red Hat Enterprise Linux에서 네트워크 구성에 대한 자세한 내용은 Red Hat Enterprise Linux 네트워킹 가이드의 네트워크 연결 설정 구성 섹션을 참조하십시오.



참고

다른 위치에 할당된 DNS 서버를 사용하도록 클라이언트를 구성하는 경우 클라이언트 모두의 IdM 서버에 연결합니다.

예 4.3. 클라이언트의 위치에 따라 다른 이름 서버 항목

*다음 예제에서는 다른 위치에 있는 클라이언트의 **/etc/resolv.conf** 파일에서 다른 이름 서버 항목을 보여줍니다.*

프라하의 고객:

```
nameserver 10.10.0.1
nameserver 10.10.0.2
```

일본의 고객:

```
nameserver 10.50.0.1
nameserver 10.50.0.3
```

Oslo의 고객:

```
nameserver 10.30.0.1
```

2017년 12월 21일:

```
nameserver 10.30.0.1
```

각 DNS 서버가 IdM의 위치에 할당되면 클라이언트는 해당 위치에서 IdM 서버를 사용합니다.

4.9. 추가 리소스

- *Ansible을 사용하여 IdM의 DNS 위치 관리를* 참조하십시오.

5장. ANSIBLE을 사용하여 IDM의 DNS 위치 관리

IdM(Identity Management) 관리자는 **ansible-freeipa** 패키지에서 사용할 수 있는 **location** 모듈을 사용하여 IdM DNS 위치를 관리할 수 있습니다. 이 장에서는 다음 주제 및 절차에 대해 설명합니다.

- DNS 기반 서비스 검색
- DNS 위치에 대한 배포 고려 사항
- DNS 시간(TTL)
- Ansible을 사용하여 IdM 위치 확인
- Ansible을 사용하여 IdM 위치 없음

5.1. DNS 기반 서비스 검색

DNS 기반 서비스 검색은 클라이언트가 DNS 프로토콜을 사용하여 **LDAP** 또는 **Kerberos** 와 같은 특정 서비스를 제공하는 네트워크의 서버를 찾는 프로세스입니다. 일반적인 작업 유형 중 하나는 클라이언트가 더 높은 처리량과 네트워크 대기 시간을 줄이며 전체 비용을 절감하므로 가장 가까운 네트워크 인프라 내에서 인증 서버를 찾을 수 있도록 하는 것입니다.

서비스 검색의 주요 장점은 다음과 같습니다.

- 가까운 서버의 이름으로 명시적으로 구성할 필요는 없습니다.
- DNS 서버는 정책의 중앙 공급자로 사용됩니다. 동일한 DNS 서버를 사용하는 클라이언트는 서비스 공급자 및 기본 순서에 대한 동일한 정책에 액세스할 수 있습니다.

IdM(Identity Management) 도메인에서 **LDAP**,**Kerberos** 및 기타 서비스에 대한 DNS 서비스 레코드(SRV 레코드)가 있습니다. 예를 들어 다음 명령은 IdM DNS 도메인에 TCP 기반 **Kerberos** 서비스를 제공하는 호스트의 DNS 서버를 쿼리합니다.

예 5.1. DNS 위치 독립 결과

```
$ dig -t SRV +short _kerberos._tcp.idm.example.com
0 100 88 idmserver-01.idm.example.com.
0 100 88 idmserver-02.idm.example.com.
```

출력에는 다음 정보가 포함됩니다.

- **0**(우선 순위): 대상 호스트의 우선 순위입니다. 더 낮은 값이 선호됩니다.
- **100**(가장). 동일한 우선순위가 있는 항목의 상대적 가중치를 지정합니다. 자세한 내용은 RFC 2782, 섹션 3 에서 참조하십시오.
- **88**(포트 번호): 서비스의 포트 번호.
- 서비스를 제공하는 호스트의 정식 이름입니다.

이 예제에서 반환된 두 호스트 이름은 동일한 우선 순위와 weight를 갖습니다. 이 경우 클라이언트는 결과 목록의 임의의 항목을 사용합니다.

클라이언트가 DNS 위치에 구성된 DNS 서버를 쿼리하도록 구성된 경우 출력이 달라집니다. 위치에 할당된 IdM 서버의 경우 맞춤형 값이 반환됩니다. 아래 예제에서 클라이언트는 위치 **germany** 에서 DNS 서버를 쿼리하도록 구성되어 있습니다.

예 5.2. DNS 위치 기반 결과

```
$ dig -t SRV +short _kerberos._tcp.idm.example.com
_kerberos._tcp.germany._locations.idm.example.com.
0 100 88 idmserver-01.idm.example.com.
50 100 88 idmserver-02.idm.example.com.
```

IdM DNS 서버는 로컬 서버를 선호하는 DNS 위치별 SRV 레코드를 가리키는 DNS 별칭(CNAME)을 자동으로 반환합니다. 이 CNAME 레코드는 출력의 첫 번째 줄에 표시됩니다. 이 예제에서 호스트 **idmserver-01.idm.example.com** 은 우선 순위가 가장 낮은 값이 있으므로 우선합니다. **idmserver-02.idm.example.com** 은 우선 순위가 높습니다. 따라서 기본 호스트를 사용할 수 없는 경우 백업으로만 사용됩니다.

5.2. DNS 위치에 대한 배포 고려 사항

IdM(Identity Management)은 통합 DNS를 사용할 때 위치별 서비스(SRV) 레코드를 생성할 수 있습니다. IdM DNS 서버는 위치별 SRV 레코드를 생성하기 때문에 각 DNS 위치에 적어도 하나의 IdM DNS 서버를 설치해야 합니다.

DNS 위치에 대한 클라이언트의 선호도는 클라이언트에서 수신한 DNS 레코드에서만 정의됩니다. 이러한 이유로 IdM DNS 서버를 비IdM DNS 소비자 서버와 결합하고 DNS 서비스 검색을 수행하는 클라이언트가 IdM DNS 서버에서 위치별 레코드를 확인하면 재구성할 수 있습니다.

IdM 및 비IdM DNS 서비스가 혼합된 대부분의 배포에서 DNS 재귀자는 왕복 시간 메트릭을 사용하여 가장 가까운 IdM DNS 서버를 자동으로 선택합니다. 일반적으로 IdM이 아닌 DNS 서버를 사용하는 클라이언트가 가장 가까운 DNS 위치에 대한 레코드를 수신하므로 최적의 IdM 서버 세트를 사용할 수 있습니다.

5.3. DNS 시간(TTL)

클라이언트는 영역의 구성에 설정된 시간 동안 DNS 리소스 레코드를 캐시할 수 있습니다. 이러한 캐싱으로 인해 클라이언트는 TTL(Time to live) 값이 만료될 때까지 변경 사항을 수신하지 못할 수 있습니다. IdM(Identity Management)의 기본 TTL 값은 **1일입니다**.

클라이언트 컴퓨터가 사이트 간에 로밍하는 경우 IdM DNS 영역의 TTL 값을 조정해야 합니다. 클라이언트가 사이트 간에 로밍하는 데 필요한 시간보다 값을 더 낮은 값으로 설정합니다. 이렇게 하면 다른 사이트에 다시 연결하기 전에 클라이언트의 캐시된 DNS 항목이 만료되므로 DNS 서버를 쿼리하여 위치별 SRV 레코드를 새로 고칩니다.

추가 리소스

- 기본 IdM DNS 영역의 구성 속성 을 참조하십시오.

5.4. ANSIBLE을 사용하여 IDM 위치 확인

IdM(Identity Management)의 시스템 관리자로서 IdM DNS 위치를 구성하여 클라이언트가 가장 가까운 네트워크 인프라 내의 인증 서버를 찾을 수 있도록 할 수 있습니다.

*다음 절차에서는 Ansible 플레이북을 사용하여 IdM에 DNS 위치가 있는지 확인하는 방법을 설명합니다. 이 예제에서는 IdM에 **germany** DNS 위치가 있는지 확인하는 방법을 설명합니다. 결과적으로 로컬 IdM 클라이언트가 서버 응답 시간을 줄이는 데 사용할 수 있도록 특정 IdM 서버를 이 위치에 할당할 수 있습니다.*

사전 요구 사항

- *IdM 관리자 암호를 알고 있습니다.*
- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다.*
- *예제에서는 **~/MyPlaybooks/** 디렉터리를 샘플 플레이북의 복사본을 저장하기 위한 중앙 위치로 생성 및 구성 했다고 가정합니다.*
- *DNS 위치에 대한 배포 고려 사항을 이해합니다.*

절차

1. ***~/MyPlaybooks/** 디렉터리로 이동합니다.*

```
$ cd ~/MyPlaybooks/
```

2. ***/usr/share/doc/ansible-freeipa/playbooks/location/** 디렉터리에 있는 **location-present.yml** 파일의 사본을 만듭니다.*

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/location/location-present.yml location-present-copy.yml
```

3. *편집할 **location-present-copy.yml** Ansible 플레이북 파일을 엽니다.*
4. ***ipalocation** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.*
   - *사용 사례에 맞게 작업 **이름을** 조정합니다.*
   - ***ipaadmin_password** 변수를 IdM 관리자의 암호로 설정합니다.*
   - ***name** 변수를 위치 이름으로 설정합니다.*

   이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: location present example
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure that the "germany" location is present
    ipalocation:
      ipaadmin_password: Secret123
      name: germany
```

5. *파일을 저장합니다.*
6. *플레이북 파일 및 인벤토리 파일을 지정하여 Ansible 플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory location-present-copy.yml
```

추가 리소스

- *IdM 웹 UI를 사용하여 DNS 위치에 IdM 서버 할당 또는 IdM CLI를 사용하여 DNS 위치에 IdM 서버 할당을 참조하십시오.*

5.5. ANSIBLE을 사용하여 IDM 위치 없음

IdM(Identity Management)의 시스템 관리자로서 IdM DNS 위치를 구성하여 클라이언트가 가장 가까운 네트워크 인프라 내의 인증 서버를 찾을 수 있도록 할 수 있습니다.

*다음 절차에서는 Ansible 플레이북을 사용하여 IdM에서 DNS 위치가 없는지 확인하는 방법을 설명합니다. 이 예제에서는 IdM에서 **germany** DNS 위치가 없는지 확인하는 방법을 설명합니다. 결과적으로 특정 IdM 서버를 이 위치에 할당할 수 없으며 로컬 IdM 클라이언트에서 이 서버를 사용할 수 없습니다.*

사전 요구 사항

- *IdM 관리자 암호를 알고 있습니다.*
- *IdM 서버는 **germany** DNS 위치에 할당되지 않습니다.*
- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다.*
- *예제에서는 **~/MyPlaybooks/** 디렉터리를 샘플 플레이북의 복사본을 저장하기 위한 중앙 위치로 생성 및 구성 했다고 가정합니다.*

절차

1. ***~/MyPlaybooks/** 디렉터리로 이동합니다.*

```
$ cd ~/MyPlaybooks/
```

2. ***/usr/share/doc/ansible-freeipa/playbooks/location/** 디렉터리에 있는 **location-absent.yml** 파일의 사본을 만듭니다.*

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/location/location-absent.yml location-absent-copy.yml
```

3. *편집할 **location-absent-copy.yml** Ansible 플레이북 파일을 엽니다.*
4. ***ipalocation** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.*
   - *사용 사례에 맞게 작업 **이름을** 조정합니다.*
   - ***ipaadmin_password** 변수를 IdM 관리자의 암호로 설정합니다.*
   - ***name** 변수를 DNS 위치 이름으로 설정합니다.*
   - ***state** 변수가 **absent**로 설정되어 있는지 확인합니다.*

   이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: location absent example
  hosts: ipaserver
  become: true
```

```
tasks:
- name: Ensure that the "germany" location is absent
  ipalocation:
    ipaadmin_password: Secret123
    name: germany
    state: absent
```

5. *파일을 저장합니다.*

6. *플레이북 파일 및 인벤토리 파일을 지정하여 Ansible 플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory location-absent-copy.yml
```

5.6. 추가 리소스

- ***/usr/share/doc/ansible-freeipa/*** *디렉토리에 있는* ***README-location.md*** *파일을 참조하십시오.*
- ***/usr/share/doc/ansible-freeipa/playbooks/location*** *디렉터리에서 샘플 Ansible 플레이북을 참조하십시오.*

6 장. IDM 에서 DNS 전달 관리

다음 절차에서는 IdM(Identity Management) 웹 UI 에서 DNS 글로벌 전달자 및 DNS 전달 영역을 구성하는 방법, IdM CLI 및 Ansible 을 사용하는 방법을 설명합니다.

6.1. IDM DNS 서버의 두 가지 역할

DNS 전달은 DNS 서비스가 DNS 쿼리에 응답하는 방법에 영향을 미칩니다. 기본적으로 IdM 과 통합된 Berkeley Internet Name Domain(BIND) 서비스는 권한 있는 DNS 서버와 재귀 DNS 서버 역할을 합니다.

권한 있는 DNS 서버

DNS 클라이언트에서 IdM 서버에 속하는 이름을 쿼리하면 BIND 에서 구성된 영역에 포함된 데이터로 응답합니다. 권한 있는 데이터가 항상 다른 데이터보다 우선합니다.

재귀 DNS 서버

DNS 클라이언트에서 IdM 서버에 권한이 없는 이름을 쿼리하면 BIND 에서 다른 DNS 서버를 사용하여 쿼리를 확인하려고 합니다. 전달자가 정의되지 않은 경우 BIND 는 인터넷의 루트 서버를 요청하고 재귀 해결 알고리즘을 사용하여 DNS 쿼리에 응답합니다.

경우에 따라 BIND 에서 다른 DNS 서버에 직접 연결하고 인터넷에서 사용 가능한 데이터를 기반으로 재귀를 수행하는 것이 바람직하지 않습니다. 다른 DNS 서버인 전달자 를 사용하여 쿼리를 확인하도록 BIND 를 구성할 수 있습니다.

전달자를 사용하도록 BIND 를 구성하면 IdM 서버와 전달자 간에 쿼리 및 응답이 다시 전달되고 IdM 서버는 신뢰할 수 없는 데이터의 DNS 캐시 역할을 합니다.

6.2. IDM 의 DNS 전달 정책

IdM은 **첫** 번째 및 **유일한** 표준 BIND 전달 정책은 물론 **none** IdM 관련 전달 정책을 지원합니다.

forward first (기본값)

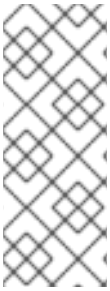
IdM BIND 서비스는 DNS 쿼리를 구성된 전달자에게 전달합니다. 서버 오류 또는 시간 초과로 인해 쿼리가 실패하면 BIND는 인터넷의 서버를 사용하여 재귀 해결으로 돌아갑니다. **첫 번째** 정책은 기본 정책이며 DNS 트래픽을 최적화하는 데 적합합니다.

forward only

IdM BIND 서비스는 DNS 쿼리를 구성된 전달자에게 전달합니다. 서버 오류 또는 시간 초과로 인해 쿼리가 실패하면 BIND에서 클라이언트에 오류를 반환합니다. 정 **방향 전용** 정책은 분할 DNS 구성이 있는 환경에 권장됩니다.

none (forwarding disabled)

DNS 쿼리는 **none** 전달 정책으로 전달되지 않습니다. 전달 비활성화는 글로벌 전달 구성에 대한 영역별 재정의로만 유용합니다. 이 옵션은 BIND 구성에서 빈 전달자 목록을 지정하는 것과 동일한 IdM입니다.



참고

전달을 사용하여 IdM의 데이터를 다른 DNS 서버의 데이터와 결합할 수 없습니다. IdM DNS에서 기본 영역의 특정 하위 영역에 대한 쿼리만 전달할 수 있습니다.

기본적으로 BIND 서비스는 쿼리된 DNS 이름이 IdM 서버에 권한이 있는 영역에 속하는 경우 쿼리를 다른 서버로 전달하지 않습니다. 이러한 상황에서 IdM 데이터베이스에서 쿼리된 DNS 이름을 찾을 수 없으면 **NXDOMAIN** 응답이 반환됩니다. 포워딩은 사용되지 않습니다.

예 6.1. 예를 들면 Scenario

IdM 서버는 **test.example에 대한 권한이 있습니다.** DNS 영역. BIND는 **192.0.2.254** IP 주소가 있는 DNS 서버로 쿼리를 전달하도록 구성됩니다.

클라이언트가 **존재하지 않는.test.example**에 대한 쿼리를 전송하는 경우 DNS 이름인 BIND에서는 IdM 서버가 **test.example.** 영역에 대한 권한이 있고 쿼리를 **192.0.2.254.** 서버로 전달하지 않음을 탐지합니다. 결과적으로 DNS 클라이언트는 **NXDomain** 오류 메시지를 수신하여 사용자에게 쿼리된 도메인이 존재하지 않음을 알립니다.

6.3. IDM 웹 UI에 글로벌 전달자 추가

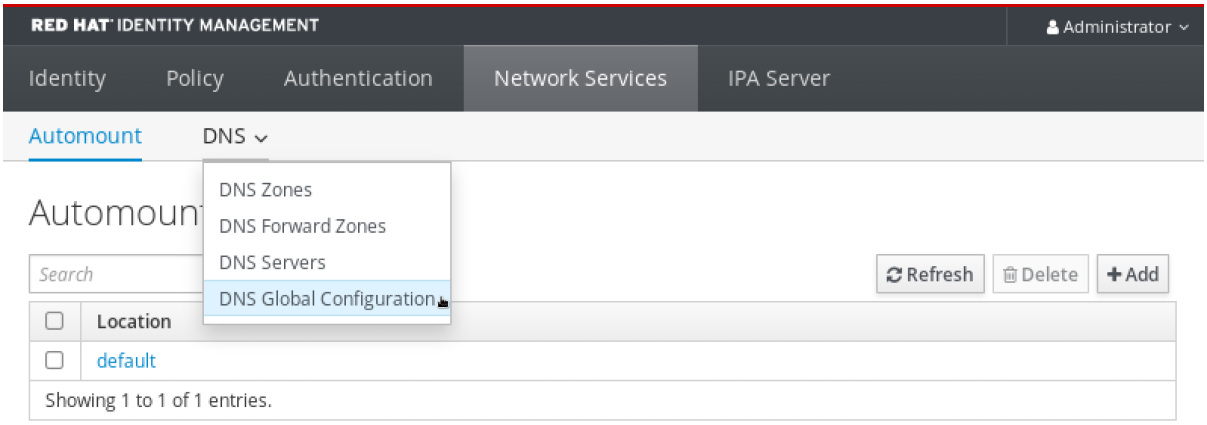
이 섹션에서는 IdM(Identity Management) 웹 UI에 글로벌 DNS 전달자를 추가하는 방법을 설명합니다.

사전 요구 사항

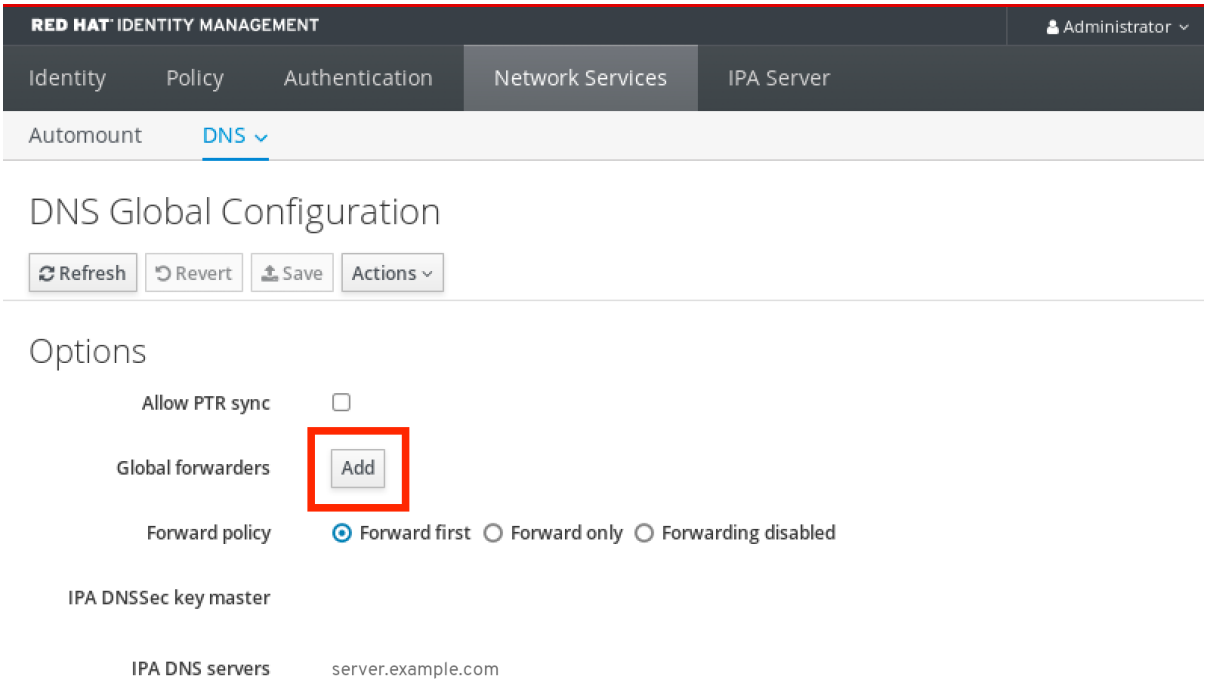
- IdM 관리자로 IdM 웹UI에 로그인되어 있습니다.
- 쿼리를 전달할 DNS 서버의 IP(Internet Protocol) 주소를 알고 있습니다.

절차

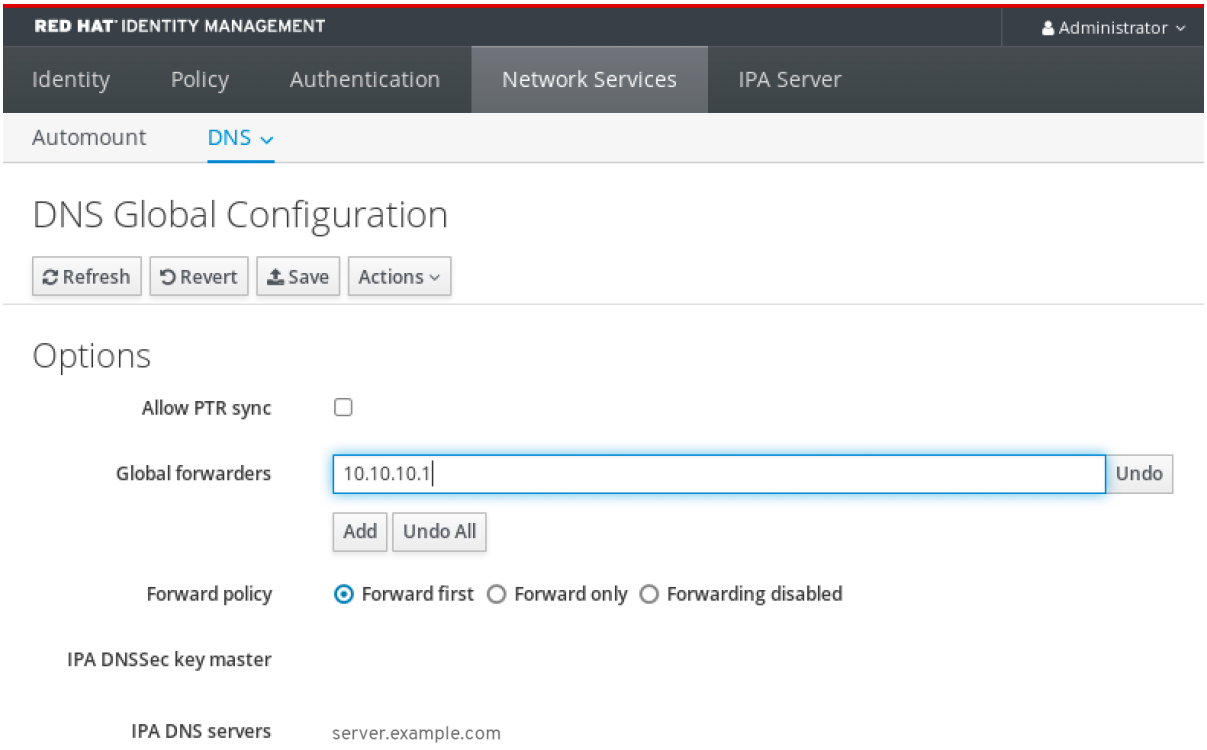
1. IdM 웹 UI에서 **네트워크 서비스 → DNS 글로벌 구성 → DNS** 를 선택합니다.



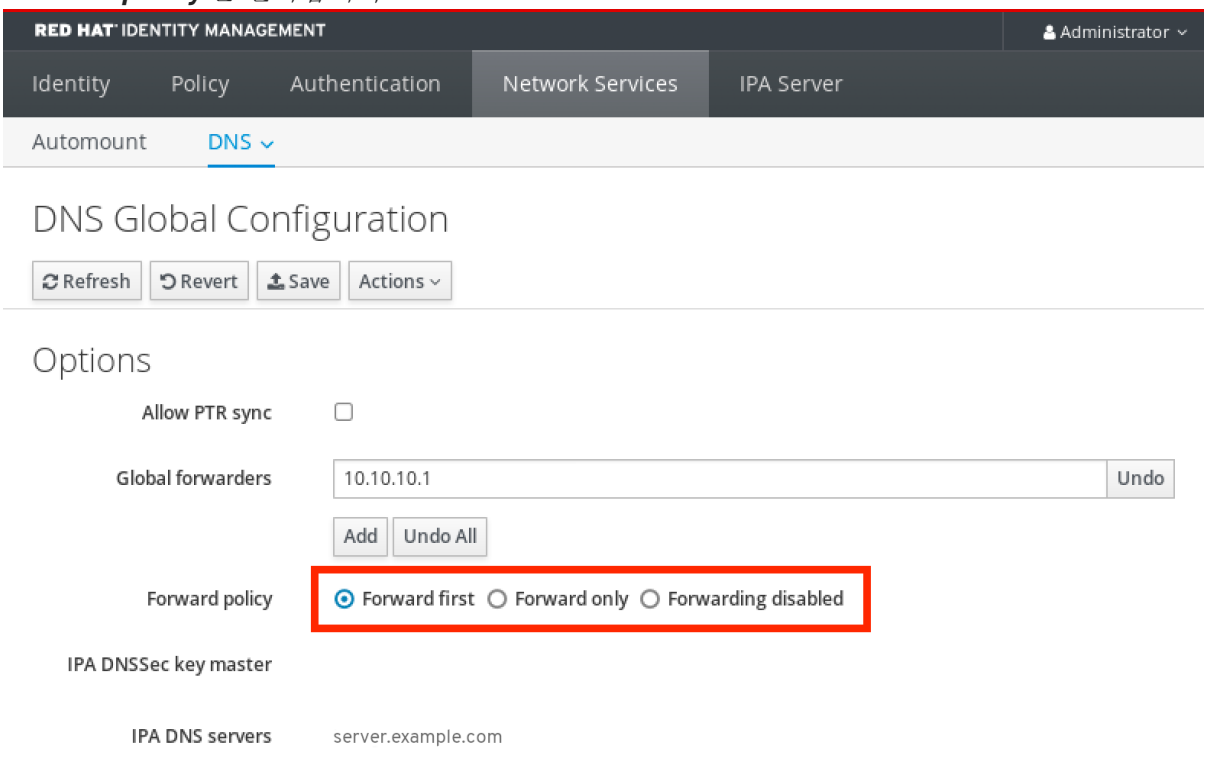
2. ***DNS** 글로벌 구성 섹션에서 **추가**를 클릭합니다.*



3. *전달된 DNS 쿼리를 수신할 DNS 서버의 IP 주소를 지정합니다.*



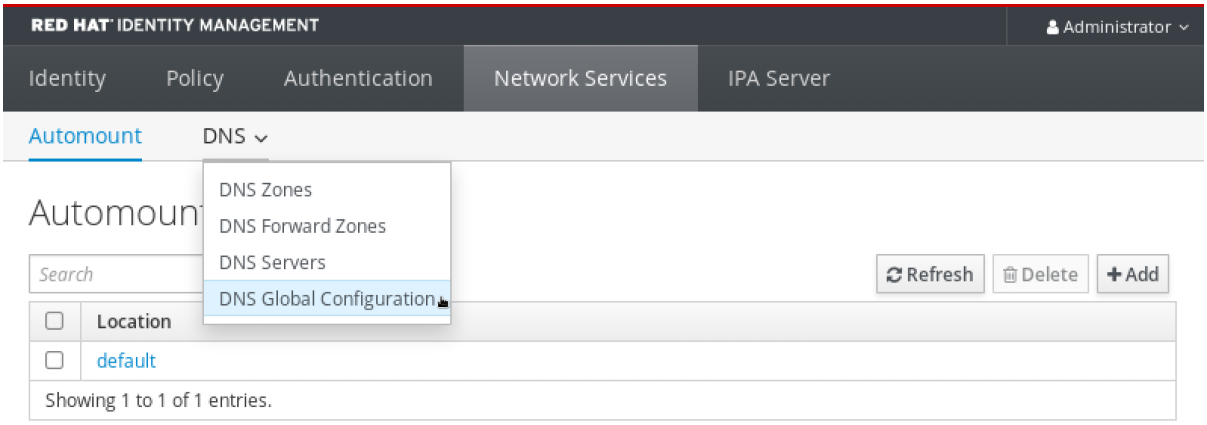
4. ***Forward policy*** *를 선택합니다.*



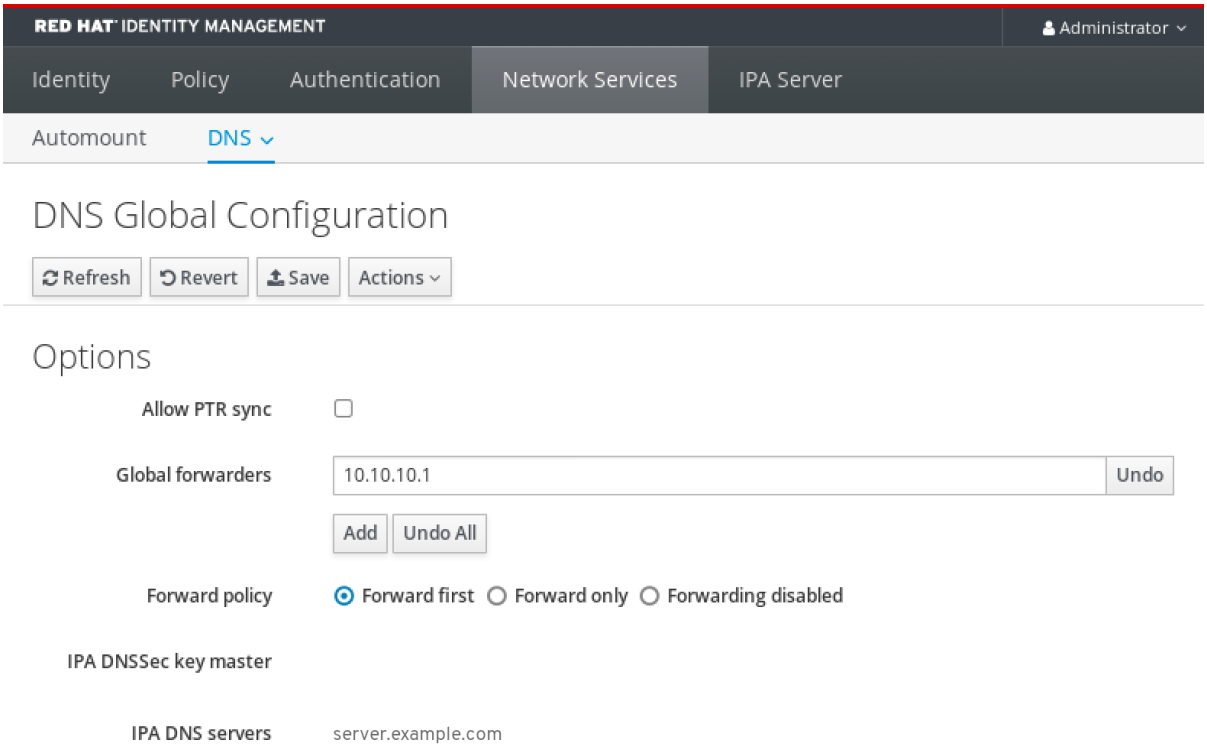
5. *창 위쪽의 저장을 클릭합니다.Click* ***Save*** *at the top of the window.*

검증 단계

1. ***네트워크 서비스 → DNS 글로벌 구성 → DNS*** *를 선택합니다.*



2. *IdM 웹 UI에 지정한 전달 정책이 있고 활성화되어 있는 글로벌 전달자가 있는지 확인합니다.*



6.4. CLI에 글로벌 전달자 추가

이 섹션에서는 CLI(명령줄 인터페이스)에서 글로벌 DNS 전달자를 추가하는 방법을 설명합니다.

사전 요구 사항

- *IdM 관리자로 로그인되어 있습니다.*
- *쿼리를 전달할 DNS 서버의 IP(Internet Protocol) 주소를 알고 있습니다.*

절차

- ***ipa dnsconfig-mod*** *명령을 사용하여 새 글로벌 전달자를 추가합니다.* ***--forwarder*** *옵션을 사용하여 DNS 전달자의 IP 주소를 지정합니다.*

```
[user@server ~]$ ipa dnsconfig-mod --forwarder=10.10.0.1
Server will check DNS forwarder(s).
This may take some time, please wait ...
```

```
Global forwarders: 10.10.0.1
IPA DNS servers: server.example.com
```

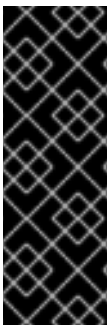
검증 단계

- ***dnsconfig-show*** *명령을 사용하여 글로벌 전달자를 표시합니다.*

```
[user@server ~]$ ipa dnsconfig-show
  Global forwarders: 10.10.0.1
  IPA DNS servers: server.example.com
```

6.5. IDM 웹 UI에 DNS FORWARD ZONE 추가

이 섹션에서는 IdM(Identity Management) 웹 UI에 DNS 전달 영역을 추가하는 방법을 설명합니다.



중요

반드시 필요한 경우를 제외하고 전달 영역을 사용하지 마십시오. 전달 영역은 표준 솔루션이 아니며 이를 사용하면 예기치 않은 문제가 발생할 수 있습니다. 전달 영역을 사용해야 하는 경우 글로벌 전달 구성을 재정의하는 데 사용을 제한합니다.

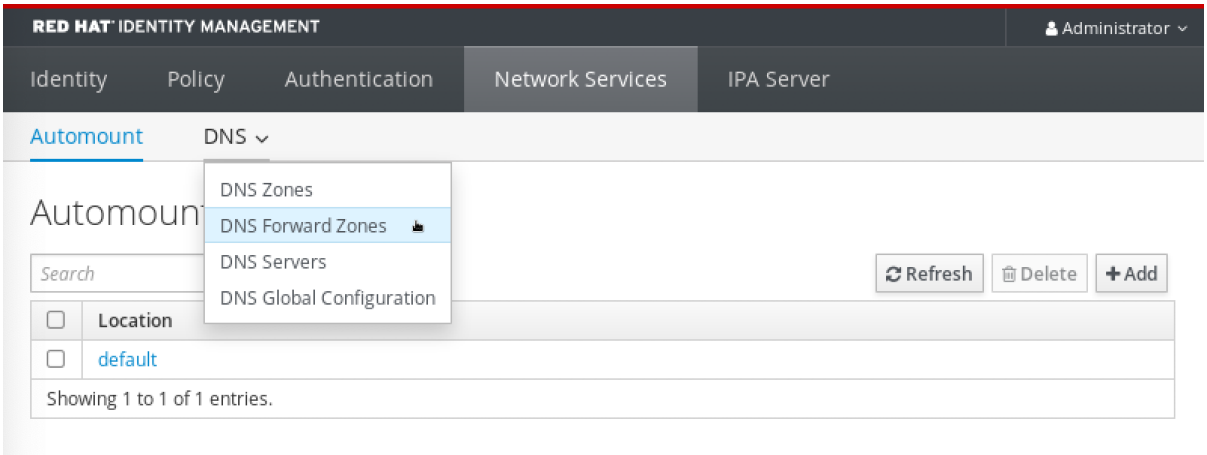
새 DNS 영역을 생성할 때 Red Hat은 항상 네임서버(NS) 레코드를 사용하여 표준 DNS 위임을 사용하고 정방향 영역을 방지하는 것이 좋습니다. 대부분의 경우 글로벌 전달자를 사용하는 것으로 충분하며 전달 영역이 필요하지 않습니다.

사전 요구 사항

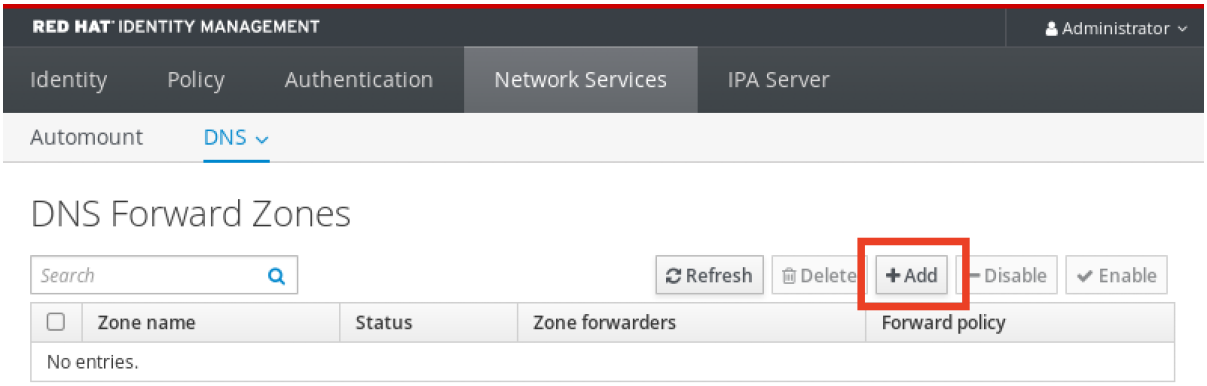
- *IdM 관리자로 IdM 웹UI에 로그인되어 있습니다.*
- *쿼리를 전달할 DNS 서버의 IP(Internet Protocol) 주소를 알고 있습니다.*

절차

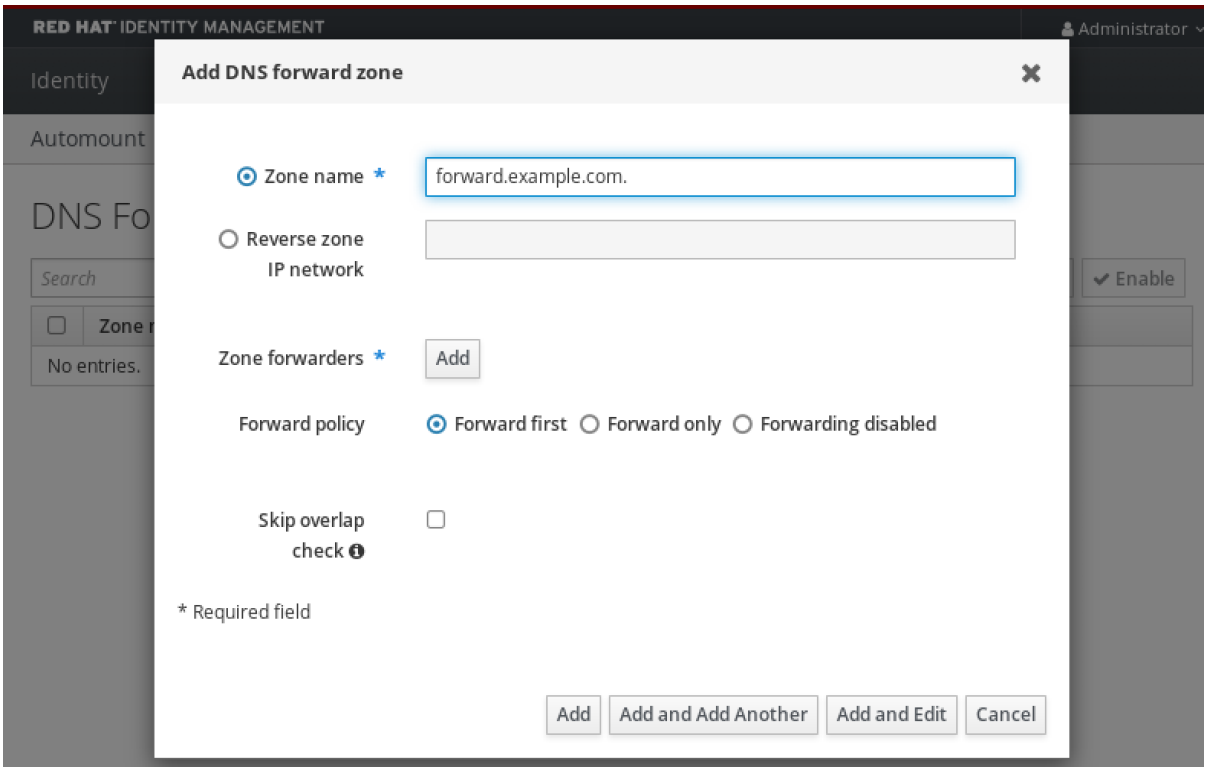
1. *IdM 웹 UI에서* ***네트워크 서비스 → DNS Forward Zones → DNS*** *를 선택합니다.*



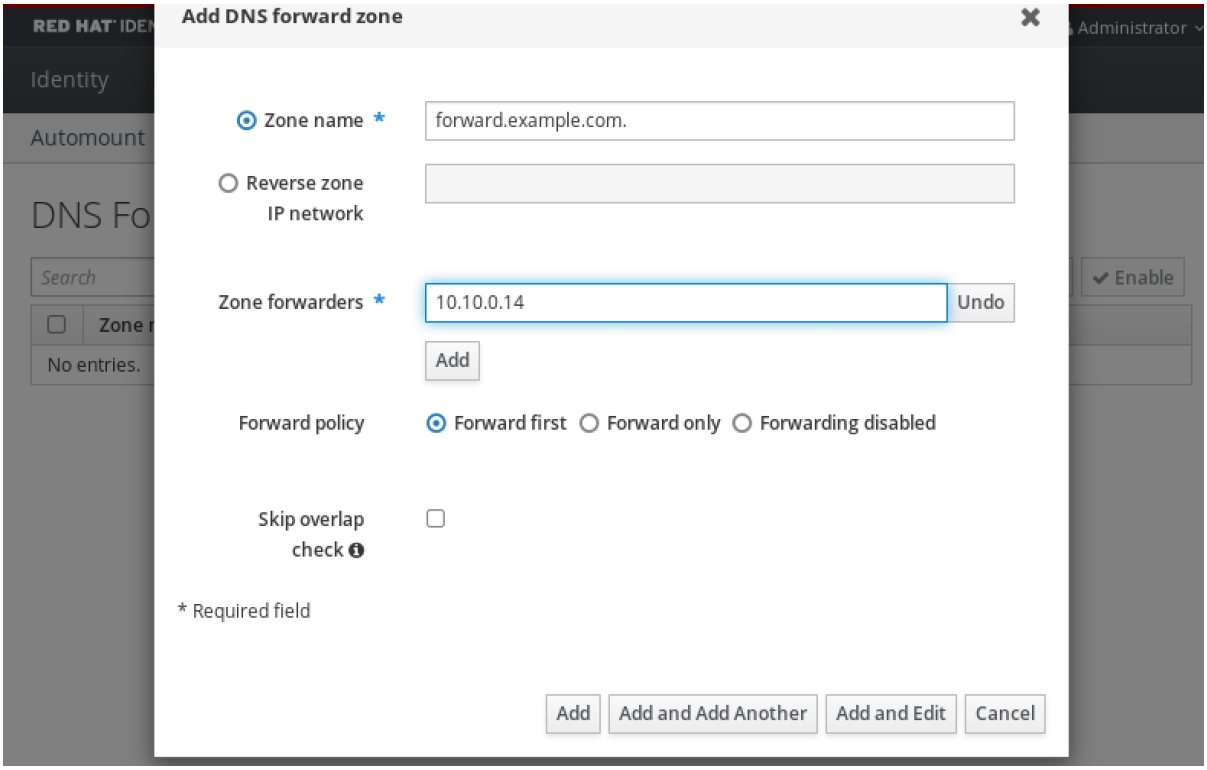
2. ***DNS Forward Zones*** *섹션에서* ***추가****를 클릭합니다.*



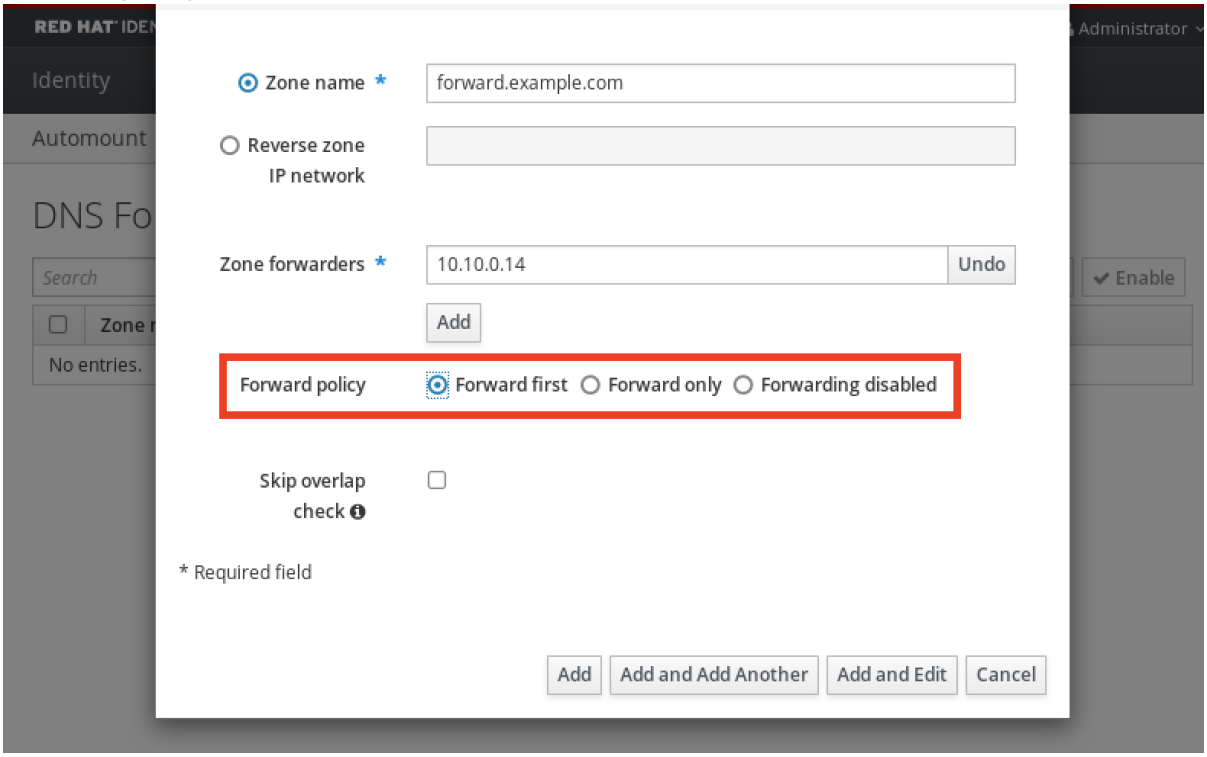
3. ***DNS 전달 영역 추가*** *창에서 전달 영역 이름을 지정합니다.*



4. ***추가*** *버튼을 클릭하고 DNS 서버의 IP 주소를 지정하여 전달 요청을 받습니다. 정방향 영역당 여러 개의 전달자를 지정할 수 있습니다.*



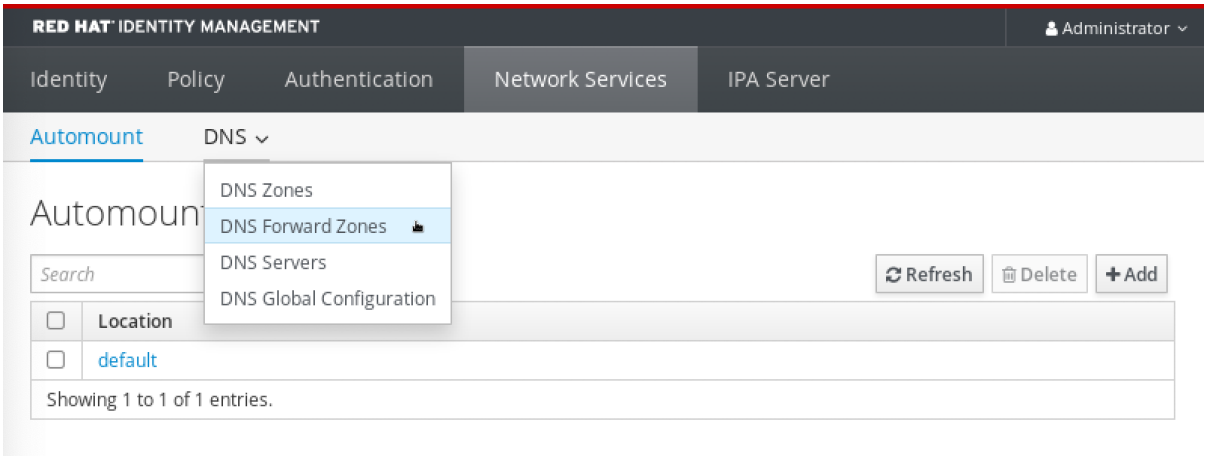
5. ***Forward policy*** *를 선택합니다.*



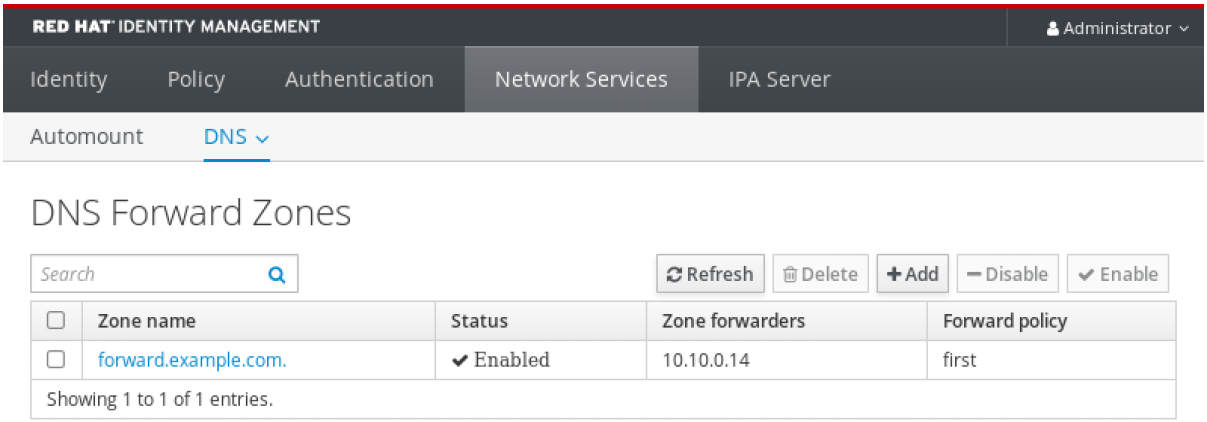
6. *창 하단에서* ***Add*** *(추가)를 클릭하여 새 전달 영역을 추가합니다.*

검증 단계

1. *IdM 웹 UI에서* ***네트워크 서비스 → DNS Forward Zones → DNS*** *를 선택합니다.*



2. *IdM 웹 UI에 지정한 forwarders 및 forward policy를 사용하여 생성한 정방향 영역이 있는지 확인합니다.*



6.6. CLI에서 DNS FORWARD ZONE 추가

이 섹션에서는 CLI(명령줄 인터페이스)에서 DNS 전달 영역을 추가하는 방법을 설명합니다.

중요

반드시 필요한 경우를 제외하고 전달 영역을 사용하지 마십시오. 전달 영역은 표준 솔루션이 아니며 이를 사용하면 예기치 않은 문제가 발생할 수 있습니다. 전달 영역을 사용해야 하는 경우 글로벌 전달 구성을 재정의하는 데 사용을 제한합니다.

새 DNS 영역을 생성할 때 Red Hat은 항상 네임서버(NS) 레코드를 사용하여 표준 DNS 위임을 사용하고 정방향 영역을 방지하는 것이 좋습니다. 대부분의 경우 글로벌 전달자를 사용하는 것으로 충분하며 전달 영역이 필요하지 않습니다.

사전 요구 사항

- *IdM 관리자로 로그인되어 있습니다.*
- *쿼리를 전달할 DNS 서버의 IP(Internet Protocol) 주소를 알고 있습니다.*

절차

- ***dnsforwardzone-add** 명령을 사용하여 새 전달 영역을 추가합니다. 전달 정책이 **없는** 경우 **--forwarder** 옵션으로 하나 이상의 전달자를 지정하고 **--forward-policy** 옵션으로 전달 정책을 지정합니다.*

■

```
[user@server ~]$ ipa dnsforwardzone-add forward.example.com. --
forwarder=10.10.0.14 --forwarder=10.10.1.15 --forward-policy=first

Zone name: forward.example.com.
Zone forwarders: 10.10.0.14, 10.10.1.15
Forward policy: first
```

검증 단계

- ***dnsforwardzone-show*** *명령을 사용하여 방금 생성한 DNS 전달 영역을 표시합니다.*

```
[user@server ~]$ ipa dnsforwardzone-show forward.example.com.

Zone name: forward.example.com.
Zone forwarders: 10.10.0.14, 10.10.1.15
Forward policy: first
```

6.7. ANSIBLE을 사용하여 IDM에서 DNS GLOBAL FORWARDER 설정

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS Global Forwarder를 설정하는 방법을 설명합니다.

아래 예제 절차에서 IdM 관리자는 포트 ***53*** *에서 인터넷 프로토콜(IP) v4 주소가* ***8.8.6.6*** *이고 IPv6 주소* ***2001:4860:4860::8800*** *이 있는 DNS 서버에 DNS를 생성합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*
- *IdM 관리자 암호를 알고 있습니다.*

절차

1. ***/usr/share/doc/ansible-freeipa/playbooks/dnsconfig*** *디렉터리로 이동합니다.*

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. *인벤토리 파일을 열고 구성하려는 IdM 서버가* ***[ipaserver]*** *섹션에 나열되어 있는지 확인합니다. 예를 들어,* ***server.idm.example.com*** *을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.*

```
[ipaserver]
server.idm.example.com
```

3. ***set-configuration.yml*** *Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.*

```
$ cp set-configuration.yml establish-global-forwarder.yml
```

4. *편집할* ***establish-global-forwarder.yml*** *파일을 엽니다.*
5. *다음 변수를 설정하여 파일을 조정합니다.*

a. ***IdM DNS에서 글로벌 전달자를 설정하도록 플레이북의 name 변수를 Playbook로*** *변경합니다.*

b. ***tasks*** *섹션에서* ***DNS 글로벌 전달자를 8.8.6.6 및 2001:4860:4860::8800으로 만들기 위해*** *작업* ***이름을*** *변경합니다.*

c. ***ipadnsconfig*** *부분의* ***forwarders*** *섹션에서 다음을 수행합니다.*

i. *첫 번째* ***ip_address*** *값을 글로벌 forwarder의 IPv4 주소로 변경합니다.* ***8.8.6.6***.

ii. *두 번째* ***ip_address*** *값을 global forwarder의 IPv6 주소로 변경합니다.* ***2001:4860:4860::8800***.

iii. ***포트*** *값이* ***53*** *으로 설정되어 있는지 확인합니다.*

d. ***forward_policy*** *를* ***먼저*** *로 변경합니다.*
이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to establish a global forwarder in IdM DNS
  hosts: ipaserver
  become: true

  tasks:
  - name: Create a DNS global forwarder to 8.8.6.6 and 2001:4860:4860::8800
    ipadnsconfig:
      forwarders:
        - ip_address: 8.8.6.6
        - ip_address: 2001:4860:4860::8800
          port: 53
      forward_policy: first
      allow_sync_ptr: yes
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file establish-global-forwarder.yml
```

추가 리소스

- ***/usr/share/doc/ansible-freeipa/*** *디렉토리의* ***README-dnsconfig.md*** *파일을 참조하십시오.*

6.8. ANSIBLE을 사용하여 IDM에 DNS 글로벌 전달자가 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS 글로벌 전달자가 있는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 포트 53에 있는 IP(Internet Protocol) v4 주소가 ***7.7.9.9*** *인 DNS 서버로 DNS 글로벌 전달자가 있는지, 포트* ***53*** *에서* ***2001:db8::1:0*** *의 IP v6 주소가 있는지 확인합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*

- *IdM 관리자 암호를 알고 있습니다.*

절차

1. ***/usr/share/doc/ansible-freeipa/playbooks/dnsconfig*** *디렉터리로 이동합니다.*

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. *인벤토리 파일을 열고 구성하려는 IdM 서버가* ***[ipaserver]*** *섹션에 나열되어 있는지 확인합니다. 예를 들어,* ***server.idm.example.com*** *을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.*

```
[ipaserver]
server.idm.example.com
```

3. ***forwarders-absent.yml*** *Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.*

```
$ cp forwarders-absent.yml ensure-presence-of-a-global-forwarder.yml
```

4. *편집할 수 있도록* ***ensure-presence-of-a-global-forwarder.yml*** *파일을 엽니다.*

5. *다음 변수를 설정하여 파일을 조정합니다.*

   a. ***IdM DNS에 글로벌 전달자가 있는지 확인하기 위해 플레이북의 name 변수를 Playbook로*** *변경합니다.*

   b. ***tasks*** *섹션에서 작업의* ***이름을*** *변경하여* ***DNS 글로벌 전달자가 포트 53의 7.7.9.9 및 2001:db8::1:0으로 있는지 확인합니다***.

   c. ***ipadnsconfig*** *부분의* ***forwarders*** *섹션에서 다음을 수행합니다.*

      i. *첫 번째* ***ip_address*** *값을 global forwarder:* ***7.7.9.9*** *의 IPv4 주소로 변경합니다.*

      ii. *두 번째* ***ip_address*** *값을 global forwarder:* ***2001:db8::1:0*** *의 IPv6 주소로 변경합니다.*

      iii. ***포트*** *값이* ***53*** *으로 설정되어 있는지 확인합니다.*

   d. ***state*** *를* ***present*** *로 변경합니다.*
   이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to ensure the presence of a global forwarder in IdM DNS
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure the presence of a DNS global forwarder to 7.7.9.9 and 2001:db8::1:0 on port
53
    ipadnsconfig:
      forwarders:
        - ip_address: 7.7.9.9
        - ip_address: 2001:db8::1:0
          port: 53
      state: present
```

6. *파일을 저장합니다.*

7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file ensure-presence-of-a-global-forwarder.yml
```

추가 리소스

- ***/usr/share/doc/ansible-freeipa/*** 디렉토리의 ***README-dnsconfig.md*** 파일을 참조하십시오.

6.9. ANSIBLE을 사용하여 IDM에 DNS 글로벌 전달자가 없는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS 글로벌 전달자가 없는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 인터넷 프로토콜(IP) v4 주소가 ***8.8.6.6*** 이고 포트 ***53*** 에서 ***2001:4860:4860::8800*** 의 IP v6 주소가 있는 DNS 글로벌 전달자가 없는지 확인합니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.
- IdM 관리자 암호를 알고 있습니다.

절차

1. ***/usr/share/doc/ansible-freeipa/playbooks/dnsconfig*** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 ***[ipaserver]*** 섹션에 나열되어 있는지 확인합니다. 예를 들어, ***server.idm.example.com*** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. ***forwarders-absent.yml*** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp forwarders-absent.yml ensure-absence-of-a-global-forwarder.yml
```

4. 편집할 수 있도록 ***ensure-absence-of-a-global-forwarder.yml*** 파일을 엽니다.

5. 다음 변수를 설정하여 파일을 조정합니다.

   a. ***IdM DNS에 글로벌 전달자가 없도록 플레이북의 name 변수를 Playbook로*** 변경합니다.

   b. ***tasks*** 섹션에서 작업의 ***이름을*** 변경하여 ***DNS 글로벌 전달자가 8.8.6.6 및 2001:4860:4860:4860::8800이 없는 것을 확인하도록 합니다***.

   c. ***ipadnsconfig*** 부분의 ***forwarders*** 섹션에서 다음을 수행합니다.

      i. 첫 번째 ***ip_address*** 값을 글로벌 forwarder의 IPv4 주소로 변경합니다. ***8.8.6.6***.

      ii. 두 번째 ***ip_address*** 값을 global forwarder의 IPv6 주소로 변경합니다. ***2001:4860:4860::8800***.

iii. *포트* 값이 **53** 으로 설정되어 있는지 확인합니다.

d. *작업* 변수를 **member** 로 설정합니다.

e. *상태가* **absent** 로 설정되어 있는지 확인합니다.

이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to ensure the absence of a global forwarder in IdM DNS
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure the absence of a DNS global forwarder to 8.8.6.6 and
2001:4860:4860::8800 on port 53
    ipadnsconfig:
      forwarders:
        - ip_address: 8.8.6.6
        - ip_address: 2001:4860:4860::8800
          port: 53
      action: member
      state: absent
```

중요

action: member *를 사용하지 않고 플레이북에서* **state: absent** *옵션만 사용하면 플레이북이 실패합니다.*

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file ensure-absence-of-a-global-forwarder.yml
```

추가 리소스

- **/usr/share/doc/ansible-freeipa/** *디렉토리의* **README-dnsconfig.md** *파일*
- *ipadnsconfig ansible-freeipa 모듈의* **action: member** *옵션*

6.10. ANSIBLE을 사용하여 IDM에서 DNS GLOBAL FORWARDERS가 비활성화되었는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS Global Forwarders가 비활성화되었는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 글로벌 전달자의 전달 정책이 **none** *으로 설정되어 전역 전달자를 효과적으로 비활성화합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*

- IdM 관리자 암호를 알고 있습니다.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. 모든 DNS 글로벌 전달자를 비활성화하도록 이미 구성된 **disable-global-forwarders.yml** Ansible 플레이북 파일의 내용을 확인합니다. 예를 들어 다음과 같습니다.

```
$ cat disable-global-forwarders.yml
---
- name: Playbook to disable global DNS forwarders
  hosts: ipaserver
  become: true

  tasks:
  - name: Disable global forwarders.
    ipadnsconfig:
      forward_policy: none
```

4. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file disable-global-forwarders.yml
```

추가 리소스

- **/usr/share/doc/ansible-freeipa/** 디렉토리의 **README-dnsconfig.md** 파일을 참조하십시오.

6.11. ANSIBLE을 사용하여 IDM에 DNS FORWARD ZONE이 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS Forward Zone이 있는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 **example.com** 의 DNS 전달 영역이 8. **8.8.8.8** 인 IP(Internet Protocol) 주소가 있는 DNS 서버로 있는지 확인합니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.
- IdM 관리자 암호를 알고 있습니다.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **forwarders-absent.yml** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp forwarders-absent.yml ensure-presence-forwardzone.yml
```

4. 편집할 수 있도록 **ensure-presence-forwardzone.yml** 파일을 엽니다.

5. 다음 변수를 설정하여 파일을 조정합니다.

   a. **IdM DNS에 dnsforwardzone이 있는지 확인하기 위해 플레이북의 name 변수를 Playbook로** 변경합니다.

   b. **tasks** 섹션에서 **example.com의 dnsforwardzone이 8.8.8.8로 있는지 확인하도록** 작업 **이름을** 변경합니다.

   c. **tasks** 섹션에서 **ipadnsconfig** 제목을 **ipadnsforwardzone** 로 변경합니다.

   d. **ipadnsforwardzone** 섹션에서 다음을 수행합니다.

      i. **ipaadmin_password** 변수를 추가하고 IdM 관리자 암호로 설정합니다.

      ii. **name** 변수를 추가하고 **example.com** 으로 설정합니다.

      iii. **전달자** 섹션에서 다음을 수행합니다.

         A. **ip_address** 및 **port** 행을 제거합니다.

         B. 대시 뒤에 지정하여 전달된 요청을 수신하도록 DNS 서버의 IP 주소를 추가합니다.

```
- 8.8.8.8
```

      iv. **forwardpolicy** 변수를 추가하고 **먼저** 로 설정합니다.

      v. **skip_overlap_check** 변수를 추가하고 **true** 로 설정합니다.

      vi. **state** 변수를 **present** 로 변경합니다.

   이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to ensure the presence of a dnsforwardzone in IdM DNS
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure the presence of a dnsforwardzone for example.com to 8.8.8.8
  ipadnsforwardzone:
     ipaadmin_password: password01
```

```
  name: example.com
  forwarders:
    - 8.8.8.8
  forwardpolicy: first
  skip_overlap_check: true
  state: present
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file ensure-presence-forwardzone.yml
```

추가 리소스

- ***/usr/share/doc/ansible-freeipa/*** *디렉토리에서* ***README-dnsforwardzone.md*** *파일을 참조하십시오.*

6.12. ANSIBLE을 사용하여 IDM에 DNS FORWARD ZONE이 여러 전달자가 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM의 DNS Forward Zone에 여러 전달자가 있는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 ***example.com*** *의 DNS 전달 영역이 8.* ***8.8 및 4.4.4. 4*** *로 전달되어 있는지 확인합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*
- *IdM 관리자 암호를 알고 있습니다.*

절차

1. ***/usr/share/doc/ansible-freeipa/playbooks/dnsconfig*** *디렉터리로 이동합니다.*

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. *인벤토리 파일을 열고 구성하려는 IdM 서버가* ***[ipaserver]*** *섹션에 나열되어 있는지 확인합니다. 예를 들어,* ***server.idm.example.com*** *을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.*

```
[ipaserver]
server.idm.example.com
```

3. ***forwarders-absent.yml*** *Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.*

```
$ cp forwarders-absent.yml ensure-presence-multiple-forwarders.yml
```

4. *편집할 수 있도록* ***ensure-presence-multiple-forwarders.yml*** *파일을 엽니다.*

5. *다음 변수를 설정하여 파일을 조정합니다.*

a. ***IdM DNS의 dnsforwardzone에 여러 개의 전달자가 있는지 확인하기 위해 플레이북의 name 변수를 Playbook로*** *변경합니다.*

b. ***tasks*** *섹션에서 dnsforwardzone의* ***dnsforwardzone의 8.8.8.8 및 4.4.4.4 forwarders가 있는지 확인하도록 작업 이름을 변경합니다.***

c. ***tasks*** *섹션에서* ***ipadnsconfig*** *제목을* ***ipadnsforwardzone*** *로 변경합니다.*

d. ***ipadnsforwardzone*** *섹션에서 다음을 수행합니다.*

i. ***ipaadmin_password*** *변수를 추가하고 IdM 관리자 암호로 설정합니다.*

ii. ***name*** *변수를 추가하고* ***example.com*** *으로 설정합니다.*

iii. ***전달자*** *섹션에서 다음을 수행합니다.*

A. ***ip_address*** *및* ***port*** *행을 제거합니다.*

B. *확인하려는 DNS 서버의 IP 주소를 대시 앞에 추가합니다.*

```
- 8.8.8.8
- 4.4.4.4
```

iv. *state 변수를 present로 변경합니다.*

이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: name: Playbook to ensure the presence of multiple forwarders in a dnsforwardzone
in IdM DNS
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure presence of 8.8.8.8 and 4.4.4.4 forwarders in dnsforwardzone for
example.com
  ipadnsforwardzone:
      ipaadmin_password: password01
     name: example.com
      forwarders:
         - 8.8.8.8
         - 4.4.4.4
      state: present
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file ensure-presence-multiple-forwarders.yml
```

추가 리소스

- ***/usr/share/doc/ansible-freeipa/*** *디렉토리에서* ***README-dnsforwardzone.md*** *파일을 참조하십시오.*

6.13. ANSIBLE을 사용하여 IDM에서 DNS FORWARD ZONE이 비활성화되었는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS Forward Zone이 비활성화되었는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 **example.com**의 DNS 전달 영역이 비활성화되었는지 확인합니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.
- IdM 관리자 암호를 알고 있습니다.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsconfig** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **forwarders-absent.yml** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp forwarders-absent.yml ensure-disabled-forwardzone.yml
```

4. 편집할 **ensure-disabled-forwardzone.yml** 파일을 엽니다.

5. 다음 변수를 설정하여 파일을 조정합니다.

   a. **IdM DNS에서 dnsforwardzone이 비활성화되도록 플레이북의 name 변수를 Playbook로** 변경합니다.

   b. **tasks** 섹션에서 **example.com의 dnsforwardzone이 비활성화되었는지 확인** 하도록 작업 **이름을** 변경합니다.

   c. **tasks** 섹션에서 **ipadnsconfig** 제목을 **ipadnsforwardzone** 로 변경합니다.

   d. **ipadnsforwardzone** 섹션에서 다음을 수행합니다.

      i. **ipaadmin_password** 변수를 추가하고 IdM 관리자 암호로 설정합니다.

      ii. **name** 변수를 추가하고 **example.com** 으로 설정합니다.

      iii. 전체 **전달자 섹션을 제거합니다**.

      iv. **state** 변수를 **disabled** 로 변경합니다.

   이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
```

```
- name: Playbook to ensure a dnsforwardzone is disabled in IdM DNS
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure a dnsforwardzone for example.com is disabled
  ipadnsforwardzone:
      ipaadmin_password: password01
      name: example.com
      state: disabled
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file ensure-disabled-forwardzone.yml
```

추가 리소스

- ***/usr/share/doc/ansible-freeipa/*** *디렉토리에서* ***README-dnsforwardzone.md*** *파일을 참조하십시오.*

6.14. ANSIBLE을 사용하여 IDM의 DNS 전달 영역이 없는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM에서 DNS Forward Zone이 없는지 확인하는 방법을 설명합니다. 아래 예제 절차에서 IdM 관리자는 ***example.com*** *의 DNS 전달 영역이 없음을 확인합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*

- *IdM 관리자 암호를 알고 있습니다.*

절차

1. ***/usr/share/doc/ansible-freeipa/playbooks/dnsconfig*** *디렉터리로 이동합니다.*

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsconfig
```

2. *인벤토리 파일을 열고 구성하려는 IdM 서버가* ***[ipaserver]*** *섹션에 나열되어 있는지 확인합니다. 예를 들어,* ***server.idm.example.com*** *을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.*

```
[ipaserver]
server.idm.example.com
```

3. ***forwarders-absent.yml*** *Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.*

```
$ cp forwarders-absent.yml ensure-absence-forwardzone.yml
```

4. *편집할 수 있도록* ***ensure-absence-forwardzone.yml*** *파일을 엽니다.*

5. *다음 변수를 설정하여 파일을 조정합니다.*

   a. ***IdM DNS에 dnsforwardzone이 없는 것을 확인하도록 플레이북의 name 변수를 Playbook 로 변경합니다.***

   b. ***tasks** 섹션에서 작업 **이름을 변경하여 example.com의 dnsforwardzone이 없는 것을 확인 합니다.***

   c. ***tasks** 섹션에서 **ipadnsconfig** 제목을 **ipadnsforwardzone** 로 변경합니다.*

   d. ***ipadnsforwardzone** 섹션에서 다음을 수행합니다.*

      i. ***ipaadmin_password** 변수를 추가하고 IdM 관리자 암호로 설정합니다.*

      ii. ***name** 변수를 추가하고 **example.com** 으로 설정합니다.*

      iii. *전체 **전달자 섹션을 제거합니다.***

      iv. ***state** 변수를 **absent** 로 둡니다.*

   이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Playbook to ensure the absence of a dnsforwardzone in IdM DNS
  hosts: ipaserver
  become: true

  tasks:
  - name: Ensure the absence of a dnsforwardzone for example.com
    ipadnsforwardzone:
      ipaadmin_password: password01
      name: example.com
      state: absent
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file ensure-absence-forwardzone.yml
```

추가 리소스

- ***/usr/share/doc/ansible-freeipa/** 디렉토리에서 **README-dnsforwardzone.md** 파일을 참조하십 시오.*

6.15. 추가 리소스

- *DNS 전달*

7장. IDM에서 DNS 레코드 관리

이 장에서는 IdM(Identity Management)에서 DNS 레코드를 관리하는 방법을 설명합니다. IdM 관리자는 IdM에서 DNS 레코드를 추가, 수정 및 삭제할 수 있습니다. 이 장에는 다음 섹션이 포함되어 있습니다.

- IdM의 DNS 레코드
- IdM 웹 UI에서 DNS 리소스 레코드 추가
- IdM CLI에서 DNS 리소스 레코드 추가
- 일반적인 ipa dnsrecord-add 옵션
- IdM 웹 UI에서 DNS 레코드 삭제
- IdM 웹 UI에서 전체 DNS 레코드 삭제
- IdM CLI에서 DNS 레코드 삭제

사전 요구 사항

- IdM 배포에 통합 DNS 서버가 포함되어 있습니다. 통합 DNS로 IdM을 설치하는 방법은 다음 링크 중 하나를 참조하십시오.
  - IdM 서버 설치: 통합 CA를 루트 CA로 통합 DNS와 함께 .
  - IdM 서버 설치: 루트 CA로 외부 CA를 사용하는 통합 DNS와 함께 .

7.1. IDM의 DNS 레코드

IdM(Identity Management)은 다양한 DNS 레코드 유형을 지원합니다. 다음 4개가 가장 자주 사용됩니다.

A

호스트 이름과 IPv4 주소에 대한 기본 맵입니다. A 레코드의 레코드 이름은 과 같은 호스트 ***이름입니다***. A 레코드의 ***IP 주소*** 값은 ***192.0.2.1*** 과 같은 IPv4 주소입니다.
A 레코드에 대한 자세한 내용은 RFC 1035 에서 참조하십시오.

AAAA

호스트 이름과 IPv6 주소에 대한 기본 맵입니다. AAAA 레코드의 레코드 이름은 과 같은 호스트 ***이름입니다***. ***IP 주소*** 값은 ***2001:DB8::1111*** 과 같은 IPv6 주소입니다.
AAAA 레코드에 대한 자세한 내용은 RFC 3596 을 참조하십시오.

SRV

서비스(SRV) 리소스 레코드 는 서비스 이름을 특정 서비스를 제공하는 서버의 DNS 이름에 매핑합니다. 예를 들어 이 레코드 유형은 LDAP 디렉터리와 같은 서비스를 관리하는 서버에 매핑할 수 있습니다. SRV 레코드의 레코드 이름은 ***_service._protocol(예: _ ldap._tcp***)입니다. SRV 레코드에 대한 구성 옵션에는 대상 서비스의 우선순위, 가중치, 포트 번호, 호스트 이름이 포함됩니다.

SRV 레코드에 대한 자세한 내용은 RFC 2782 를 참조하십시오.

PTR

포인터 레코드(PTR)는 IP 주소를 도메인 이름에 매핑하는 역방향 DNS 레코드를 추가합니다.

참고

IPv4 주소에 대한 역방향 DNS 조회는 모두 ***in-addr.arpa.*** *도메인에 정의된 역방향 항목을 사용합니다. 사람이 읽을 수 있는 형식의 역방향 주소는* ***in-addr.arpa.*** *도메인이 추가된 일반 IP 주소와 정확하게 역방향입니다. 예를 들어 네트워크 주소* ***192.0.2.0/24*** *의 경우 역방향 영역은* ***2.0.192.in-addr.arpa*** *입니다.*

PTR의 레코드 이름은 RFC 1035 에 지정된 표준 형식이어야 하며 RFC 2317 및 RFC 3596 에서 확장되어야 합니다. 호스트 이름 값은 레코드를 생성하려는 호스트의 정식 호스트 이름이어야 합니다.

참고

또한 ***.ip6.arpa.*** *도메인의 영역을 사용하여 IPv6 주소에 대해 역방향 영역을 구성할 수도 있습니다. IPv6 역방향 영역에 대한 자세한 내용은 RFC 3596 을 참조하십시오.*

DNS 리소스 레코드를 추가할 때 많은 레코드에 다른 데이터가 필요하다는 점에 유의하십시오. 예를 들어 CNAME 레코드에는 호스트 이름이 필요하지만 A 레코드에는 IP 주소가 필요합니다. IdM 웹 UI에서 현재 선택한 레코드 유형에 필요한 데이터를 반영하도록 새 레코드를 추가하는 양식의 필드가 자동으로 업데이트됩니다.

7.2. IDM 웹 UI에서 DNS 리소스 레코드 추가

이 섹션에서는 IdM(Identity Management) 웹 UI에 DNS 리소스 레코드를 추가하는 방법을 설명합니다.

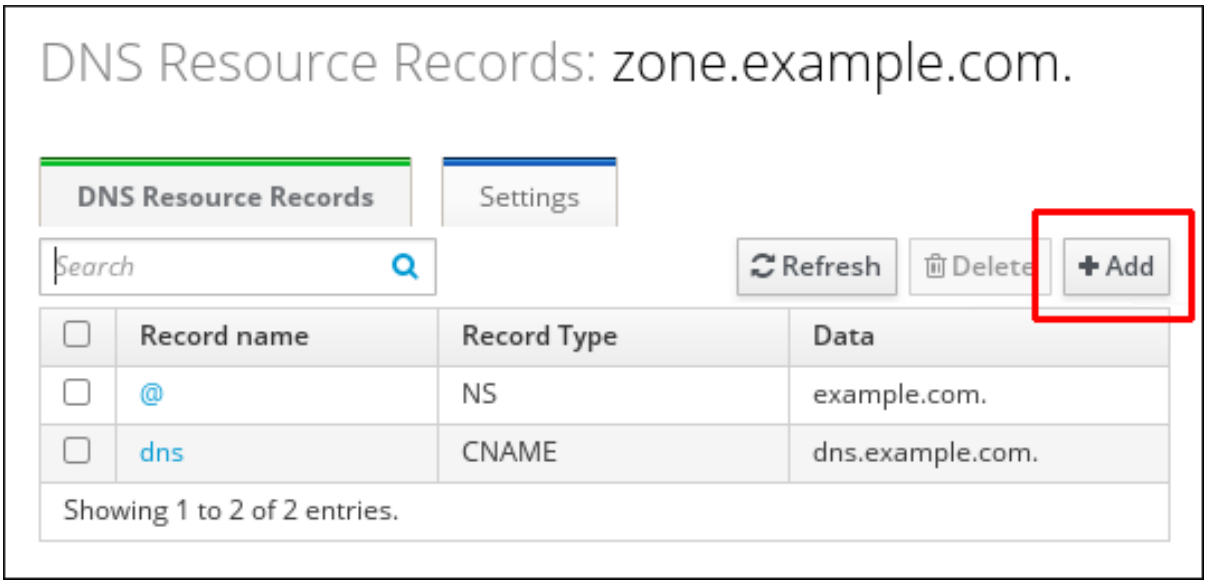
사전 요구 사항

- *DNS 레코드를 추가하려는 DNS 영역이 존재하며 IdM에서 관리합니다. IdM DNS에서 DNS 영역을 생성하는 방법에 대한 자세한 내용은 IdM의 DNS 영역 관리를 참조하십시오.*
- *IdM 관리자로 로그인되어 있습니다.*

절차

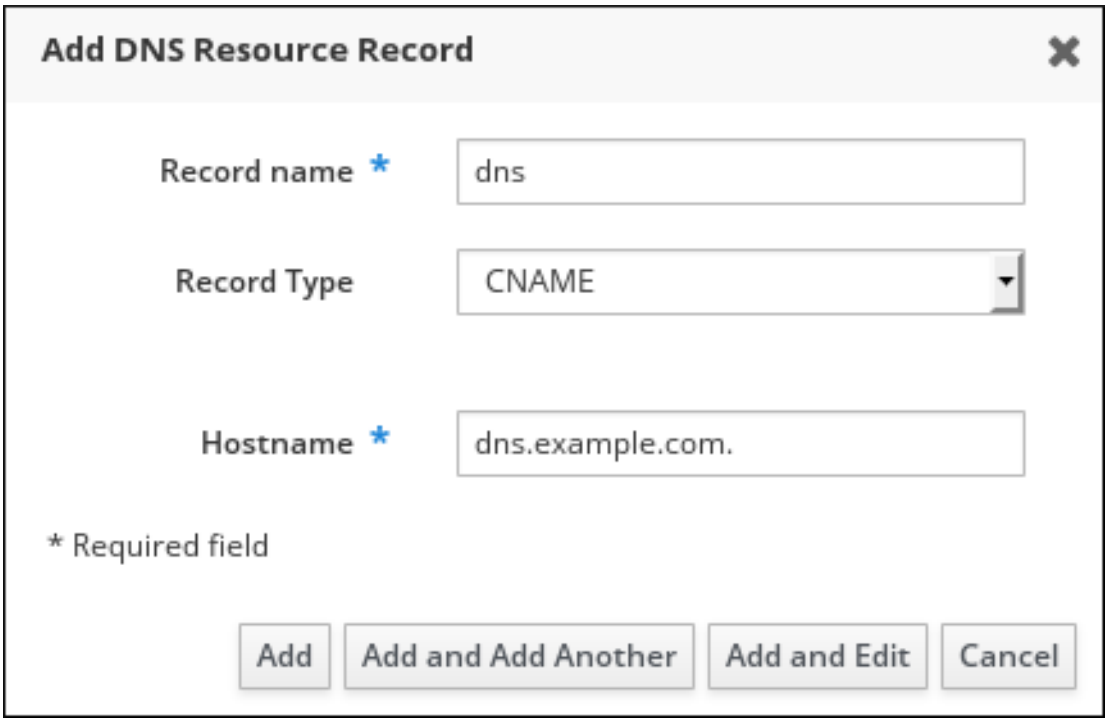
1. *IdM 웹 UI에서* ***네트워크 서비스 → DNS → DNS 영역을*** *클릭합니다.*
2. *DNS 레코드를 추가할 DNS 영역을 클릭합니다.*
3. ***DNS 리소스 레코드*** *섹션에서* ***추가를*** *클릭하여 새 레코드를 추가합니다.*

그림 7.1. 새 DNS 리소스 레코드 추가



4. *만들 레코드 유형을 선택하고 필요에 따라 다른 필드를 작성합니다.*

그림 7.2. 새 DNS 리소스 레코드 정의



5. ***Add**(추가)를 클릭하여 새 레코드를 확인합니다.*

7.3. IDM CLI에서 DNS 리소스 레코드 추가

이 섹션에서는 CLI(명령줄 인터페이스)에서 모든 유형의 DNS 리소스 레코드를 추가하는 방법을 설명합니다.

사전 요구 사항

- *DNS 레코드를 추가하려는 DNS 영역이 있습니다. IdM DNS에서 DNS 영역을 생성하는 방법에 대한 자세한 내용은 IdM의 DNS 영역 관리를 참조하십시오.*

- *IdM 관리자로 로그인되어 있습니다.*

절차

1. *DNS 리소스 레코드를 추가하려면* ***ipa dns records-add*** *명령을 사용합니다. 이 명령은 다음 구문을 따릅니다.*

```
$ ipa dnsrecord-add zone_name record_name --record_type_option=data
```

위의 명령에서:

- *zone_name 은 레코드를 추가할 DNS 영역의 이름입니다.*
- *record_name 은 새 DNS 리소스 레코드의 식별자입니다.*

예를 들어 ***host1*** *의 A 유형 DNS 레코드를* ***idm.example.com*** *영역에 추가하려면 다음을 입력합니다.*

```
$ ipa dnsrecord-add idm.example.com host1 --a-rec=192.168.122.123
```

7.4. 일반적인 IPA DNSRECORD- 옵션*

이 섹션에서는 IdM(Identity Management)에 가장 일반적인 DNS 리소스 레코드 유형을 추가, 수정 및 삭제할 때 사용할 수 있는 옵션에 대해 설명합니다.

- *A (IPv4)*
- *AAAA (IPv6)*
- *SRV*
- *PTR*

Bash *에서는* ***--tektonoption={val1,val2,val3}*** *와 같이 curly braces 내부의 쉼표로 구분된 목록에 값을 나열하여 여러 항목을 정의할 수 있습니다.*

표 7.1. 일반 레코드 옵션

옵션	*설명*
--ttl=*number*	레코드를 저장할 시간을 설정합니다.
--structured	원시 DNS 레코드를 구문 분석하고 구조화된 형식으로 반환합니다.

표 7.2. "A" 레코드 옵션

옵션	*설명*	*예제*
--a-rec=*ARECORD*	단일 A 레코드 또는 A 레코드 목록을 전달합니다.	**ipa dnsrecord-add idm.example.com host1 --a-rec=192.168.122.123**

옵션	설명	예제
	지정된 IP 주소를 사용하여 와일드카드 A 레코드를 만들 수 있습니다.	**ipa dnsrecord-add idm.example.com "*" --a-rec=192.168.122.123** [a]
--a-ip-address=*string*	레코드의 IP 주소를 지정합니다. 레코드를 만들 때 **A** 레코드 값을 지정하는 옵션은 **--a-rec** 입니다. 그러나 **A** 레코드를 수정할 때 **--a-rec** 옵션을 사용하여 **A** 레코드의 현재 값을 지정합니다. 새 값은 **--a-ip-address** 옵션으로 설정됩니다.	**ipa dnsrecord-mod idm.example.com --a-rec 192.168.122.123 --a-ip-address 192.168.122.124**

[a] 이 예제에서는 IP 주소가 192.0.2.123인 와일드카드 **A** 레코드를 생성합니다.

표 7.3. "AAAA" 레코드 옵션

옵션	설명	예제
--aaaa-rec=*AAAARECORD*	단일 AAAA(IPv6) 레코드 또는 AAAA 레코드 목록을 전달합니다.	**ipa dnsrecord-add idm.example.com www --aaaa-rec 2001:db8::1231:5675**
--aaaa-ip-address=*string*	레코드의 IPv6 주소를 제공합니다. 레코드를 만들 때 **A** 레코드 값을 지정하는 옵션은 **--aaa-rec** 입니다. 그러나 **A** 레코드를 수정할 때 **--aaaa-rec** 옵션은 **A** 레코드의 현재 값을 지정하는 데 사용됩니다. 새 값은 **--a-ip-address** 옵션으로 설정됩니다.	**IPA dnsrecord-mod idm.example.com --aaaa-rec 2001:db8::1231:5675 --aaaa-ip-address 2001:db8::1231:5676**

표 7.4. "PTR" 레코드 옵션

옵션	설명	예제
--ptr-rec=*PTRRECORD*	단일 PTR 레코드 또는 PTR 레코드 목록을 전달합니다. 역방향 DNS 레코드를 추가할 때 다른 DNS 레코드를 추가하는 사용과 비교하여 **ipa dnsrecord-add** 명령과 함께 사용되는 영역 이름이 리버스됩니다. 일반적으로 호스트 IP 주소는 지정된 네트워크에서 IP 주소의 마지막 노출입니다. 오른쪽에 있는 첫 번째 예제에서는 IPv4 주소가 **192.168.122.4인 server4.idm.example.com 의 PTR 레코드를 추가합니다.** 두 번째 예제에서는 reverse DNS 항목을 **0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa 에 추가합니다.** IP 주소가 2001:DB8::1111 인 호스트 server2.example.com 의 IPv6 역방향 영역입니다.	**ipa dnsrecord-add 122.168.192.in-addr.arpa 4 --ptr-rec server4.idm.example.com.** **$ ipa dnsrecord-add 0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa. 1.1.1.0.0.0.0.0.0.0.0.0.0.0.0.0 --ptr-rec server2.idm.example.com.**
--ptr-hostname=*string*	레코드의 호스트 이름을 지정합니다.	

표 7.5. "SRV" 레코드 옵션

옵션	설명	예제
--srv-rec=*SRVRECORD*	단일 SRV 레코드 또는 SRV 레코드 목록을 전달합니다. 오른쪽 예제에서 **_ldap._tcp** 는 SRV 레코드에 대한 서비스 유형 및 연결 프로토콜을 정의합니다. **srv-rec** 옵션은 priority, weight, port 및 target 값을 정의합니다. 예에서 51 및 49의 weight 값은 100까지 추가하고 특정 레코드가 사용되는 확률을 백분율로 나타냅니다.	**# ipa dnsrecord-add idm.example.com _ldap._tcp --srv-rec="0 51 389 server1.idm.example.com."** **# IPA dnsrecord-add server.idm.example.com _ldap._tcp --srv-rec="1 49 389 server2.idm.example.com"**
--srv-priority=*number*	레코드의 우선 순위를 설정합니다. 서비스 유형에 대한 SRV 레코드가 여러 개 있을 수 있습니다. 우선 순위(0 - 65535)는 레코드의 순위를 설정하며, 우선 순위가 낮을 수 있습니다. 서비스는 우선 순위가 가장 높은 레코드를 먼저 사용해야 합니다.	**# IPA dnsrecord-mod server.idm.example.com _ldap._tcp --srv-rec="1 49 389 server2.idm.example.com" --srv-priority=0**
--srv-weight=*number*	레코드의 가중치를 설정합니다. 동일한 우선 순위의 SRV 레코드 순서를 결정하는 데 도움이 됩니다. 설정 가중치는 100까지 추가해야 하며 특정 레코드가 사용되는 확률(분율)을 나타냅니다.	**# IPA dnsrecord-mod server.idm.example.com _ldap._tcp --srv-rec="0 49 389 server2.idm.example.com." --srv-weight=60**
--srv-port=*number*	대상 호스트에서 서비스의 포트를 지정합니다.	**# IPA dnsrecord-mod server.idm.example.com _ldap._tcp --srv-rec="0 60 389 server2.idm.example.com." --srv-port=636**
--srv-target=*string*	대상 호스트의 도메인 이름을 지정합니다. 도메인에서 서비스를 사용할 수 없는 경우 이 기간은 단일 기간(.)일 수 있습니다.	

추가 리소스

- ***ipa dnsrecord-add --help*** *를 실행합니다.*

7.5. IDM 웹 UI에서 DNS 레코드 삭제

이 섹션에서는 IdM 웹 UI를 사용하여 IdM(Identity Management)에서 DNS 레코드를 삭제하는 방법을 설명합니다.

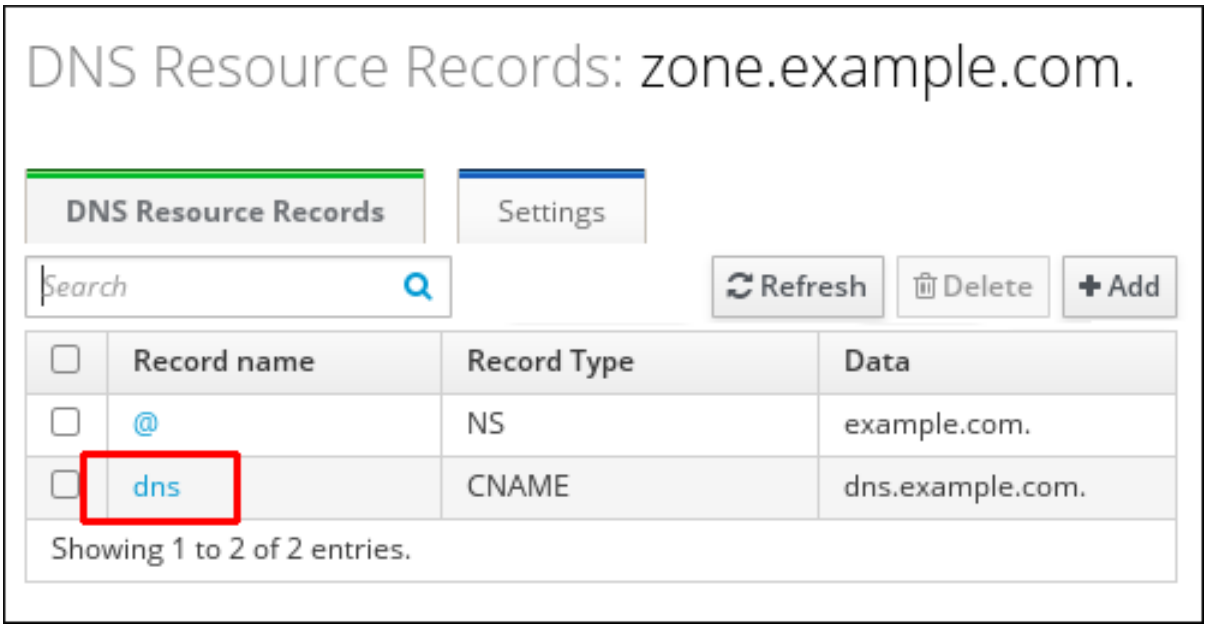
사전 요구 사항

- *IdM 관리자로 로그인되어 있습니다.*

절차

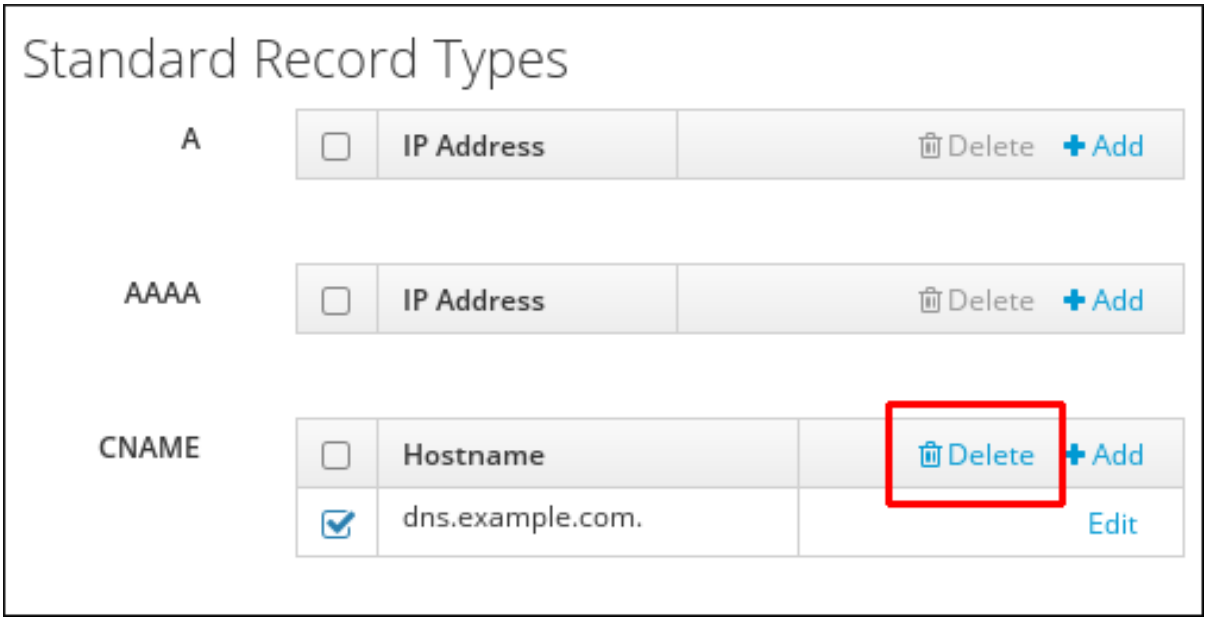
1. IdM 웹 UI에서 **네트워크 서비스 → DNS → DNS 영역**을 클릭합니다.
2. DNS 레코드를 삭제할 영역(예: **example.com**)을 클릭합니다.
3. **DNS 리소스 레코드** 섹션에서 리소스 레코드의 이름을 클릭합니다.

그림 7.3. DNS 리소스 레코드 선택



4. 삭제할 레코드 유형의 이름으로 확인란을 선택합니다.
5. **삭제**를 클릭합니다.

그림 7.4. DNS 리소스 레코드 삭제



선택한 레코드 유형이 삭제되었습니다. 리소스 레코드의 다른 구성은 그대로 유지됩니다.

추가 리소스

- IdM 웹 UI에서 전체 DNS 레코드 삭제를 참조하십시오.

7.6. IDM 웹 UI에서 전체 DNS 레코드 삭제

이 섹션에서는 IdM(Identity Management) 웹 UI를 사용하여 영역의 특정 리소스에 대한 모든 레코드를 삭제하는 방법을 설명합니다.

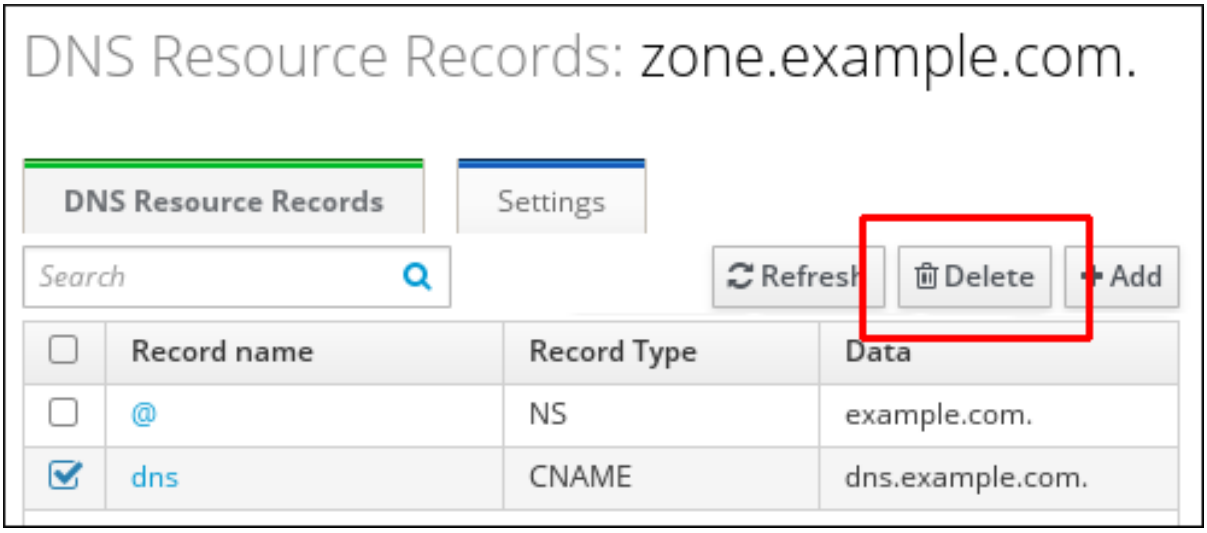
사전 요구 사항

- IdM 관리자로 로그인되어 있습니다.

절차

1. IdM 웹 UI에서 **네트워크 서비스 → DNS → DNS 영역**을 클릭합니다.
2. DNS 레코드를 삭제할 영역(예: **zone.example.com.**)을 클릭합니다.
3. **DNS 리소스 레코드** 섹션에서 삭제할 리소스 레코드의 확인란을 선택합니다.
4. **삭제**를 클릭합니다.

그림 7.5. 전체 리소스 레코드 삭제



이제 전체 리소스 레코드가 삭제되었습니다.

7.7. IDM CLI에서 DNS 레코드 삭제

이 섹션에서는 IdM(Identity Management) DNS에서 관리하는 영역에서 DNS 레코드를 제거하는 방법을 설명합니다.

사전 요구 사항

- IdM 관리자로 로그인되어 있습니다.

절차

- 영역에서 레코드를 제거하려면 **ipa dns records-del 명령을 사용하여 record** 값과 함께 **--recordType-rec** 옵션을 추가합니다. 예를 들어 A 유형 레코드를 제거하려면 다음을 수행합니다.

```
$ ipa dnsrecord-del example.com www --a-rec 192.0.2.1
```

*옵션 없이 **ipa dnsrecord-del** 을 실행하면 명령에서 삭제할 레코드에 대한 정보를 입력하라는 메시지를 표시합니다. 명령으로 **--del-all** 옵션을 전달하면 영역의 연관된 모든 레코드가 제거됩니다.*

추가 리소스

- ***ipa dns history-del --help** 명령을 실행합니다.*

7.8. 추가 리소스

- *Ansible을 사용하여 IdM의 DNS 레코드 관리를 참조하십시오.*

8 장. ANSIBLE 을 사용하여 IDM 의 DNS 레코드 관리

이 장에서는 Ansible 플레이북을 사용하여 IdM(Identity Management) 에서 DNS 레코드를 관리하는 방법을 설명합니다. IdM 관리자는 IdM 에서 DNS 레코드를 추가, 수정, 삭제할 수 있습니다. 이 장에는 다음 섹션이 포함되어 있습니다.

- Ansible 을 사용하여 IdM 에 A 및 AAAA DNS 레코드가 있는지 확인
- Ansible 을 사용하여 IdM 에 A 및 PTR DNS 레코드가 있는지 확인
- Ansible 을 사용하여 IdM 에 여러 DNS 레코드가 있는지 확인
- Ansible 을 사용하여 IdM 에 여러 CNAME 레코드가 있는지 확인
- Ansible 을 사용하여 IdM 에 SRV 레코드가 있는지 확인

8.1. IDM 의 DNS 레코드

IdM(Identity Management) 은 다양한 DNS 레코드 유형을 지원합니다. 다음 4 개가 가장 자주 사용됩니다.

A

호스트 이름과 IPv4 주소에 대한 기본 맵입니다. A 레코드의 레코드 이름은 과 같은 호스트 **이름입니다**. A 레코드의 **IP 주소** 값은 **192.0.2.1** 과 같은 IPv4 주소입니다.
A 레코드에 대한 자세한 내용은 RFC 1035 에서 참조하십시오.

AAAA

호스트 이름과 IPv6 주소에 대한 기본 맵입니다. AAAA 레코드의 레코드 이름은 과 같은 호스트 **이름입니다**. **IP 주소** 값은 **2001:DB8::1111** 과 같은 IPv6 주소입니다.
AAAA 레코드에 대한 자세한 내용은 RFC 3596 을 참조하십시오.

SRV

서비스(SRV) 리소스 레코드 는 서비스 이름을 특정 서비스를 제공하는 서버의 DNS 이름에 매핑합니다. 예를 들어 이 레코드 유형은 LDAP 디렉터리와 같은 서비스를 관리하는 서버에 매핑할 수 있습니다. SRV 레코드의 레코드 이름은 **_service._protocol(예: _ ldap._tcp)** 입니다. SRV 레코드에 대한 구성 옵션에는 대상 서비스의 우선순위, 가중치, 포트 번호, 호스트 이름이 포함됩니다.

SRV 레코드에 대한 자세한 내용은 RFC 2782 를 참조하십시오.

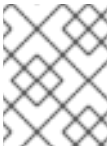
PTR

포인터 레코드(PTR) 는 IP 주소를 도메인 이름에 매핑하는 역방향 DNS 레코드를 추가합니다.

참고

IPv4 주소에 대한 역방향 DNS 조회는 모두 **in-addr.arpa.** 도메인에 정의된 역방향 항목을 사용합니다. 사람이 읽을 수 있는 형식의 역방향 주소는 **in-addr.arpa.** 도메인이 추가된 일반 IP 주소와 정확하게 역방향입니다. 예를 들어 네트워크 주소 **192.0.2.0/24** 의 경우 역방향 영역은 **2.0.192.in-addr.arpa** 입니다.

PTR 의 레코드 이름은 RFC 1035 에 지정된 표준 형식이어야 하며 RFC 2317 및 RFC 3596 에서 확장되어야 합니다. 호스트 이름 값은 레코드를 생성하려는 호스트의 정식 호스트 이름이어야 합니다.



참고

*또한 **.ip6.arpa.** 도메인의 영역을 사용하여 IPv6 주소에 대해 역방향 영역을 구성할 수도 있습니다. IPv6 역방향 영역에 대한 자세한 내용은 RFC 3596 을 참조하십시오.*

DNS 리소스 레코드를 추가할 때 많은 레코드에 다른 데이터가 필요하다는 점에 유의하십시오. 예를 들어 CNAME 레코드에는 호스트 이름이 필요하지만 A 레코드에는 IP 주소가 필요합니다. IdM 웹 UI에서 현재 선택한 레코드 유형에 필요한 데이터를 반영하도록 새 레코드를 추가하는 양식의 필드가 자동으로 업데이트됩니다.

8.2. 일반적인 IPA DNSRECORD-* 옵션

이 섹션에서는 IdM(Identity Management)에 가장 일반적인 DNS 리소스 레코드 유형을 추가, 수정 및 삭제할 때 사용할 수 있는 옵션에 대해 설명합니다.

- *A (IPv4)*
- *AAAA (IPv6)*
- *SRV*
- *PTR*

***Bash** 에서는 **--tektonoption={val1,val2,val3}** 와 같이 curly braces 내부의 쉼표로 구분된 목록에 값을 나열하여 여러 항목을 정의할 수 있습니다.*

표 8.1. 일반 레코드 옵션

옵션	*설명*
--ttl=*number*	레코드를 저장할 시간을 설정합니다.
--structured	원시 DNS 레코드를 구문 분석하고 구조화된 형식으로 반환합니다.

표 8.2. "A" 레코드 옵션

옵션	*설명*	*예제*
--a-rec=*ARECORD*	단일 A 레코드 또는 A 레코드 목록을 전달합니다.	**ipa dnsrecord-add idm.example.com host1 --a-rec=192.168.122.123**
	지정된 IP 주소를 사용하여 와일드카드 A 레코드를 만들 수 있습니다.	**ipa dnsrecord-add idm.example.com "*" --a-rec=192.168.122.123** [a]

옵션	*설명*	*예제*
--a-ip-address=*string*	레코드의 IP 주소를 지정합니다. 레코드를 만들 때 **A** 레코드 값을 지정하는 옵션은 **--a-rec** 입니다. 그러나 **A** 레코드를 수정할 때 **--a-rec** 옵션을 사용하여 **A** 레코드의 현재 값을 지정합니다. 새 값은 **--a-ip-address** 옵션으로 설정됩니다.	**ipa dnsrecord-mod idm.example.com --a-rec 192.168.122.123 --a-ip-address 192.168.122.124**
[a] 이 예제에서는 IP 주소가 192.0.2.123인 와일드카드 **A** 레코드를 생성합니다.		

표 8.3. "AAAA" 레코드 옵션

옵션	*설명*	*예제*
--aaaa-rec=*AAAARECORD*	단일 AAAA(IPv6) 레코드 또는 AAAA 레코드 목록을 전달합니다.	**ipa dnsrecord-add idm.example.com www --aaaa-rec 2001:db8::1231:5675**
--aaaa-ip-address=*string*	레코드의 IPv6 주소를 제공합니다. 레코드를 만들 때 **A** 레코드 값을 지정하는 옵션은 **--aaa-rec** 입니다. 그러나 **A** 레코드를 수정할 때 **--aaaa-rec** 옵션은 **A** 레코드의 현재 값을 지정하는 데 사용됩니다. 새 값은 **--a-ip-address** 옵션으로 설정됩니다.	**IPA dnsrecord-mod idm.example.com --aaaa-rec 2001:db8::1231:5675 --aaaa-ip-address 2001:db8::1231:5676**

표 8.4. "PTR" 레코드 옵션

옵션	*설명*	*예제*
--ptr-rec=*PTRRECORD*	단일 PTR 레코드 또는 PTR 레코드 목록을 전달합니다. 역방향 DNS 레코드를 추가할 때 다른 DNS 레코드를 추가하는 사용과 비교하여 **ipa dnsrecord-add** 명령과 함께 사용되는 영역 이름이 리버스됩니다. 일반적으로 호스트 IP 주소는 지정된 네트워크에서 IP 주소의 마지막 노출입니다. 오른쪽에 있는 첫 번째 예제에서는 IPv4 주소가 **192.168.122.4인 server4.idm.example.com 의 PTR 레코드를 추가합니다.** 두 번째 예제에서는 reverse DNS 항목을 **0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa 에 추가합니다.** IP 주소가 **2001:DB8::1111** 인 호스트 **server2.example.com** 의 IPv6 역방향 영역입니다.	**ipa dnsrecord-add 122.168.192.in-addr.arpa 4 --ptr-rec server4.idm.example.com.** **$ ipa dnsrecord-add 0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa. 1.1.1.0.0.0.0.0.0.0.0.0.0.0.0.0 --ptr-rec server2.idm.example.com.**
--ptr-hostname=*string*	레코드의 호스트 이름을 지정합니다.	

표 8.5. "SRV" 레코드 옵션

옵션	설명	예제
--srv-rec=*SRVRECORD*	단일 SRV 레코드 또는 SRV 레코드 목록을 전달합니다. 오른쪽 예제에서 _ldap._tcp 는 SRV 레코드에 대한 서비스 유형 및 연결 프로토콜을 정의합니다. **srv-rec** 옵션은 priority, weight, port 및 target 값을 정의합니다. 예에서 51 및 49의 weight 값은 100까지 추가하고 특정 레코드가 사용되는 확률을 백분율로 나타냅니다.	**# ipa dnsrecord-add idm.example.com _ldap._tcp --srv-rec="0 51 389 server1.idm.example.com."** **# IPA dnsrecord-add server.idm.example.com _ldap._tcp --srv-rec="1 49 389 server2.idm.example.com"**
--srv-priority=*number*	레코드의 우선 순위를 설정합니다. 서비스 유형에 대한 SRV 레코드가 여러 개 있을 수 있습니다. 우선 순위(0 - 65535)는 레코드의 순위를 설정하며, 우선 순위가 낮을 수 있습니다. 서비스는 우선 순위가 가장 높은 레코드를 먼저 사용해야 합니다.	**# IPA dnsrecord-mod server.idm.example.com _ldap._tcp --srv-rec="1 49 389 server2.idm.example.com" --srv-priority=0**
--srv-weight=*number*	레코드의 가중치를 설정합니다. 동일한 우선 순위의 SRV 레코드 순서를 결정하는 데 도움이 됩니다. 설정 가중치는 100까지 추가해야 하며 특정 레코드가 사용되는 확률(분율)을 나타냅니다.	**# IPA dnsrecord-mod server.idm.example.com _ldap._tcp --srv-rec="0 49 389 server2.idm.example.com." --srv-weight=60**
--srv-port=*number*	대상 호스트에서 서비스의 포트를 지정합니다.	**# IPA dnsrecord-mod server.idm.example.com _ldap._tcp --srv-rec="0 60 389 server2.idm.example.com." --srv-port=636**
--srv-target=*string*	대상 호스트의 도메인 이름을 지정합니다. 도메인에서 서비스를 사용할 수 없는 경우 이 기간은 단일 기간(.)일 수 있습니다.	

추가 리소스

- ***ipa dnsrecord-add --help*** *를 실행합니다.*

8.3. ANSIBLE을 사용하여 IDM에 A 및 AAAA DNS 레코드가 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 특정 IdM 호스트의 A 및 AAAA 레코드가 있는지 확인하는 방법을 보여줍니다. 아래 절차에 사용된 예제에서 IdM 관리자는 ***idm.example.com*** *DNS 영역에* ***host1*** *의 A 및 AAAA 레코드가 있는지 확인합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*
- *IdM 관리자 암호를 알고 있습니다.*

- **idm.example.com** 영역이 존재하며 IdM DNS에서 관리합니다. IdM DNS에서 기본 DNS 영역을 추가하는 방법에 대한 자세한 내용은 Ansible 플레이북을 사용하여 IdM DNS 영역 관리를 참조하십시오.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsrecord** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsrecord
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **ensure-A-and-AAAA-records-are-present.yml** Ansible 플레이북 파일을 복사합니다. 예를 들어 다음과 같습니다.

```
$ cp ensure-A-and-AAAA-records-are-present.yml ensure-A-and-AAAA-records-are-present-copy.yml
```

4. 편집을 위해 **ensure-A-and-AAAA- records-are-present-copy.yml** 파일을 엽니다.

5. **ipadnsrecord** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.

   - **ipaadmin_password** 변수를 IdM 관리자 암호로 설정합니다.
   - **zone_name** 변수를 **idm.example.com** 으로 설정합니다.
   - **records** 변수에서 **name** 변수를 **host1** 로 설정하고 **a_ip_address** 변수를 **192.168.122.123** 로 설정합니다.
   - **records** 변수에서 **name** 변수를 **host1** 로 설정하고 **aaaa_ip_address** 변수를 **::1** 로 설정합니다.
     이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Ensure A and AAAA records are present
  hosts: ipaserver
  become: true
  gather_facts: false

  tasks:
  # Ensure A and AAAA records are present
  - name: Ensure that 'host1' has A and AAAA records.
    ipadnsrecord:
      ipaadmin_password: Secret123
      zone_name: idm.example.com
      records:
      - name: host1
        a_ip_address: 192.168.122.123
      - name: host1
        aaaa_ip_address: ::1
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file ensure-A-and-AAAA-records-are-present-copy.yml
```

추가 리소스

- *IdM의 DNS 레코드를 참조하십시오.*
- */usr/share/doc/ansible -freeipa/ 디렉토리의 README-dns history.md 파일을 참조하십시오.*
- */usr/share/doc/ansible-freeipa/playbooks/dnsrecord 디렉터리의 샘플 Ansible 플레이북을 참조하십시오.*

8.4. ANSIBLE을 사용하여 IDM에 A 및 PTR DNS 레코드가 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 해당 PTR 레코드와 함께 특정 IdM 호스트에 대한 A 레코드가 있는지 확인하는 방법을 보여줍니다. 아래 절차에 사용된 예에서 IdM 관리자는 idm.example.com 영역에 IP 주소가 192.168.122.45 인 host1 의 A 및 PTR 레코드가 있는지 확인합니다.

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*
- *IdM 관리자 암호를 알고 있습니다.*
- *idm.example.com DNS 영역이 존재하며 IdM DNS에서 관리합니다. IdM DNS에서 기본 DNS 영역을 추가하는 방법에 대한 자세한 내용은 Ansible 플레이북을 사용하여 IdM DNS 영역 관리를 참조하십시오.*

절차

1. */usr/share/doc/ansible-freeipa/playbooks/dnsrecord 디렉터리로 이동합니다.*

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsrecord
```

2. *인벤토리 파일을 열고 구성하려는 IdM 서버가 [ipaserver] 섹션에 나열되어 있는지 확인합니다. 예를 들어, server.idm.example.com 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.*

```
[ipaserver]
server.idm.example.com
```

3. *ensure-dns history-with-reverse-is-present.yml Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.*

```
$ cp ensure-dnsrecord-with-reverse-is-present.yml ensure-dnsrecord-with-reverse-is-present-copy.yml
```

4. *편집을 위해 ensure-dns history-with-reverse-reverse-is-present-copy.yml 파일을 엽니다.*

5. ***ipadnsrecord*** *작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.*

   - ***ipaadmin_password*** *변수를 IdM 관리자 암호로 설정합니다.*
   - ***name*** *변수를* ***host1*** *로 설정합니다.*
   - ***zone_name*** *변수를* ***idm.example.com*** *으로 설정합니다.*
   - ***ip_address*** *변수를* ***192.168.122.45*** *로 설정합니다.*
   - ***create_reverse*** *변수를* ***yes*** *로 설정합니다.*
     이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Ensure DNS Record is present.
  hosts: ipaserver
  become: true
  gather_facts: false

  tasks:
  # Ensure that dns record is present
  - ipadnsrecord:
      ipaadmin_password: Secret123
      name: host1
      zone_name: idm.example.com
      ip_address: 192.168.122.45
      create_reverse: yes
      state: present
```

6. *파일을 저장합니다.*

7. *플레이북을 실행합니다.*

```
$ ansible-playbook -v -i inventory.file ensure-dnsrecord-with-reverse-is-present-copy.yml
```

추가 리소스

- *IdM의 DNS 레코드를 참조하십시오.*
- ***/usr/share/doc/ansible -freeipa/ 디렉토리의 README-dns history.md*** *파일을 참조하십시오.*
- ***/usr/share/doc/ansible-freeipa/playbooks/dnsrecord*** *디렉터리의 샘플 Ansible 플레이북을 참조하십시오.*

8.5. ANSIBLE을 사용하여 IDM에 여러 DNS 레코드가 있는지 확인

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 여러 값이 특정 IdM DNS 레코드에 연결되어 있는지 확인하는 방법을 보여줍니다. 아래 절차에 사용된 예에서 IdM 관리자는 ***idm.example.com*** *DNS 영역에* ***host1*** *에 대한 여러 A 레코드가 있는지 확인합니다.*

사전 요구 사항

- *Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.*

- IdM 관리자 암호를 알고 있습니다.
- **idm.example.com** 영역이 존재하며 IdM DNS에서 관리합니다. IdM DNS에서 기본 DNS 영역을 추가하는 방법에 대한 자세한 내용은 Ansible 플레이북을 사용하여 IdM DNS 영역 관리를 참조하십시오.

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsrecord** 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsrecord
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, **server.idm.example.com** 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. **ensure-presence-multiple-records.yml** Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp ensure-presence-multiple-records.yml ensure-presence-multiple-records-copy.yml
```

4. 편집할 수 있도록 **ensure-presence-multiple-records-copy.yml** 파일을 엽니다.
5. **ipadnsrecord** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.
   - **ipaadmin_password** 변수를 IdM 관리자 암호로 설정합니다.
   - **records** 섹션에서 **name** 변수를 **host1** 로 설정합니다.
   - **records** 섹션에서 **zone_name** 변수를 **idm.example.com** 으로 설정합니다.
   - **records** 섹션에서 **a_rec** 변수를 **192.168.122.112** 로 설정하고 **192.168.122.122** 로 설정합니다.
   - **records** 섹션에서 두 번째 레코드를 정의합니다.
     - **name** 변수를 **host1** 로 설정합니다.
     - **zone_name** 변수를 **idm.example.com** 으로 설정합니다.
     - **aaa_rec** 변수를 **::1** 로 설정합니다.

   이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Test multiple DNS Records are present.
  hosts: ipaserver
  become: true
  gather_facts: false

  tasks:
  # Ensure that multiple dns records are present
  - ipadnsrecord:
```

```
  ipaadmin_password: Secret123
  records:
    - name: host1
      zone_name: idm.example.com
      a_rec: 192.168.122.112
      a_rec: 192.168.122.122
    - name: host1
      zone_name: idm.example.com
      aaaa_rec: ::1
```

6. 파일을 저장합니다.

7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file ensure-presence-multiple-records-copy.yml
```

추가 리소스

- IdM의 DNS 레코드를 참조하십시오.
- **/usr/share/doc/ansible -freeipa/** 디렉토리의 **README-dns history.md** 파일을 참조하십시오.
- **/usr/share/doc/ansible-freeipa/playbooks/dnsrecord** 디렉터리의 샘플 Ansible 플레이북을 참조하십시오.

8.6. ANSIBLE을 사용하여 IDM에 여러 CNAME 레코드가 있는지 확인

Canonical Name 레코드(CNAME 레코드)는 하나의 도메인 이름인 별칭을 다른 이름인 다른 이름으로 매핑하는 DNS(Domain Name System)의 리소스 레코드 유형입니다.

단일 IP 주소에서 여러 서비스를 실행할 때 CNAME 레코드(예: FTP 서비스 및 웹 서비스)가 각각 다른 포트에서 실행되는 것을 확인할 수 있습니다.

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM DNS에 여러 CNAME 레코드가 있는지 확인하는 방법을 보여줍니다. 아래 절차에 사용된 예에서 **host03** 은 HTTP 서버와 FTP 서버입니다. IdM 관리자는 **idm.example.com** 영역에 **host03** A 레코드의 및 **ftp** CNAME 레코드가 있는지 확인합니다.

사전 요구 사항

- **Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.**
- **IdM 관리자 암호를 알고 있습니다.**
- **idm.example.com 영역이 존재하며 IdM DNS에서 관리합니다. IdM DNS에서 기본 DNS 영역을 추가하는 방법에 대한 자세한 내용은 Ansible 플레이북을 사용하여 IdM DNS 영역 관리를참조하십시오.**
- **host03 A 레코드는idm.example.com 영역에 있습니다.**

절차

1. **/usr/share/doc/ansible-freeipa/playbooks/dnsrecord 디렉터리로 이동합니다.**

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsrecord
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 **[ipaserver]** 섹션에 나열되어 있는지 확인합니다. 예를 들어, server.idm.example.com 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. ensure-CNAME-record-is-present.yml Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp ensure-CNAME-record-is-present.yml ensure-CNAME-record-is-present-copy.yml
```

4. 편집할 수 있도록 ensure-CNAME- records-is-present-copy.yml 파일을 엽니다.

5. **ipadnsrecord** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.

   - (선택 사항) 플레이 이름으로 제공한 설명을 지정합니다.
   - **ipaadmin_password** 변수를 IdM 관리자 암호로 설정합니다.
   - **zone_name** 변수를 idm.example.com 으로 설정합니다.
   - **records** 변수 섹션에서 다음 변수 및 값을 설정합니다.
     - **name** 변수를 www 로 설정합니다.
     - **cname_hostname** 변수를 host03 으로 설정합니다.
     - **name** 변수를 ftp 로 설정합니다.
     - **cname_hostname** 변수를 host03 으로 설정합니다.

   이는 현재 예제에서 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Ensure that 'www.idm.example.com' and 'ftp.idm.example.com' CNAME
records point to 'host03.idm.example.com'.
  hosts: ipaserver
  become: true
  gather_facts: false

  tasks:
  - ipadnsrecord:
      ipaadmin_password: Secret123
      zone_name: idm.example.com
      records:
      - name: www
        cname_hostname: host03
      - name: ftp
        cname_hostname: host03
```

6. 파일을 저장합니다.

7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file ensure-CNAME-record-is-present.yml
```

추가 리소스

- /usr/share/doc/ansible -freeipa/ 디렉토리의 README-dns history.md 파일을 참조하십시오.
- /usr/share/doc/ansible-freeipa/playbooks/dnsrecord 디렉터리의 샘플 Ansible 플레이북을 참조하십시오.

8.7. ANSIBLE을 사용하여 IDM에 SRV 레코드가 있는지 확인

DNS 서비스(SRV) 레코드는 도메인에서 사용할 수 있는 서비스의 호스트 이름, 포트 번호, 전송 프로토콜, 우선 순위 및 가중치를 정의합니다. IdM(Identity Management)에서 SRV 레코드를 사용하여 IdM 서버 및 복제본을 찾을 수 있습니다.

이 섹션에서는 IdM(Identity Management) 관리자가 Ansible 플레이북을 사용하여 IdM DNS에 SRV 레코드가 있는지 확인하는 방법을 보여줍니다. 아래 절차에 사용된 예에서 IdM 관리자는 _kerberos._udp.idm.example.com SRV 레코드가10 50 88 idm.example.com의 값으로 있는지 확인합니다. 이렇게 하면 다음 값이 설정됩니다.

- 서비스의 우선 순위를 10으로 설정합니다.
- 서비스의 가중치를 50으로 설정합니다.
- 서비스에서 사용할 포트를 88으로 설정합니다.

사전 요구 사항

- Ansible 컨트롤러에 ansible-freeipa 패키지를 설치했습니다. 프로세스의 단계를 실행하는 호스트입니다.
- IdM 관리자 암호를 알고 있습니다.
- idm.example.com 영역이 존재하며 IdM DNS에서 관리합니다. IdM DNS에서 기본 DNS 영역을 추가하는 방법에 대한 자세한 내용은 Ansible 플레이북을 사용하여 IdM DNS 영역 관리를참조하십시오.

절차

1. /usr/share/doc/ansible-freeipa/playbooks/dnsrecord 디렉터리로 이동합니다.

```
$ cd /usr/share/doc/ansible-freeipa/playbooks/dnsrecord
```

2. 인벤토리 파일을 열고 구성하려는 IdM 서버가 [ipaserver] 섹션에 나열되어 있는지 확인합니다. 예를 들어, server.idm.example.com 을 구성하도록 Ansible에 지시하려면 다음을 입력합니다.

```
[ipaserver]
server.idm.example.com
```

3. ensure-SRV- history-is-present.yml Ansible 플레이북 파일의 사본을 만듭니다. 예를 들어 다음과 같습니다.

```
$ cp ensure-SRV-record-is-present.yml ensure-SRV-record-is-present-copy.yml
```

4. 편집할 수 있도록 ensure-SRV- history-is-present-copy.yml 파일을 엽니다.

5. **ipadnsrecord** 작업 섹션에서 다음 변수를 설정하여 파일을 조정합니다.

   - **ipaadmin_password** 변수를 IdM 관리자 암호로 설정합니다.
   - **name** 변수를_kerberos._udp.idm.example.com 으로 설정합니다.
   - **srv_rec** 변수를'10 50 88 idm.example.com' 으로 설정합니다.
   - **zone_name** 변수를idm.example.com 으로 설정합니다.
     이는 현재 예에 대해 수정된 Ansible 플레이북 파일입니다.

```
---
- name: Test multiple DNS Records are present.
  hosts: ipaserver
  become: true
  gather_facts: false

  tasks:
  # Ensure a SRV record is present
  - ipadnsrecord:
      ipaadmin_password: Secret123
      name: _kerberos._udp.idm.example.com
      srv_rec: '10 50 88 idm.example.com'
      zone_name: idm.example.com
      state: present
```

6. 파일을 저장합니다.

7. 플레이북을 실행합니다.

```
$ ansible-playbook -v -i inventory.file ensure-SRV-record-is-present.yml
```

추가 리소스

- IdM의 DNS 레코드를 참조하십시오.
- **/usr/share/doc/ansible -freeipa/** 디렉토리의 **README-dns history.md** 파일을 참조하십시오.
- **/usr/share/doc/ansible-freeipa/playbooks/dnsrecord** 디렉터리의 샘플 Ansible 플레이북을 참조하십시오.

9 장. IDM 에서 정식 DNS 호스트 이름 사용

잠재적인 보안 위험을 방지하기 위해 DNS 표준화는 IdM(Identity Management) 클라이언트에서 기본적으로 비활성화되어 있습니다. 예를 들어 공격자가 도메인의 DNS 서버와 호스트를 제어하는 경우 공격자는 데모 와 같은 짧은 호스트 이름을 생성하여**malicious.example.com** 과 같은 손상된 호스트를 확인할 수 있습니다. 이 경우 사용자는 예상과는 다른 서버에 연결합니다.

이 섹션에서는 IdM 클라이언트에서 정식 호스트 이름을 사용하는 방법을 설명합니다.

9.1. 호스트 주체에 별칭 추가

기본적으로 **ipa-client-install** 명령을 사용하여 등록된 IdM(Identity Management) 클라이언트에서 서비스 주체에서 짧은 호스트 이름을 사용할 수 없습니다. 예를 들어, 사용자는 서비스에 액세스할 때 **host/demo@EXAMPLE.COM** 대신 host/demo.example.com@EXAMPLE.COM만 사용할 수 있습니다.

이 섹션에서는 Kerberos 주체에 별칭을 추가하는 방법을 설명합니다. 또는 **/etc/krb5.conf** 파일에서 호스트 이름을 표준화할 수 있습니다. 자세한 내용은 *클라이언트의 서비스 주체에서 호스트 이름 표준화*를 참조하십시오.

사전 요구 사항

- IdM 클라이언트가 설치되어 있습니다.
- 호스트 이름은 네트워크에서 고유합니다.

절차

1. **admin** 사용자로 IdM에 인증합니다.

```
$ kinit admin
```

2. 별칭을 호스트 주체에 추가합니다. 예를 들어 **demo.examle.com** 호스트 주체에 데모 별칭을 추가하려면 다음을 실행합니다.

```
$ ipa host-add-principal demo.example.com --principal=demo
```

9.2. 클라이언트의 서비스 주체의 호스트 이름 표준화 활성화

이 섹션에서는 클라이언트의 서비스 주체에서 호스트 이름 표준화를 활성화하는 방법을 설명합니다.

호스트 주체에 별칭 *추가에 설명된 대로 호스트 주체 별칭을*사용하는 경우 표준화를 활성화할 필요가 없습니다.

사전 요구 사항

- IdM(Identity Management) 클라이언트가 설치되어 있습니다.
- **root** 사용자로 IdM 클라이언트에 로그인되어 있습니다.
- 호스트 이름은 네트워크에서 고유합니다.

절차

1. */etc/krb5.conf* 파일의 *[libdefaults]* 섹션에서 *dns_canonicalize_hostname* 매개변수를 *false* 로 설정합니다.

```
[libdefaults]
...
dns_canonicalize_hostname = true
```

9.3. DNS 호스트 이름 정식화가 활성화된 호스트 이름 사용 옵션

클라이언트의 서비스 주체에서 호스트 이름 표준화에 설명된 대로/etc/krb5.conf 파일에서 dns_canonicalize_hostname = true 를 설정하면 서비스 주체에서 호스트 이름을 사용할 때 다음과 같은 옵션이 있습니다.

- *IdM(Identity Management) 환경에서는 호스트 /demo.example.com@EXAMPLE.COM와 같은 서비스 주체에서 전체 호스트 이름을 사용할 수 있습니다.*
- *IdM이 없는 환경에서는 RHEL 호스트가 AD(Active Directory) 도메인의 구성원으로 된 경우 AD 도메인 컨트롤러(DC)가 AD에 등록된 머신의 sysfs 이름에 대한 서비스 주체를 자동으로 생성하기 때문에 추가 고려 사항이 필요하지 않습니다.*