Red Hat OpenStack Platform 17.1

DNS를 서비스로 구성

Red Hat OpenStack Platform의 DNS 서비스를 사용하여 DNS(Domain Name System)를 관리하는 방법에 대한 정보

Last Updated: 2024-09-26

Red Hat OpenStack Platform 17.1 DNS를 서비스로 구성

Red Hat OpenStack Platform의 DNS 서비스를 사용하여 DNS(Domain Name System)를 관리하는 방법에 대한 정보

OpenStack Team
rhos-docs@redhat.com

법적 공지

초록

RHOSP DNS 서비스(designate)를 설치, 구성, 운영, 문제 해결 및 업그레이드하기 위한 가이드입니다.

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.

RED HAT 문서에 관한 피드백 제공

문서 개선을 위한 의견을 보내 주십시오. Red Hat이 어떻게 더 나은지 알려주십시오.

Jira에서 문서 피드백 제공

문제 생성 양식을 사용하여 문서에 대한 피드백을 제공합니다. Jira 문제는 Red Hat OpenStack Platform Jira 프로젝트에서 생성되어 피드백의 진행 상황을 추적할 수 있습니다.

1. Jira에 로그인했는지 확인합니다. Jira 계정이 없는 경우 피드백을 제출할 계정을 생성합니다.
2. 다음 링크를 클릭하여 **문제 생성** 페이지를 엽니다.
   https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12336920&summary=Documentation%20feedback:%20%3CAdd%20summary%20here%3E&is
   <Include+the+documentation+URL,+the%20chapter+or+section+number,+and+a+detailed+descrip
3. **요약** 및 **설명** 필드를 작성합니다. **설명** 필드에 문서 URL, 장 또는 섹션 번호, 문제에 대한 자세한 설명을 포함합니다. 양식의 다른 필드를 수정하지 마십시오.
4. **생성**을 클릭합니다.

1장. DNS 서비스 소개

DNS 서비스(designate)는 RHOSP(Red Hat OpenStack Platform) 배포를 위한 DNS-as-a-Service 구현을 제공합니다.

이 섹션에서는 몇 가지 DNS(Domain Name System) 기본 사항을 간략하게 설명하고, DNS 서비스 구성 요소를 설명하고, 간단한 사용 사례를 제공하며, DNS 서비스를 실행하는 다양한 방법을 나열합니다.

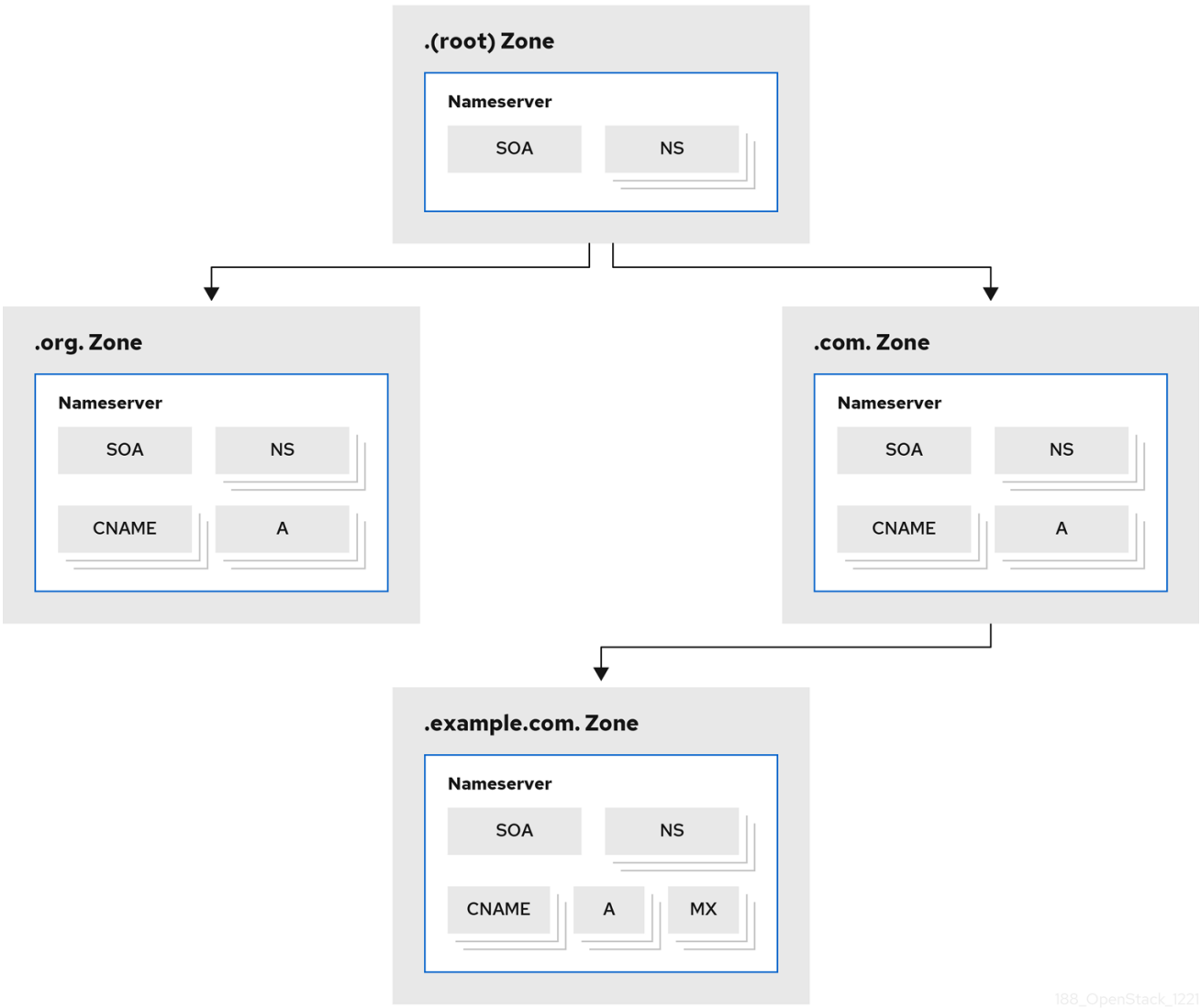
이 섹션에 포함된 항목은 다음과 같습니다.

1.1. DNS(DOMAIN NAME SYSTEM)의 기본 사항

DNS(Domain Name System)는 사설 또는 공용 네트워크에 연결된 리소스의 이름 지정 시스템입니다. 계층적 분산 데이터베이스인 DNS는 *영역*이라는 다양한 그룹으로 구성된 도메인 이름과 리소스에 대한 정보를 연결합니다. 권한 있는 이름 서버는 확인자가 네트워크 데이터를 라우팅하기 위한 리소스를 식별하고 찾을 수 있는 레코드에 리소스 및 영역 정보를 저장합니다.

이름은 위임을 용이하게 하는 영역의 계층 구조로 나뉩니다. 별도의 이름 서버는 특정 영역을 담당합니다.

그림 1.1. 도메인 이름 시스템



간단히 . (점)인 루트 영역에는 다양한 최상위 도메인(TLD)을 다른 이름 서버에 위임하는 레코드가 포함되어 있습니다. 이러한 유형의 레코드는 이름 서버(NS) 레코드라고 하며 특정 도메인에 대해 권한이 있는 DNS 서버를 식별합니다. 도메인의 주 및 백업 이름 서버를 나타내는 NS 레코드가 두 개 이상 있는 것은 드문 일이 아닙니다.

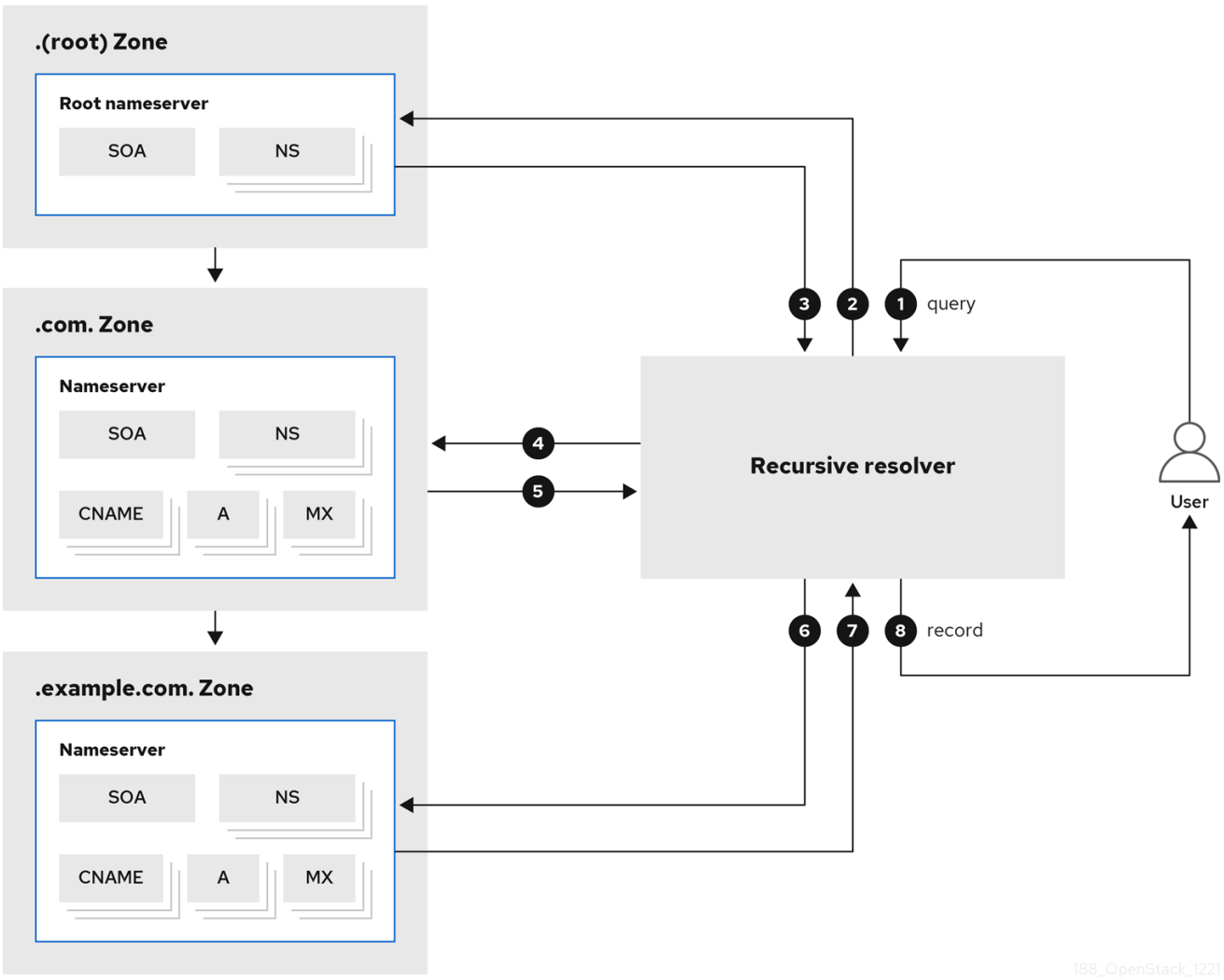
루트 영역 아래에는 TLD 내에서만 도메인에 대한 레코드가 포함된 다양한 TLD 이름 서버가 있습니다. 이는 주소 레코드 및 표준 이름 레코드이며 각각 *A* 및 *CNAME* 레코드라고 합니다.

예를 들어 **.com** 이름 서버에는 **example.com** 의 CNAME 레코드와 다른 이름 서버에 영역을 위임하는 NS 레코드가 포함되어 있습니다. 도메인 **example.com** 에는 자체 이름 서버가 있을 수 있으므로 **cloud.example.com** 과 같은 다른 도메인을 생성할 수 있습니다.

확인자는 일반적으로 두 부분으로 구성됩니다. 일반적으로 사용자 컴퓨터의 라이브러리인 *스텁* 확인자 및 사용자에게 결과를 반환하기 전에 이름 서버에 대한 쿼리를 수행하는 *재귀 확인* 자입니다. 도메인을 검색할 때 확인자는 도메인 끝에 시작하여 도메인 시작 방향으로 작동합니다.

예를 들어 **cloud.example.com** 을 검색할 때 확인자는 루트 이름 서버로 시작합니다 . 루트는 **.com** 이름 서버의 위치로 응답합니다. 그런 다음 확인자가 **.com** 이름 서버에 연결하여 **example.com** 이름 서버를 가져옵니다. 마지막으로 확인자는 **cloud.example.com** 레코드를 찾아서 사용자에게 반환합니다.

그림 1.2. DNS 쿼리 확인



1	사용자가 **cloud.example.com** 주소를 쿼리합니다.
2	재귀 확인자는 **cloud.example.com** 의 **루트** 영역 이름 서버를 쿼리합니다.
3	레코드를 찾을 수 없으며 **루트** 영역은 **.com** 의 이름 서버를 제공합니다.
4	확인자는 **cloud.example .com 의 .** com 이름 서버를 쿼리합니다.
5	레코드를 찾을 수 없으며 **.com** 영역은 **example.com** 의 이름 서버를 제공합니다.
6	확인자는 **cloud. example.com 의 example.** com 이름 서버를 쿼리합니다.
7	**example.com** 이름 서버는 **cloud.example.com** 을 찾고 **cloud.example.com** 의 A 레코드를 확인자에게 제공합니다.
8	확인자는 **cloud.example.com** 의 A 레코드를 사용자에게 전달합니다.

이 검색을 보다 효율적으로 수행하기 위해 결과는 확인자에서 캐시되므로 첫 번째 사용자가 **cloud.example.com** 을 요청한 후 확인자는 후속 요청에 대해 캐시된 결과를 신속하게 반환할 수 있습니다.

추가 리소스

- https://en.wikipedia.org/wiki/Domain_Name_System
- https://tools.ietf.org/html/rfc1034
- 1.2절. "RHOSP DNS 서비스 소개"

1.2. RHOSP DNS 서비스 소개

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)는 DNS 레코드, 이름 및 영역을 관리할 수 있는 멀티 테넌트 서비스입니다. RHOSP DNS 서비스는 REST API를 제공하며 사용자 관리를 위해 RHOSP Identity 서비스(keystone)와 통합됩니다.

RHOSP director를 사용하여 DNS 레코드를 포함하도록 BIND 인스턴스를 배포하거나 DNS 서비스를 기존 BIND 인프라에 통합할 수 있습니다. 또한 director는 DNS 서비스 통합을 RHOSP Networking 서비스(neutron)와 구성하여 컴퓨팅 인스턴스, 네트워크 포트 및 유동 IP에 대한 레코드를 자동으로 생성할 수 있습니다.

추가 리소스

- 1.1절. "DNS(Domain Name System)의 기본 사항"
- 1.3절. "DNS 서비스 구성 요소"

1.3. DNS 서비스 구성 요소

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)는 기본적으로 하나 이상의 RHOSP 컨트롤러 호스트의 컨테이너에서 실행되는 여러 다른 서비스로 구성됩니다.

API (designate-api 컨테이너) 지정

사용자와 RHOSP Networking 서비스(neutron)가 designate와 상호 작용할 수 있도록 OpenStack 표준 REST API를 제공합니다. API는 RPC(원격 프로시저 호출)를 통해 중앙 서비스에 요청을 전송하여 요청을 처리합니다.

생산자(designate-producer 컨테이너)

지정에 따라 실행되는 주기적인 작업을 오케스트레이션합니다. 이러한 작업은 장기 실행 중이며, Ceilometer에 대해 **dns.zone.exists** 를 발송하고 데이터베이스에서 삭제된 영역을 제거하고, 새로 고침 간격으로 보조 영역을 폴링하고, 지연된 **NOTIFY** 트랜잭션을 생성하고, 오류 상태에서 영역을 주기적으로 복구하는 등 잠재적으로 대규모 작업입니다.

중앙 (지정-중앙 컨테이너)

영역 및 레코드 세트 생성, 업데이트 및 삭제를 오케스트레이션합니다. 중앙 서비스는 Designate API 서비스에서 보낸 RPC 요청을 수신하고 영구 스토리지를 조정하는 동안 필요한 비즈니스 논리를 데이터에 적용합니다.

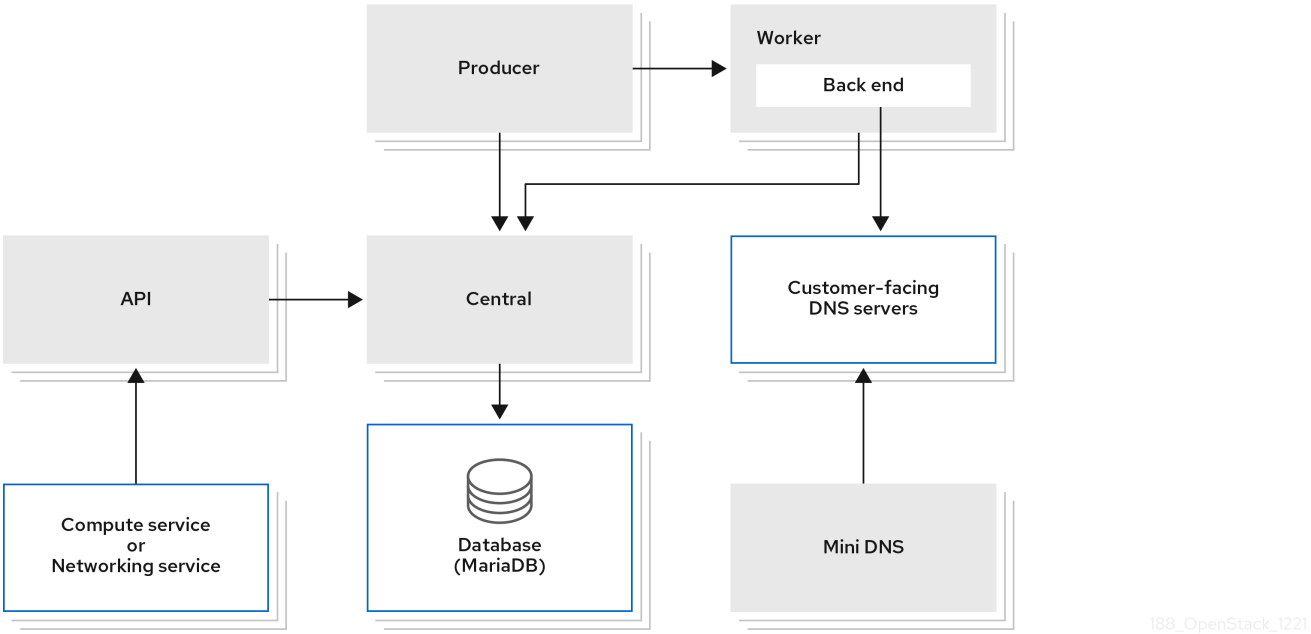
작업자(지정 작업 컨테이너)

을 지정하는 DNS 서버의 드라이버에 인터페이스를 제공합니다. Worker 서비스는 designate 데이터베이스에서 서버 구성을 읽고 Producer에서 요청하는 주기적인 작업도 관리합니다.

Mini DNS (designate-mdns 컨테이너)

이름 서버에서 AXFR(권한 있는 전송) 요청을 관리합니다. Mini DNS 서비스는 지정 인프라 외부에서 호스팅되는 DNS 영역에 대한 DNS 정보도 가져옵니다.

그림 1.3. DNS 서비스 아키텍처



RHOSP에서 기본적으로 DNS 구성 요소는 BIND 9 및 unbound입니다.

BIND 9(바인 컨테이너바인딩)

DNS 서비스에 대한 DNS 서버를 제공합니다. BIND는 DNS 소프트웨어의 오픈 소스 제품군이며, 특히 권한 있는 이름 서버 역할을 합니다.

바인딩되지 않은 컨테이너(바인딩되지 않은 컨테이너)

DNS 재귀 확인기의 역할을 수행하며, 이는 DNS 요청을 IP 주소로 변환하는 데 필요한 쿼리를 시작하고 정렬합니다. unbound는 DNS 서비스가 재귀 확인기로 사용하는 오픈 소스 프로그램입니다.

DNS 서비스는 oslo 호환 데이터베이스를 사용하여 서비스 간 통신을 용이하게 하기 위해 데이터 및 oslo 메시징을 저장합니다. API 프로세스는 로드 밸런서 뒤에 있는 종종 API 프로세스를 통해 고가용성 배포를 용이하게 하기 위해 DNS 서비스의 여러 인스턴스를 실행할 수 있습니다.

추가 리소스

- 1.4절. "DNS 서비스의 일반적인 배포 시나리오"
- 1.5절. "DNS 서비스를 사용하는 다양한 방법"

1.4. DNS 서비스의 일반적인 배포 시나리오

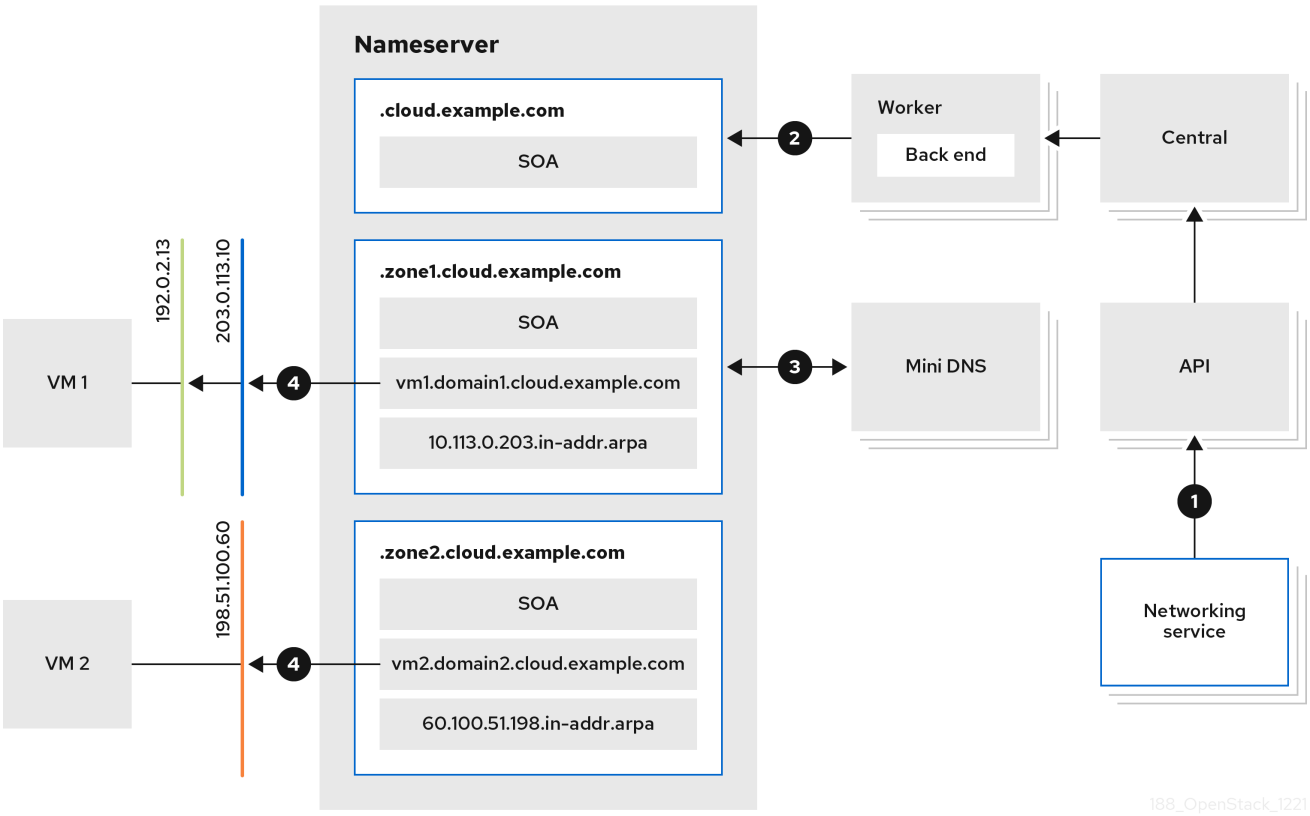
사용자가 **zone1.cloud.example.com** 및 **zone2.cloud.example.com** 이라는 두 개의 영역을 생성했으며 DNS 서비스는 DNS 이름 서버에서 각각 새 영역의 새 SOA(시작 기관) 레코드와 새 이름 서버(NS) 레코드를 추가합니다.

RHOSP Networking 서비스를 사용하여 사용자는 개인 네트워크를 생성하여 **zone1** 및 public 네트워크에 연결하고 이를 **zone2** 에 연결합니다.

마지막으로 사용자는 VM 인스턴스를 사설 네트워크에 연결하고 유동 IP를 연결합니다. 사용자는 두 번째 인스턴스를 공용 네트워크에 직접 연결합니다. 이러한 연결은 네트워킹 서비스를 트리거하여 사용자를 대

신하여 레코드를 생성하도록 DNS 서비스를 요청합니다. DNS 서비스는 인스턴스 이름을 권한 있는 이름 서버의 도메인에 매핑하고 PTR 레코드를 생성하여 역방향 조회를 활성화합니다.

그림 1.4. 일반적인 DNS 서비스 배포



1	RHOSP Networking 서비스의 유동 IP, 포트 및 네트워크와 도메인 및 이름을 연결할 수 있습니다. RHOSP Networking 서비스는 designate API를 사용하여 포트가 생성되고 삭제될 때 레코드를 관리합니다.
2	designate Worker는 이름 서버에 영역 정보를 업데이트하도록 지시합니다.
3	이름 서버는 Mini DNS에서 업데이트된 영역 정보를 요청합니다.
4	이름 서버는 정방향 및 역방향 레코드를 모두 생성합니다.

추가 리소스

- 1.2절. “RHOSP DNS 서비스 소개”
- 1.3절. “DNS 서비스 구성 요소”

1.5. DNS 서비스를 사용하는 다양한 방법

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)는 REST API를 제공하며 일반적으로 다음 세 가지 방법으로 사용됩니다.

- 가장 일반적인 방법은 RHOSP 서비스와 상호 작용하는 명령과 함께 python 명령줄 툴인 RHOSP OpenStack 클라이언트를 사용하는 것입니다.

- 그래픽 사용자 인터페이스인 RHOSP 대시보드(horizon)를 통해 DNS 서비스를 사용할 수도 있습니다.
- 개발자는 OpenStack SDK를 사용하여 애플리케이션을 작성할 수 있습니다. 자세한 내용은 openstacksdk 를 참조하십시오.

2장. DNS 서비스 배포 계획

이 섹션에서는 Red Hat OpenStack Platform을 사용하여 DNS 서비스(designate) 배포를 계획할 때 고려해야 할 사항에 대해 설명합니다.

이 섹션에 포함된 항목은 다음과 같습니다.

2.1. DNS 서버 기능 지원 매트릭스

다음 표에는 RHOSP(Red Hat OpenStack Platform) 17에서 지원하는 DNS 서비스(designate)의 기능이 나열되어 있습니다.

표 2.1. DNS 서비스(designate) 기능 지원 매트릭스

기능	RHOSP 17에서 지원됩니까?
x86_64 하드웨어 아키텍처	제공됨
기타 모든 하드웨어 아키텍처	없음
BIND 9 백엔드	제공됨
기타 모든 백엔드	없음
Denylists (blacklists)	제공됨
v1 API 지정	없음
v2 API 지정	제공됨
관리자 API 지정	없음
중앙 서비스 지정	제공됨
Producer 서비스 지정	제공됨
작업자 서비스 지정	제공됨
MiniDNS 서비스 지정	제공됨

에이전트 서비스 지정	없음
Zone Manager 서비스 지정	없음
Pool Manager 서비스 지정	없음
OpenStack 클라이언트 플러그인 지정(CLI)	제공됨
클라이언트 (CLI) 지정	없음
OpenStack Python SDK (designate)	제공됨
클라이언트 지정 (SDK)	없음
지평선 대시보드 지정	제공됨
임시 플러그인 지정	제공됨
데이터베이스 MariaDB/Galera 지정	제공됨
기타 모든 데이터베이스	없음
분산 잠금 관리자(Redis)	제공됨
기타 모든 분산 잠금 관리자 옵션	없음
싱크 지정	없음
알림 지정	제공됨
고가용성 배포	제공됨
IPv4	제공됨
IPv6	제공됨
Monasca 통합	없음
기본 풀 스케줄러	제공됨
기타 모든 풀 스케줄러	없음
단일 풀	제공됨
여러 풀	없음
할당량	제공됨

RBAC(역할 기반 액세스 제어)	제공됨
레코드 유형 A	제공됨
레코드 유형 AAAA	제공됨
레코드 유형 CNAME	제공됨
레코드 유형 MX	제공됨
레코드 유형 SRV	제공됨
레코드 유형 TXT	제공됨
레코드 유형 SPF	제공됨
레코드 유형 NS	제공됨
레코드 유형 PTR	제공됨
레코드 유형 SSHFP	제공됨
레코드 유형 SOA	제공됨
레코드 유형 NAPTR	제공됨
레코드 유형 CAA	제공됨
다른 모든 레코드 유형	없음
최상위 도메인(TLD)	제공됨
TSIG 키	제공됨
바인딩되지 않은 재귀 해결기	제공됨
기타 모든 재귀 해결 방법	없음
기본 영역	제공됨
보조 영역	없음
영역 가져오기 및 내보내기	제공됨
영역 중단	없음
영역 소유권 전송	제공됨

2.2. DNS 서비스 소프트웨어 요구 사항

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)는 다음 RHOSP 핵심 구성 요소에 따라 다릅니다.

- Identity 서비스(keystone)
- RabbitMQ
- MariaDB
- Redis

RHOSP 설치 및 설정 툴셋인 director는 DNS 서비스에 대해 이러한 구성 요소를 자동으로 구성합니다.

DNS 서비스에서 DNS 레코드를 자동으로 생성할 VLAN 또는 오버레이 네트워크를 사용하는 경우 이러한 네트워크에 대한 일부 네트워크 분할 ID를 별도로 설정합니다. DNS 서비스는 분할 ID가 네트워킹 서비스(neutron) **ml2_conf.ini** 파일에 지정된 범위 내에 속하는 네트워크의 DNS 레코드를 생성하지 않습니다.

추가 리소스

- 2.4절. "권장되는 DNS 서비스 토폴로지"
- DNS 통합을 위한 프로젝트 설정

2.3. DNS 서비스에 대한 기존 BIND 서버 구성

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)를 기존 BIND 인프라와 통합하는 경우 BIND 9이 올바르게 구성되었는지 확인하려면 몇 가지 작업을 수행해야 합니다.

중요

이 기능은 이번 릴리스에서 *기술 프리뷰*로 제공되므로 Red Hat에서 완전히 지원되지 않습니다. 테스트 용도로만 사용해야 하며 프로덕션 환경에 배포해서는 안 됩니다. 기술 프리뷰 기능에 대한 자세한 내용은 적용 범위 상세 정보를 참조하십시오.

참고

기존 BIND 인프라가 없는 경우 RHOSP director는 자동으로 BIND를 설정합니다.

사전 요구 사항

- BIND 9 서버를 변경할 수 있는 적절한 권한이 있는 사용자여야 합니다.
- BIND에서 **/etc/rndc.conf** 및 **/etc/rndc.key** 파일에 액세스할 수 있는지 확인합니다.
- BIND에서 RHOSP DNS 서비스(designate) **에서 r**ndc 유틸리티 메시지를 수신할 수 있는지 확인합니다.

프로세스

1. BIND 9 서버에 로그인합니다.
2. **/etc/rndc.key** 가 올바르게 구성되었는지 확인합니다.

rndc-key 에는 HMAC(Hash-based Message Authentication Code), SHA-256 알고리즘 및 Base64로 인코딩된 시크릿이 있어야 합니다.

```
key "rndc-key" {
    algorithm hmac-sha256;
    secret "<base64-encoded string>";
};
```

3. 아직 연결되지 않은 경우 BIND를 활성화하여 **rndc** 유틸리티를 사용하여 원격으로 영역을 생성하고 삭제합니다.
/etc/named.conf 에서 **options {** 에서 다음 행이 있는지 확인합니다. 없는 경우 새 행을 생성하고 추가합니다.

```
allow-new-zones yes;
```

4. 아직 수행하지 않은 경우 최소 응답을 보내도록 BIND를 구성합니다.
/etc/named.conf 에서도 **options {** 에서 다음 행이 있는지 확인합니다. 없는 경우 새 행을 생성하고 추가합니다.

```
minimal-responses yes;
```

기본적으로 BIND 9에는 영역 외부 레코드와 클라이언트에 전송하는 응답에 추가 섹션이 포함되어 있습니다. **최소 응답을 yes** 로 설정하면 영역 부족이 처리되지 않고 DNS 캐시 중독 공격에 대한 취약성이 제거됩니다.

추가 리소스

- 기존 BIND 9 서버를 사용하여 DNS 서비스 배포

2.4. 권장되는 DNS 서비스 토폴로지

권장 토폴로지는 RHOSP(Red Hat OpenStack Platform) 컨트롤러 호스트에 DNS 서비스를 배포하는 것으로 구성됩니다. RHOSP 배포가 고가용성인 경우 각각 DNS 서비스가 포함된 최소 3개의 RHOSP 컨트롤러가 있습니다.

그림 2.1. 권장되는 DNS 서비스 토폴로지

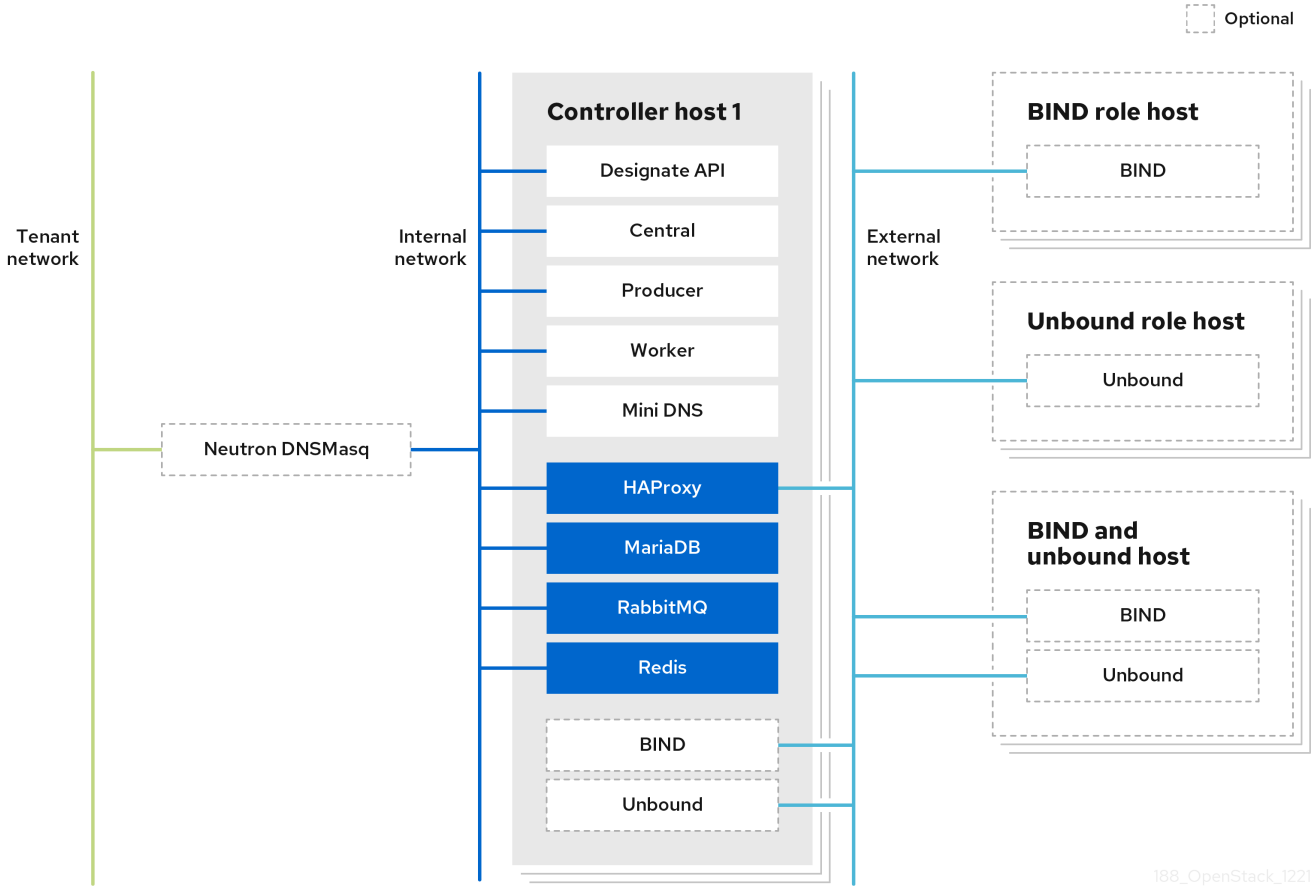


그림 2.1에서는 DNS 서비스 구성 요소가 해당 컨테이너에서 실행됩니다. 색상이 더 어두운 컨테이너는 DNS 서비스가 다른 RHOSP 서비스와 공유하는 리소스입니다.

점선된 컨테이너는 BIND 및 unbound에 대한 선택적 배치를 나타냅니다. 사이트에 데이터 트래픽 공간이 많은 경우 전용 호스트를 사용하여 각각 BIND 및 unbound를 포함할 수 있습니다.

2.5. DNS 서비스 고가용성 정보

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)는 *활성-활성* 고가용성 모드라는 고가용성 모드에서 데이터 트래픽 및 내결함성을 결합합니다. active-active 모드에서 DNS 서비스는 3개 이상의 노드에서 해당 구성 요소 서비스를 동시에 실행합니다. 노드 중 하나가 실패하면 나머지 노드가 계속 실행되고 중단을 방지하고 성능 저하를 방지합니다. DNS 서비스는 모든 서비스 인스턴스에서 부하 분산 작업을 시도합니다.

DNS 서비스 구성 요소는 RHOSP Controller 역할로 배포된 서비스로 분류됩니다. 즉 RHOSP 설치 및 구성 툴셋인 director는 모든 컨트롤러 호스트에 DNS 서비스를 자동으로 배포합니다. 따라서 세 개 이상의 컨트롤러가 세 개 이상의 다른 호스트에 배포된 경우 DNS 서비스를 사용할 수 있습니다.

추가 리소스

- DNS 서비스 구성 요소

3장. DNS 서비스 설치 및 구성

RHOSP(Red Hat OpenStack Platform)를 배포하거나 재배포할 때 지정 환경 파일을 포함하여 DNS 서비스(designate)를 설치하고 구성합니다. RHOSP, director 배포를 위한 툴셋에서는 오케스트레이션 서비스(heat) 환경 템플릿 및 환경 파일을 DNS 서비스 및 RHOSP 배포의 설치 및 구성 방법에 대한 계획 세트로 사용합니다.

DNS 서비스를 배포할 때 director는 active-active High Availability 모드의 DNS 서비스 활성화 및 포트 및 유동 IP 주소에 대한 자동화를 활성화하는 등의 작업을 자동으로 수행합니다. director는 DNS 서비스에 포함된 unbound 해석기를 가리키도록 Networking 서비스(neutron)를 설정합니다.

참고

사용자 정의 heat 환경 파일에서 **UnboundForwardResolvers** 를 설정하여 unbound 해석기의 구성을 명시적으로 비활성화할 수 있습니다.

director에 필요한 DNS 서버 정보를 제공하여 기존 DNS 인프라와 DNS 서비스를 통합할 수도 있습니다.

중요

RHOSP 17.1에서 DNS 서비스를 기존 DNS 인프라와 통합하는 것은 기술 프리뷰 기능입니다.

이 섹션에 포함된 항목은 다음과 같습니다.

- 3.1절. “DNS 서비스 배포”
- 3.2절. “기존 BIND 9 서버를 사용하여 DNS 서비스 배포”
- 3.3절. “DNS 서비스 기본 설정 변경”

3.1. DNS 서비스 배포

RHOSP(Red Hat OpenStack Platform) director를 사용하여 DNS 서비스(designate)를 배포합니다. director는 RHOSP 배포를 위한 일련의 계획인 오케스트레이션 서비스(heat) 템플릿 및 환경 파일을 사용합니다. 언더클라우드는 이러한 계획을 가져오고 해당 지침에 따라 DNS 서비스와 RHOSP 배포를 설치하고 구성합니다.

사전 요구 사항

- RHOSP 언더클라우드에 액세스할 수 있는 **stack** 사용자여야 합니다.

프로세스

1. DNS 서버를 기존 DNS 인프라와 통합하는 경우 3.2절. “기존 BIND 9 서버를 사용하여 DNS 서비스 배포” 주제로 이동합니다.
2. 언더클라우드 호스트에 stack 사용자로 로그인합니다.
3. 언더클라우드 인증 정보 파일을 가져옵니다.

```
$ source ~/stackrc
```

4. DNS 서버(designate) 풀에 상주하는 하위 영역의 이름 서버 레코드(NS 레코드) 값이 해당하는 **DesignateBindNSRecords** 매개변수에 대한 선언이 포함된 사용자 지정 환경 YAML 파일을 만듭니다.

```
parameter_defaults:
  DesignateBindNSRecords: ['<NS_record_child-zone-1>', '<NS_record_child-zone-2>', '...']
```

예제

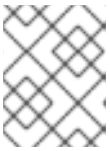
이 예에서 DNS 풀에는 하위 영역이 포함됩니다. **ns1.sales.example.org.**, **ns2.sales.example.org.**, 상위 영역 **example.org의 경우 ns3.sales**. **example.org.**:

```
parameter_defaults:
  DesignateBindNSRecords: ['ns1.sales.example.org.', 'ns2.sales.example.org.',
'ns3.sales.example.org.']
```

5. 배포 명령을 실행하고 코어 heat 템플릿, 기타 환경 파일, **designate.yaml** 환경 파일, 풀 NS 레코드가 포함된 파일을 포함합니다.

예제

```
$ openstack overcloud deploy --templates \
-e <other_environment_files> \
-e /usr/share/openstack-tripleo-heat-templates/environments/\
services/designate.yaml \
-e /home/stack/my_pool_ns_records.yaml
```



참고

director는 스택 업데이트 또는 업그레이드 중에 다양한 DNS 서비스 구성 요소를 최신 지정 이미지로 업데이트합니다.

검증

- DNS 서비스가 설치되고 엔드포인트가 정의되어 있는지 확인합니다.

```
$ openstack endpoint list -c "Service Name" -c Enabled -c URL
```

샘플 출력

```
+--------------+---------+-----------------------------------------------+
| Service Name | Enabled | URL                                           |
+--------------+---------+-----------------------------------------------+
| swift        | True    | http://198.51.100.61:8080                     |
| designate    | True    | http://203.0.113.103:9001                     |
| heat-cfn     | True    | http://192.0.2.137:8000/v1                    |
| designate    | True    | http://192.0.2.137:9001                       |
| placement    | True    | http://203.0.113.103:8778/placement           |
| cinderv3     | True    | http://203.0.113.103:8776/v3/%(tenant_id)s    |
| heat         | True    | http://203.0.113.103:8004/v1/%(tenant_id)s    |
| heat-cfn     | True    | http://203.0.113.103:8000/v1                  |
| nova         | True    | http://203.0.113.103:8774/v2.1                |
| heat         | True    | http://192.0.2.137:8004/v1/%(tenant_id)s      |
| glance       | True    | http://203.0.113.103:9292                     |
```

```
| heat         | True    | http://203.0.113.103:8004/v1/%(tenant_id)s      |
| glance       | True    | http://203.0.113.103:9292                       |
| neutron      | True    | http://203.0.113.103:9696                       |
| nova         | True    | http://192.0.2.137:8774/v2.1                    |
| cinderv3     | True    | http://192.0.2.137:8776/v3/%(tenant_id)s        |
| placement    | True    | http://203.0.113.103:8778/placement             |
| keystone     | True    | http://192.168.24.17:35357                      |
| neutron      | True    | http://192.0.2.137:9696                         |
| nova         | True    | http://203.0.113.103:8774/v2.1                  |
| heat-cfn     | True    | http://203.0.113.103:8000/v1                    |
| cinderv3     | True    | http://203.0.113.103:8776/v3/%(tenant_id)s      |
| glance       | True    | http://192.0.2.137:9292                         |
| placement    | True    | http://192.0.2.137:8778/placement               |
| swift        | True    | http://198.51.100.61:8080/v1/AUTH_%(tenant_id)s |
| swift        | True    | http://192.0.2.137:8080/v1/AUTH_%(tenant_id)s   |
| designate    | True    | http://203.0.113.103:9001                       |
| keystone     | True    | http://192.0.2.137:5000                         |
| neutron      | True    | http://203.0.113.103:9696                       |
| keystone     | True    | http://203.0.113.103:5000                       |
+--------------+---------+-------------------------------------------------+
```

추가 리소스

- *director 가이드를 사용하여 Red Hat OpenStack Platform 설치 및 관리의* 배포 명령 옵션

3.2. 기존 BIND 9 서버를 사용하여 DNS 서비스 배포

RHOSP(Red Hat OpenStack Platform) director를 사용하여 DNS 서비스(designate)를 설치 및 구성하고 기존 BIND 9 DNS 인프라와 통합합니다. director는 RHOSP 배포를 위한 일련의 계획인 오케스트레이션 서비스(heat) 템플릿 및 환경 파일을 사용합니다. DNS 서버에 대한 특정 정보를 heat 환경 파일에 추가합니다. 언더클라우드는 이러한 계획을 가져와서 RHOSP 및 DNS 서비스를 설치 및 구성하고 이를 DNS 인프라와 통합하기 위한 지침을 따릅니다.

중요

이 기능은 이번 릴리스에서 *기술 프리뷰*로 제공되므로 Red Hat에서 완전히 지원되지 않습니다. 테스트 용도로만 사용해야 하며 프로덕션 환경에 배포해서는 안 됩니다. 기술 프리뷰 기능에 대한 자세한 내용은 적용 범위 상세 정보를 참조하십시오.

사전 요구 사항

- BIND 9 서버를 사용하는 기존 DNS 인프라가 있습니다.
- BIND 9 서버가 DNS 서비스에 대한 기존 BIND 서버 구성에 설명된 구성 을 충족하는지 확인합니다.
- RHOSP 언더클라우드에 액세스할 수 있는 **stack** 사용자여야 합니다.

프로세스

1. DNS 서버를 기존 DNS 인프라와 통합 **하지 않는** 경우 3.1절. “DNS 서비스 배포” 주제로 이동하십시오.
2. 언더클라우드 호스트에 stack 사용자로 로그인합니다.

3. 언더클라우드 인증 정보 파일을 가져옵니다.

```
$ source ~/stackrc
```

4. 사용자 지정 환경 YAML 파일을 생성합니다.

예제

```
$ vi /home/stack/templates/my-designate-environment.yaml
```

5. 환경 파일에는 키워드 **parameter_defaults** 및 **DesignateExternalBindServers** 가 포함되어야 합니다. **DesignateExternalBindServers** 아래의 새 행에 BIND 9 DNS 서버 각각에 대한 IP 주소 및 RDMA(원격 이름 데몬 제어) 키를 추가합니다.

예제

이 예제에는 각각 RNDC 키가 있는 두 개의 기존 BIND 9 서버 **203.0.113.3** 과 **203.0.113.4** 가 있습니다.

```
parameter_defaults:
  DesignateExternalBindServers:
    - host: 203.0.113.3
      rndc_key: "FJOdVqZr5gVXbU9kIagY0IJVDq7CV/mDVb/M7mlLMgY="
    - host; 203.0.113.4
      rndc_key: "QAAACCdIV3KXPJh6U71ImVH0+j4uKRpVV49zVU7A8uvm"
```

6. DNS 서버(designate) 풀에 상주하는 하위 영역의 이름 서버 레코드(NS 레코드) 값이 해당하는 **DesignateBindNSRecords** 매개변수에 대한 선언을 추가합니다.

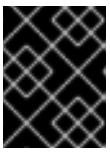
```
parameter_defaults:
...
  DesignateBindNSRecords: ['<NS_record_child-zone-1>', '<NS_record_child-zone-2>', '...']
```

예제

이 예에서 DNS 풀에는 하위 영역이 포함됩니다. **ns1.sales.example.org.**, **ns2.sales.example.org.**, 상위 영역 **example.org의 경우 ns3.sales**. **example.org.**:

```
parameter_defaults:
...
  DesignateBindNSRecords: ['ns1.sales.example.org.', 'ns2.sales.example.org.',
'ns3.sales.example.org.']
```

7. 배포 명령을 실행하고 코어 heat 템플릿, 기타 환경 파일, **designate.yaml** 환경 파일 및 이 새로운 사용자 지정 환경 파일을 포함합니다.



중요

후속 환경 파일에 정의된 매개변수와 리소스가 우선하므로 환경 파일의 순서가 중요합니다.

예제

```
$ openstack overcloud deploy --templates \
-e <other_environment_files> \
-e /usr/share/openstack-tripleo-heat-templates/environments/\
services/designate.yaml
```

참고

director는 스택 업데이트 또는 업그레이드 중에 다양한 DNS 서비스 구성 요소를 최신 지정 이미지로 업데이트합니다.

추가 리소스

- *director 가이드를 사용하여 Red Hat OpenStack Platform 설치 및 관리의* 배포 명령 옵션
- *Red Hat OpenStack Platform 배포 가이드의 환경 파일 사용자 지정*
- *Red Hat OpenStack Platform 배포 가이드 사용자 지정 에 오버클라우드 생성에 환경 파일 포함*

3.3. DNS 서비스 기본 설정 변경

YAML 형식의 환경 파일을 수정하고 RHOSP 오버클라우드를 재배포하여 RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)를 변경합니다. RHOSP director는 오케스트레이션 서비스(heat) 템플릿 및 환경 파일을 DNS 서비스를 구성하는 계획으로 사용하는 툴셋입니다.

사전 요구 사항

- RHOSP 언더클라우드에 액세스할 수 있는 **stack** 사용자여야 합니다.
- 수정할 RHOSP DNS 서비스 매개변수를 결정합니다.
  다음은 몇 가지 예입니다.
  - **DesignateRpcResponseTimeout**
    DNS 서비스의 RPC 응답 시간(초)입니다. 기본값은 60초입니다.
  - **DesignateWorkers**
    Designate 서비스의 작업자 수입니다. 기본값은 0(0)입니다. 즉, 배포 스크립트가 운영 체제 작업자에 RHOSP director 값을 사용합니다.

    자세한 내용은 *director 가이드를 사용하여 Red Hat OpenStack Platform 설치 및 관리의* 환경 확장 결정을 참조하십시오.
  - **DesignateMdnsProxyBasePort**
    외부 또는 공용 액세스 네트워크의 MiniDNS 프록시 엔드 포인트의 기본 포트입니다. 기본 포트는 16000입니다.

프로세스

1. 언더클라우드 호스트에 **stack** 사용자로 로그인합니다.
2. 언더클라우드 인증 정보 파일을 가져옵니다.

```
$ source ~/stackrc
```

3. 사용자 지정 YAML 환경 파일을 생성합니다.

예제

```
$ vi /home/stack/templates/my-designate-environment.yaml
```

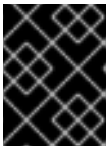
환경 파일에는 키워드 **parameter_defaults** 가 포함되어야 합니다. **parameter_defaults** 키워드 뒤에 매개변수 값 쌍을 배치합니다.

예제

이 예에서 RPC 응답 시간 초과는 120초로 설정됩니다.

```
parameter_defaults:
  DesignateRpcResponseTimeout: '120'
```

4. 배포 명령을 실행하고 코어 heat 템플릿, 기타 환경 파일, **designate.yaml** 환경 파일 및 이 새로운 사용자 지정 환경 파일을 포함합니다.



중요

후속 환경 파일에 정의된 매개변수와 리소스가 우선하므로 환경 파일의 순서가 중요합니다.

예제

```
$ openstack overcloud deploy --templates \
-e <other_environment_files> \
-e /usr/share/openstack-tripleo-heat-templates/environments/\
services/designate.yaml \
-e /home/stack/templates/my-designate-environment.yaml
```

추가 리소스

- *Overcloud 매개변수 가이드의 DNS(designate) 매개변수* 가이드
- *Red Hat OpenStack Platform 배포 가이드의 환경 파일 사용자 지정*
- *Red Hat OpenStack Platform 배포 가이드 사용자 지정 에 오버클라우드 생성에 환경 파일 포함*

4장. 통합 DNS 서비스 사용

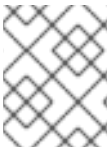
RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)는 Networking 서비스(neutron)와 통합되어 포트 및 Compute 서비스(nova), 가상 머신 인스턴스를 통해 자동으로 레코드 세트 생성을 제공합니다.

클라우드 관리자는 DNS 서비스를 사용하여 네트워크에 연결된 영역을 생성합니다. 클라우드 관리자가 제공하는 이 네트워크를 사용하여 클라우드 사용자는 가상 머신 인스턴스, 포트 또는 유동 IP를 생성할 수 있으며 DNS 서비스는 필요한 DNS 레코드를 자동으로 생성합니다.

DNS 서비스를 배포하는 동안 RHOSP director는 네트워킹 서비스(neutron) 확장인 **dns_domain_ports**를 로드합니다. 이 확장을 사용하면 RHOSP 포트, 네트워크 및 유동 IP에 다음 DNS 속성을 추가할 수 있습니다.

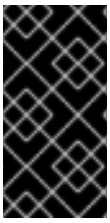
표 4.1. RHOSP 네트워킹 및 DNS 서비스에서 지원하는 DNS 설정

리소스	DNS 이름	DNS 도메인 (zone)
포트	제공됨	제공됨
네트워크	없음	제공됨
부동 IP	제공됨	제공됨



참고

네트워크와 유동 IP 모두에 지정된 DNS 도메인의 경우 유동 IP의 포트에 있는 도메인이 네트워크에 설정된 도메인보다 우선합니다.



중요

RHOSP(Red Hat OpenStack Platform) 17.1 GA에서는 RHOSP Networking 서비스(neutron) ML2/OVN과 RHOSP DNS 서비스(designate) 간의 통합에 기술 프리뷰를 사용할 수 있습니다. 결과적으로 DNS 서비스는 새로 생성된 VM에 대한 DNS 항목을 자동으로 추가하지 않습니다.

이 섹션에 포함된 항목은 다음과 같습니다.

4.1. DNS 통합을 위한 프로젝트 설정

클라우드 관리자는 클라우드 사용자가 가상 머신 인스턴스, 포트 또는 유동 IP를 생성할 때 지정해야 하는 필수 영역, 네트워크 및 서브넷을 만듭니다. RHOSP Networking 서비스(neutron)는 DNS 서비스(designate)와 통합되므로 클라우드 사용자가 이러한 오브젝트를 생성할 때 DNS 서비스에 자동으로 추가됩니다.

중요

이 기능은 이번 릴리스에서 *기술 프리뷰*로 제공되므로 Red Hat에서 완전히 지원되지 않습니다. 테스트 용도로만 사용해야 하며 프로덕션 환경에 배포해서는 안 됩니다. 기술 프리뷰 기능에 대한 자세한 내용은 적용 범위 상세 정보를 참조하십시오.

사전 요구 사항

- **admin** 역할이 있는 RHOSP 사용자여야 합니다.
- 포트 및 VM에 사용되는 네트워크에는 **router:external** 속성이 **True** 로 설정될 수 없습니다. 네트워크를 생성할 때 **--external** 옵션을 지정할 수 없습니다.
- 네트워크는 Cryostat, VLAN, GRE, VXLAN 또는 GENEVE 유형 중 하나여야 합니다.
- VLAN, GRE, VXLAN 또는 GENEVE 네트워크의 경우 분할 ID는 네트워킹 서비스 **ml2_conf.ini** 파일에 구성된 범위를 벗어나야 합니다.
  ml2_conf.ini 파일은 컨트롤러 노드 호스트에 있습니다. **/var/lib/config-data/puppet-generated/neutron/etc/neutron/plugins/ml2**.+ 다음 표를 사용하여 네트워크 세그먼트 ID 범위를 참조하는 섹션 및 옵션을 확인합니다.

표 4.2. 네트워크 분할 ID를 설정하는 데 사용되는ml2_conf.ini 옵션

네트워크 유형	섹션	옵션
Geneve	[ml2_type_geneve]	vni_ranges
GRE	[ml2_type_gre]	tunnel_id_ranges
VLAN	[ml2_type_vlan]	network_vlan_ranges
VXLAN	[ml2_type_vxlan]	vni_ranges

참고

이러한 사전 요구 사항이 모두 충족되지 않으면 Networking 서비스에서 기본 **dns_domain** 값인 **openstacklocal.** 을 사용하여 내부 확인자에 DNS 할당을 생성합니다.

프로세스

1. 클라우드 관리자로 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. 특정 프로젝트의 사용자가 DNS 항목을 생성할 영역을 생성합니다.

   예제

   이 예에서 클라우드 관리자는 **example.com** 이라는 영역을 생성하고 프로젝트 ID인 **f75ec24a-d361-ab86-54c0-dfe6093245a3** 의 사용자에게 레코드 세트를 영역에 추가할 수 있는 권한을 갖도록 지정합니다.

```
$ openstack zone create --email example@example.com example.com. --sudo-project-id
f75ec24a-d361-ab86-54c0-dfe6093245a3
```

참고

DNS 도메인은 항상 FQDN(정규화된 도메인 이름)이어야 합니다. 즉, 항상 마침표로 끝납니다.

3. 특정 프로젝트의 사용자가 DNS 항목을 생성할 네트워크를 생성합니다.

예제

이 예에서 클라우드 관리자는 이전에 생성된 영역 **example.com., 및 segmentation ID, 즉 ml2_conf.ini에 정의된 범위를 벗어나는 네트워크를** 생성합니다.

```
$ openstack network create --dns-domain example.com. \
--provider-segment 2017 --provider-network-type geneve \
example-network
```

4. **네트워크에서 서브넷을 만듭니다.**

예제

이 예제에서 클라우드 관리자는 네트워크 example-network 에 서브넷 example-subnet 을 생성합니다.

```
$ openstack subnet create \
  --allocation-pool start=192.0.2.10,end=192.0.2.200 \
  --network example-network \
  --subnet-range 192.0.2.0/24 \
  example-subnet
```

5. **프로젝트의 클라우드 사용자에게 인스턴스, 포트, 유동 IP를 추가할 때 생성한 영역과 네트워크를 사용하도록 지시합니다.**

주의

인스턴스, 포트 또는 유동 IP를 생성하는 사용자에게 영역에 레코드 세트를 생성할 수 있는 권한이 없거나 영역이 DNS 서비스에 없는 경우 네트워킹 서비스는 다음을 수행합니다.

- 제공된 dns_domain 을 사용하여 dns_assignment 필드를 사용하여 포트를 생성합니다.
- DNS 서비스에 레코드 세트를 생성하지 않습니다.
- "외부 DNS 서비스에 포트 데이터를 게시하는 동안 오류가 발생했습니다."

검증

- 생성한 네트워크가 있는지 확인합니다.

예제

```
$ openstack network show example-network
```

샘플 출력

```
+---------------------------+--------------------------------------+
| Field                     | Value                                |
+---------------------------+--------------------------------------+
| admin_state_up            | UP                                   |
| availability_zone_hints   |                                      |
| availability_zones        |                                      |
| created_at                | 2022-09-07T19:03:32Z                 |
| description               |                                      |
```

```
| dns_domain              | example.com.                         |
| id                      | 9ae5b3d5-f12c-4a67-b0e5-655d53cd4f7c |
| ipv4_address_scope      | None                                 |
| ipv6_address_scope      | None                                 |
| is_default              | None                                 |
| is_vlan_transparent     | None                                 |
| mtu                     | 1450                                 |
| name                    | network-example                      |
| port_security_enabled   | True                                 |
| project_id              | f75ec24a-d361-ab86-54c0-dfe6093245a3 |
| provider:network_type   | vxlan                                |
| provider:physical_network | None                               |
| provider:segmentation_id  | 2017                               |
| qos_policy_id           | None                                 |
| revision_number         | 3                                    |
| router:external         | Internal                             |
| segments                | None                                 |
| shared                  | False                                |
| status                  | ACTIVE                               |
| subnets                 | 15546c9d-6faf-43aa-83e7-b1e705eed060 |
| tags                    |                                      |
| updated_at              | 2022-09-07T19:03:43Z                 |
+-------------------------+--------------------------------------+
```

추가 리소스

- *명령줄 인터페이스 참조*의 영역
- *명령줄 인터페이스 참조*의 네트워크
- *명령줄 인터페이스 참조*의 서브넷

4.2. DNS와 가상 머신 인스턴스 통합

Networking 서비스(neutron)와 DNS 서비스(designate) 간의 통합을 통해 가상 머신 인스턴스를 생성할 때마다 DNS를 자동으로 활성화할 수 있습니다.

사전 요구 사항

- **클라우드 관리자는 DNS 지원 인스턴스를 생성할 때 사용할 필수 네트워크를 제공했습니다.**

프로세스

1. **자격 증명 파일을 가져옵니다.**

   예제

   ```
   $ source ~/overcloudrc
   ```

2. **클라우드 관리자가 제공한 네트워크를 사용하여 인스턴스를 생성합니다.**

   예제

   이 예제에서 cloud 사용자는 my_vm 이라는 인스턴스를 생성합니다.

   ```
   $ openstack server create --image cirros-0.5.2-x86_64-disk --flavor m1.micro --nic net-id=example-network my_vm
   ```

검증

- **생성한 인스턴스의 DNS 서비스에 레코드가 있는지 확인합니다.**

  예제

  이 예에서는 example.com. 영역에 대한 DNS 서비스를 쿼리합니다.

  ```
  $ openstack recordset list --type A example.com.
  ```

  샘플 출력

  ```
  +--------------+--------------------+------+------------+--------+--------+
  | id           | name               | type | records    | status | action |
  +--------------+--------------------+------+------------+--------+--------+
  | 7b8d1be6-1b23 | my_vm.example.com.  | A    | 192.0.2.44 | ACTIVE | NONE   |
  | -478a-94d5-60 |                    |      |            |        |        |
  | b876dca2c8    |                    |      |            |        |        |
  +--------------+--------------------+------+------------+--------+--------+
  ```

추가 리소스

- *명령줄 인터페이스 참조*에서 **server create**

4.3. DNS와 포트 통합

Networking 서비스(neutron)와 DNS 서비스(designate) 간의 통합을 통해 포트를 생성할 때마다 DNS 레코드 세트를 자동으로 추가할 수 있습니다.

사전 요구 사항

- **클라우드 관리자는 DNS 사용 포트를 생성할 때 사용할 필수 네트워크를 제공했습니다.**

프로세스

1. **자격 증명 파일을 가져옵니다.**

   예제

   ```
   $ source ~/overcloudrc
   ```

2. **클라우드 관리자가 제공한 영역 및 네트워크를 사용하여 포트를 생성합니다.**

   예제

   이 예에서 클라우드 사용자는 네트워크에 example-port 의 DNS 이름을 사용하여 my-port 포트를 생성합니다.

   ```
   $ openstack port create --network example-network \
   --dns-name example-port \
   my-port
   ```

검증

- 생성한 포트의 **DNS** 서비스에 레코드가 있는지 확인합니다.

예제

이 예에서는 **example.com.** 영역에 대한 **DNS** 서비스를 쿼리합니다.

```
$ openstack recordset list --type A example.com.
```

샘플 출력

```
+--------------+-------------------------+------+-------------+--------+--------+
| id           | name                    | type | records     | status | action |
+--------------+-------------------------+------+-------------+--------+--------+
| 9ebbe94f-2442 | example-port.example.com. | A    | 192.0.2.149 | ACTIVE | NONE   |
| -4bb8-9cfa-6d |                         |      |             |        |        |
| ca1daba73f    |                         |      |             |        |        |
+--------------+-------------------------+------+-------------+--------+--------+
```

추가 리소스

- *명령줄 인터페이스 참조*에서 **port create**

4.4. DNS와 유동 IP 통합

Networking 서비스(**neutron**)와 **DNS** 서비스(**designate**) 간의 통합을 통해 유동 **IP**를 생성할 때마다 **DNS** 레코드 세트를 자동으로 추가할 수 있습니다.

사전 요구 사항

- 클라우드 관리자는 **DNS** 지원 유동 **IP**를 생성할 때 사용할 필수 외부 네트워크를 제공했습니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. 클라우드 관리자가 제공한 영역과 외부 네트워크를 사용하여 유동 IP를 만듭니다.

예제

이 예제에서 **cloud** 사용자는 **DNS** 이름 **example-fip, network, public** 을 사용하여 유동 **IP**를 생성합니다.

```
$ openstack floating ip create --dns-name example-fip \
--dns-domain example.com. \
public
```

검증

- 생성한 유동 IP의 DNS 서비스에 레코드가 있는지 확인합니다.

예제

이 예에서는 **example.com.** 영역에 대한 **DNS** 서비스를 쿼리합니다.

```
$ openstack recordset list --type A example.com.
```

샘플 출력

```
+--------------+-------------------------+------+-------------+--------+--------+
| id           | name                    | type | records     | status | action |
+--------------+-------------------------+------+-------------+--------+--------+
| e1eca823-169d | example-fip.example.com. | A    | 192.0.2.106 | ACTIVE | NONE   |
| -4d0a-975e-91 |                         |      |             |        |        |
| a9907ec0c1    |                         |      |             |        |        |
+--------------+-------------------------+------+-------------+--------+--------+
```

추가 리소스

- *명령줄 인터페이스 참조*의 **floating ip create**

5장. 최상위 도메인 이름 관리

이 섹션에서는 최상위 도메인을 소개하고 **Red Hat OpenStack Platform DNS** 서비스(**designate**)에서 생성 및 관리하는 방법을 설명합니다. 사용자가 생성할 수 있는 도메인 이름을 관리하는 방법은 거부 목록을 사용하는 것입니다.

이 섹션에 포함된 항목은 다음과 같습니다.

5.1. 최상위 도메인 정보

최상위 도메인(TLD)을 사용하여 사용자가 영역을 생성할 수 있는 도메인을 제한할 수 있습니다. DNS(Domain Name System)에서 *TLD* 는 특별히 루트 아래에 있는 도메인 집합(예: .com)을 나타냅니다. RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)에서 TLD는 모든 유효한 도메인이 될 수 있습니다.

TLD는 허용된 도메인 세트를 정의하므로 사용자가 생성하는 영역은 TLD 중 하나에 있어야 합니다. DNS 서비스에서 TLD가 생성되지 않은 경우 사용자는 모든 영역을 생성할 수 있습니다. TLD에는 권한이 있는 사용자가 허용된 TLD 외부에서 영역을 생성할 수 있는 정책이 없습니다.

예제

.com TLD를 만든 후 사용자가 .com TLD 내에 포함되지 않은 영역을 만들려고 하면 시도에 실패합니다.

```
$ openstack zone create --email admin@test.net test.net.
```

샘플 출력

```
Invalid TLD
```

OpenStack Client openstack tld 명령을 사용하여 TLD를 생성, 나열, 표시, 수정, 삭제할 수 있습니다.

추가 리소스

- *명령줄 인터페이스 참조*의 TLD https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/17.1/html/command_line_interface_reference/tld
- *명령줄 인터페이스 참조*의 영역

5.2. 최상위 도메인 생성

최상위 도메인(TLD)을 사용하면 사용자가 영역을 생성할 수 있는 도메인을 제한할 수 있습니다. RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)에서 TLD는 모든 유효한 도메인이 될

수 있습니다. **TLD**를 만들려면 **OpenStack Client openstack tld create** 명령을 사용합니다.

사전 요구 사항

- **admin** 역할이 있는 **RHOSP** 사용자여야 합니다.

프로세스

1. 클라우드 관리자로 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. **openstack tld create** 명령을 실행하여 **TLD**를 생성합니다.

   예제

   예를 들어 **.org** 로 끝나는 영역을 생성해야 하는 경우 단일 **.org TLD**를 생성할 수 있습니다.

   ```
   $ openstack tld create --name org
   ```

   샘플 출력

   ```
   +-------------+--------------------------------------+
   | Field       | Value                                |
   +-------------+--------------------------------------+
   | created_at  | 2022-01-10T13:07:33.000000           |
   | description | None                                 |
   | id          | 9fd0a12d-511e-4024-bf76-6ec2e3e71edd |
   | name        | org                                  |
   | updated_at  | None                                 |
   +-------------+--------------------------------------+
   ```

작은 정보

openstack tld 명령을 사용하는 경우 입력한 **FQDN**(정규화된 도메인 이름)에 후행 점이 없는지 확인합니다(예: **.net**).

검증

- **openstack tld list** 명령을 실행하고 **TLD**가 있는지 확인합니다.

예제

```
$ openstack tld list --name zone1.cloud.example.com
```

추가 리소스

- *명령줄 인터페이스 참조*에서 **TLD** 만들기

5.3. 최상위 도메인 나열 및 표시

Red Hat OpenStack Platform DNS 서비스(**designate**) 데이터베이스를 쿼리하고 모든 최상위 도메인(**TLD**)을 나열하거나 특정 **TLD**의 속성을 표시할 수 있습니다. 이 작업을 수행하는 **OpenStack Client** 명령은 각각 **openstack tld list** 및 **openstack tld show** 입니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. 다음 명령을 사용하여 **DNS** 서비스 데이터베이스의 모든 **TLD**를 나열합니다.

```
$ openstack tld list
```

3. **openstack tld show <TLD_NAME_or_ID >** 명령을 사용하여 특정 **TLD**의 속성을 표시합니다.

예제

```
$ openstack tld show org
```

추가 리소스

- *명령줄 인터페이스 참조*의 **TLD** 목록
- *명령줄 인터페이스 참조*에**TLD** 표시

5.4. 최상위 도메인 수정

RHOSP(Red Hat OpenStack Platform) DNS 서비스(**designate**)를 사용하면 이름과 같은 최상위 도메인(**TLD**)의 다양한 속성을 변경할 수 있습니다. **OpenStack Client openstack tld set** 명령을 사용하여 **TLD**를 수정합니다.

사전 요구 사항

- **admin** 역할이 있는 **RHOSP** 사용자여야 합니다.

프로세스

1. 클라우드 관리자로 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. 다음 명령 옵션을 사용하여 **TLD**를 다양한 방법으로 수정할 수 있습니다.

```
openstack tld set [--name NAME] \
                  [--description DESCRIPTION | --no-description] \
                  [TLD_ID | TLD_NAME]
```

참고

이전 구문 다이어그램에는 **openstack tld set** 명령에 대한 다양한 포맷 옵션이 표시되지 않습니다. 모든 명령 옵션 목록은 나중에 "추가 리소스"의 링크를 참조하십시오.

이 예제에서 **openstack tld set** 명령은 **org TLD**의 이름을 **example.net** 으로 변경합니다.

예제

```
$ openstack tld set org --name example.net
```

샘플 출력

```
+-------------+--------------------------------------+
| Field       | Value                                |
+-------------+--------------------------------------+
| created_at  | 2022-01-10T13:07:33.000000           |
```

```
| description |                                      |
| id          | 9fd0a12d-511e-4024-bf76-6ec2e3e71edd |
| name        | example.net                          |
| updated_at  | 2022-01-10T22:35:20.000000           |
+-------------+--------------------------------------+
```

검증

- **openstack tld show <TLD_NAME_or_ID >** 명령을 실행하고 수정 사항이 있는지 확인합니다.

추가 리소스

- *명령줄 인터페이스 참조에* 설정된 **TLD**

5.5. 최상위 도메인 삭제

RHOSP(Red Hat OpenStack Platform) DNS 서비스**(designate)**를 사용하면 **OpenStack Client openstack tld delete** 명령을 사용하여 최상위 도메인**(TLD)**을 제거할 수 있습니다.

사전 요구 사항

- **admin** 역할이 있는 **RHOSP** 사용자여야 합니다.

프로세스

1. 클라우드 관리자로 자격 증명 파일을 가져옵니다.

   예제

```
$ source ~/overcloudrc
```

2. 다음 명령을 실행하여 삭제할 **TLD**의 **ID** 또는 이름을 가져옵니다.

```
$ openstack tld list
```

3. 이전 단계의 이름 또는 **ID**를 사용하여 다음 명령을 입력합니다.

```
$ openstack tld delete <TLD_NAME_or_ID>
```

이 명령이 성공할 때 출력이 없습니다.

검증

- **openstack tld show <TLD_NAME_or_ID >** 명령을 실행하고 **TLD**가 제거되었는지 확인합니다.

추가 리소스

- *명령줄 인터페이스 참조*에서 **TLD** 삭제

5.6. DNS 서비스 거부 목록 정보

RHOSP(Red Hat OpenStack Platform) DNS 서비스(**designate**)에는 사용자가 특정 정규식과 일치하는 이름으로 영역을 생성하지 못하도록 하는 거부 목록 기능이 있습니다. 예를 들어 거부 목록을 사용하여 사용자가 다음을 수행할 수 없도록 할 수 있습니다.

- 특정 영역 생성.
- 특정 문자열이 포함된 영역을 생성합니다.
- 특정 영역의 하위 영역 생성.

example.com. 가 거부 목록의 멤버이고 도메인 또는 프로젝트 사용자가 **foo.example.com.** 또는 **example.com**과 같은 영역을 만들려고 하면 오류가 발생합니다.

```
$ openstack zone create --email admin@example.com example.com.
```

```
Blacklisted zone name
$ openstack zone create --email admin@example.com foo.example.com.
Blacklisted zone name
```

참고

use_blacklisted_zone 역할 기반 액세스 제어를 충족하는 사용자는 거부 목록에 있는 이름으로 영역을 생성할 수 있습니다. 기본적으로 이 재정의가 있는 사용자만 **RHOSP** 시스템 관리자입니다.

OpenStack Client openstack zone blacklist 명령을 사용하여 거부 목록을 생성, 나열, 표시, 수정, 삭제할 수 있습니다.

추가 리소스

- *명령줄 인터페이스 참조*에서 영역 블랙리스트 생성

5.7. 거부 목록의 DNS 서비스 정규식 정보

Red Hat OpenStack Platform DNS 서비스(designate)에서 거부 목록 작업의 대부분은 사용하기 어려울 수 있는 정규식(regexes)을 사용하고 있습니다. **regex**에 대한 **Python** 문서는 유용한 소개 역할을 할 수 있습니다. 온라인 정규 표현식 툴은 **regex**를 빌드하고 테스트하여 거부 목록 **API**와 함께 사용할 때 지원할 수 있습니다.

추가 리소스

- **Python 3** 문서의 정규식 **HOWTO**
- **5.6절. “DNS 서비스 거부 목록 정보”**

5.8. DNS 서비스 거부 목록 생성

Red Hat OpenStack Platform DNS 서비스(designate)의 **Denylists**를 사용하면 사용자가 특정 정규식과 일치하는 이름으로 영역을 생성하지 못하도록 할 수 있습니다. **OpenStack Client openstack zone blacklist create** 명령을 사용하여 거부 목록을 생성합니다.

사전 요구 사항

- **admin** 역할이 있는 **RHOSP** 사용자여야 합니다.

프로세스

1. 클라우드 관리자로 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. **openstack zone blacklist create** 명령을 사용하여 거부 목록을 생성합니다.

이 예에서 **domain example.com.** 및 해당 하위 도메인의 모든 하위 도메인이 거부 목록에 추가됩니다.

예제

```
$ openstack zone blacklist create --pattern ".*example.com."
```

샘플 출력

```
+-------------+--------------------------------------+
| Field       | Value                                |
+-------------+--------------------------------------+
| created_at  | 2021-10-20T16:15:18.000000           |
| description | None                                 |
| id          | 7622e241-8c3d-4c03-a692-8747e3cf2658 |
| pattern     | .*example.com.                       |
| updated_at  | None                                 |
+-------------+--------------------------------------+
```

검증

- **openstack zone blacklist list** 명령을 실행하고 거부 목록이 있는지 확인합니다.

추가 리소스

- *명령줄 인터페이스 참조*에서 영역 블랙리스트 생성
- 5.7절. “거부 목록의 DNS 서비스 정규식 정보”

5.9. DNS 서비스 거부 목록 나열 및 표시

Red Hat OpenStack Platform DNS 서비스(**designate**) 데이터베이스를 쿼리하고 모든 거부 목록을 보거나 특정 거부 목록에 대한 속성을 표시할 수 있습니다. 이 작업을 수행하는 **OpenStack Client** 명령은 **openstack zone blacklist list** 이고 **openstack zone blacklist**에 각각 이 표시됩니다.

다른 거부 목록 명령을 사용하려면 거부 목록 **ID**를 알아야 하므로 모든 거부 목록을 볼 수 있습니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. 다음 명령을 사용하여 **DNS** 서비스 데이터베이스에 거부 목록을 나열합니다.

   ```
   $ openstack zone blacklist list
   ```

- 이전 단계에서 가져온 거부 목록 **ID**를 사용하면 **openstack zone blacklist show <denylist_ID >** 명령을 사용하여 특정 거부 목록에 대한 속성을 표시합니다.

예제

```
$ openstack zone blacklist show 7622e241-8c3d-4c03-a692-8747e3cf2658
```

추가 리소스

- *명령줄 인터페이스 참조*의 영역 블랙리스트 목록
- *명령줄 인터페이스 참조에* 영역 블랙리스트 표시

5.10. DNS 서비스 거부 목록 수정

Red Hat OpenStack Platform DNS 서비스(**designate**)를 사용하면 거부 목록을 수정할 수 있습니다. 예를 들어 사용자가 이전에 제한된 특정 도메인 이름으로 영역을 생성할 수 있도록 거부 목록을 변경할 수 있습니다. **OpenStack Client openstack zone blacklist set** 명령을 사용하여 거부 목록을 수정합니다.

사전 요구 사항

- **admin** 역할이 있는 **RHOSP** 사용자여야 합니다.

프로세스

1. 클라우드 관리자로 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. 다음 명령을 실행하여 수정할 거부 목록의 **ID**를 가져옵니다.

```
$ openstack zone blacklist list
```

3. 다음 명령 옵션을 사용하여 다양한 방법으로 거부 목록을 수정할 수 있습니다.

```
$ openstack zone blacklist set \
      [--description DESCRIPTION | --no-description] denylist_ID
```

참고

이전 구문 다이어그램에는 **openstack zone blacklist set** 명령에 대한 다양한 포맷 옵션이 표시되지 않습니다. 모든 명령 옵션 목록은 나중에 "추가 리소스"의 링크를 참조하십시오.

이 예에서는 **web.example.com** 도메인을 허용하도록 정규식**(regex)**이 변경되었습니다.

예제

```
$ openstack zone blacklist set 81fbfe02-6bf9-4812-a40e-1522ab6862ca --pattern
".*web.example.com"
```

샘플 출력

```
+-------------+--------------------------------------+
| Field       | Value                                |
+-------------+--------------------------------------+
| created_at  | 2022-01-08T09:11:43.000000           |
| description | None                                 |
| id          | 81fbfe02-6bf9-4812-a40e-1522ab6862ca |
| pattern     | .*web.example.com                    |
| updated_at  | 2022-01-15T14:26:18.000000           |
+-------------+--------------------------------------+
```

검증

- **openstack zone blacklist show <denylist_ID& gt;** 명령을 실행하고 수정 사항이 있는지 확인합니다.

추가 리소스

- *명령줄 인터페이스 참조에* 설정된 영역 블랙리스트
- 5.7절. “거부 목록의 DNS 서비스 정규식 정보”

5.11. DNS 서비스 거부 목록 삭제

Red Hat OpenStack Platform DNS 서비스(designate)의 **Denylists**를 사용하면 사용자가 특정 정규식과 일치하는 이름으로 영역을 생성하지 못하도록 할 수 있습니다. **OpenStack Client openstack zone blacklist delete** 명령을 사용하여 거부 목록을 제거합니다.

사전 요구 사항

- **admin** 역할이 있는 **RHOSP** 사용자여야 합니다.

프로세스

1. 클라우드 관리자로 자격 증명 파일을 가져옵니다.

   예제

```
$ source ~/overcloudrc
```

2. 다음 명령을 실행하여 삭제할 거부 목록의 **ID**를 가져옵니다.

```
$ openstack zone blacklist list
```

3. 이전 단계의 **ID**를 사용하여 다음 명령을 입력합니다.

```
$ openstack zone blacklist delete <denylist_ID>
```

이 명령이 성공할 때 출력이 없습니다.

검증

- **openstack zone blacklist show <denylist_ID >** 명령을 실행하고 거부 목록이 제거되었는지 확인합니다.

추가 리소스

- *명령줄 인터페이스 참조*에서 영역 블랙리스트 삭제

6장. DNS 리소스에서 할당량 보기 및 관리

RHOSP(Red Hat OpenStack Platform)는 DNS 서비스(designate)를 사용하여 클라우드 관리자가 수정할 수 있는 DNS 리소스 할당량 세트를 제공합니다. DNS 할당량을 사용하면 프로젝트의 DNS 리소스에 제한을 설정하여 서비스 거부 공격과 같은 이벤트에서 RHOSP 사이트를 보호하는 데 도움이 될 수 있습니다. DNS 할당량을 사용하면 사용자의 DNS 리소스 사용량을 추적하는 데 도움이 될 수 있습니다. 클라우드 관리자는 모든 프로젝트에 적용되는 DNS 할당량 값을 설정하거나 프로젝트별로 하나 이상의 할당량을 구성할 수 있습니다.

이 섹션에 포함된 항목은 다음과 같습니다.

6.1. DNS 리소스의 할당량 보기

DNS 서비스(designate)를 사용하여 RHOSP(Red Hat OpenStack Platform) 프로젝트의 리소스 할당량을 볼 수 있습니다.

사전 요구 사항

- 확인할 할당량이 있는 프로젝트의 멤버여야 합니다.
- admin 역할이 있는 RHOSP 사용자는 모든 프로젝트의 할당량을 볼 수 있습니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. 프로젝트에 설정된 **DNS** 리소스 할당량을 확인합니다.

```
$ openstack dns quota list
```

샘플 출력

```
+-------------------+-------+
| Field             | Value |
+-------------------+-------+
| api_export_size   | 1000  |
| recordset_records | 20    |
| zone_records      | 500   |
| zone_recordsets   | 500   |
| zones             | 10    |
+-------------------+-------+
```

3. **admin** 역할이 있는 **RHOSP** 사용자는 다른 프로젝트의 할당량을 쿼리할 수 있습니다.

   a. 수정할 할당량이 있는 프로젝트의 **ID**를 가져옵니다.

      나중에 단계를 수행하는 데 필요하므로 **ID**를 기억할 수 있습니다.

      ```
      $ openstack project list
      ```

   b. 프로젝트 **ID**를 사용하여 프로젝트에 설정된 **DNS** 리소스 할당량을 확인합니다.

      예제

      이 예에서는 프로젝트 **ID ecd4341280d645e5959d32a4b7659da1** 의 **DNS** 할당량이 표

시됩니다.

```
$ openstack dns quota list --project-id ecd4341280d645e5959d32a4b7659da1
```

샘플 출력

```
+-------------------+-------+
| Field             | Value |
+-------------------+-------+
| api_export_size   | 2500  |
| recordset_records | 25    |
| zone_records      | 750   |
| zone_recordsets   | 750   |
| zones             | 25    |
+-------------------+-------+
```

추가 리소스

- *명령줄 인터페이스 참조의* ***DNS*** 할당량 목록

6.2. DNS 리소스의 할당량 수정

DNS 서비스(designate)를 사용하여 RHOSP(Red Hat OpenStack Platform) 프로젝트의 DNS 리소스 할당량을 변경할 수 있습니다.

사전 요구 사항

- **admin 역할이 있는 RHOSP 사용자여야 합니다.**

프로세스

1. **클라우드 관리자로 자격 증명 파일을 가져옵니다.**

   예제

```
$ source ~/overcloudrc
```

2. 수정할 할당량이 있는 프로젝트의 **ID**를 가져옵니다.

   나중에 단계를 수행하는 데 필요하므로 **ID**를 기억할 수 있습니다.

```
$ openstack project list
```

3. 프로젝트 **ID**를 사용하여 프로젝트의 **DNS** 리소스 할당량을 수정합니다. 사용 가능한 할당량 목록은 **6.4**절. **"DNS** 서비스 할당량 및 기본값**"** 을 참조하십시오.

   예제

   이 예에서는 영역 할당량이 수정되었습니다. 프로젝트 **ID ecd4341280d645e5959d32a4b7659da1** 에서 사용할 수 있는 총 영역 수는 **30**입니다.

```
$ openstack dns quota set --project-id ecd4341280d645e5959d32a4b7659da1 --zones 30
```

샘플 출력

```
+-------------------+-------+
| Field             | Value |
+-------------------+-------+
| api_export_size   | 1000  |
| recordset_records | 20    |
| zone_records      | 500   |
| zone_recordsets   | 500   |
| zones             | 30    |
+-------------------+-------+
```

추가 리소스

- *명령줄 인터페이스 참조에* 설정된 **DNS** 할당량

- 6.4절. “DNS 서비스 할당량 및 기본값”

6.3. DNS 리소스 할당량을 기본값으로 재설정

DNS 서비스(designate)를 사용하여 RHOSP(Red Hat OpenStack Platform) 프로젝트의 DNS 리소스 할당량을 기본값으로 재설정할 수 있습니다.

사전 요구 사항

- admin 역할이 있는 RHOSP 사용자여야 합니다.

프로세스

1. 클라우드 관리자로 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. 재설정할 할당량이 있는 프로젝트의 ID를 가져옵니다.

   나중에 단계를 수행하는 데 필요하므로 ID를 기억할 수 있습니다.

   ```
   $ openstack project list
   ```

3. 프로젝트 ID를 사용하여 프로젝트의 DNS 리소스 할당량을 재설정합니다.

   예제

   이 예에서 ID가 ecd4341280d645e5959d32a4b7659da1 인 프로젝트의 할당량이 기본값으로 재설정됩니다.

   ```
   $ openstack dns quota reset --project-id ecd4341280d645e5959d32a4b7659da1
   ```

- 성공적인 **openstack dns quota reset** 명령의 출력이 없습니다.

검증

- 프로젝트의 **DNS** 리소스 할당량이 재설정되었는지 확인합니다.

예제

```
$ openstack dns quota list --project-id ecd4341280d645e5959d32a4b7659da1
```

샘플 출력

```
+-------------------+-------+
| Field             | Value |
+-------------------+-------+
| api_export_size   | 1000  |
| recordset_records | 20    |
| zone_records      | 500   |
| zone_recordsets   | 500   |
| zones             | 10    |
+-------------------+-------+
```

추가 리소스

- *명령줄 인터페이스 참조에서 **DNS** 할당량 재설정*
- **6.4절. “DNS 서비스 할당량 및 기본값”**

6.4. DNS 서비스 할당량 및 기본값

RHOSP(Red Hat OpenStack Platform) DNS 서비스(**designate**)에는 클라우드 관리자가 하나 또는 모

든 RHOSP 프로젝트에서 클라우드 사용자가 사용하는 DNS 리소스 사용을 제한하도록 설정할 수 있는 할당량이 있습니다.

표 6.1. 영역 할당량

할당량	Default	설명
영역	10	프로젝트당 허용된 영역 수입니다.

표 6.2. 레코드 및 레코드 세트 할당량

할당량	Default	설명
zone_recordsets	500	영역당 허용되는 레코드 세트 수입니다.
zone_records	500	영역당 허용된 레코드 수입니다.
recordset_records	20	레코드 세트당 허용된 레코드 수입니다.

표 6.3. 영역 내보내기 할당량

할당량	Default	설명
api_export_size	1000	영역 내보내기에 허용되는 레코드 세트 수입니다.

7장. 영역 관리

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)는 영역을 사용하여 네임스페이스를 쉽게 관리할 수 있는 부분으로 나눕니다. 사용자는 RHOSP 프로젝트에서 영역을 소유하고 있는 경우 영역을 생성, 수정, 삭제, 내보내기 및 가져올 수 있습니다.

이 섹션에 포함된 항목은 다음과 같습니다.

7.1. DNS 서비스의 영역

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)는 두 가지 주요 차이점이 있는 유사한 영역 소유권 모델을 DNS로 사용합니다.

예를 들어 DNS에서는 루트 영역(..) 내에 .org. 및 . com과 같은 각 최상위 도메인(TLD)의 영역이 있습니다. TLD 영역 내에는 example.org. 또는 example. com과 같은 다른 영역에 위임이 있을 수 있습니다.

다른 조직(또는 다른 이름 서버 집합)에서 소유하고 관리할 수 있습니다. 이 예제에서는 높은 수준의 영역이 하위 수준 영역에 대한 위임의 대부분으로 구성된 역할의 계층 구조를 보여줍니다.

RHOSP DNS 서비스를 사용하는 DNS와 유사하게 영역을 하나의 테넌트만 소유할 수 있습니다. 그러나 DNS 서비스와 달리 DNS 서비스는 테넌트 간 영역 위임을 지원하지 않습니다. 즉, 테넌트는 다른 테넌트에서 상위 영역을 소유한 하위 영역을 만들 수 없습니다.

DNS와 RHOSP DNS 서비스의 두 번째 차이점은 DNS 서비스가 영역과 다른 TLD를 관리하는 것입니다. DNS 서비스는 테넌트가 관리형 TLD 내에 없는 영역을 생성하지 못하도록 제한합니다. DNS 서비스에서 TLD를 관리하지 않으면 테넌트는 루트 영역 이외의 TLD 및 모든 영역을 만들 수 있습니다.

7.2. 영역 생성

영역을 사용하면 네임스페이스를 더 쉽게 관리할 수 있습니다. 기본적으로 모든 사용자는 RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate) 영역을 생성할 수 있습니다.

사전 요구 사항

- RHOSP 프로젝트는 하위 영역을 생성하는 영역을 소유해야 합니다. 그렇지 않으면 영역이 허용되는 TLD여야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. 영역의 이름과 영역을 담당하는 사람의 이메일 주소를 지정하여 영역을 생성합니다.

   예제

   -

```
$ openstack zone create --email dnsprimary@example.com example.com.
```

영역을 생성할 때 **DNS** 서비스는 **SOA** 레코드와 **NS** 레코드라는 두 개의 레코드 세트를 자동으로 생성합니다.

검증

- **openstack zone list** 명령을 실행하여 영역이 있는지 확인합니다.

샘플 출력

```
+--------------------------------------+-------------+---------+------------+--------+--------+
| id                                   | name        | type    |     serial | status | action |
+--------------------------------------+-------------+---------+------------+--------+--------+
| 14093115-0f0f-497a-ac69-42235e46c26f | example.com. | PRIMARY | 1468421656 |
ACTIVE | NONE   |
+--------------------------------------+-------------+---------+------------+--------+--------+
```

추가 리소스

- *명령줄 인터페이스 참조*에서 **zone create**
- *명령줄 인터페이스 참조*의 영역 목록

7.3. 영역 업데이트

RHOSP(Red Hat OpenStack Platform) DNS 서비스(**designate**)에서 관리하는 영역을 업데이트해야 하는 경우가 있을 수 있습니다. 예를 들어 영역과 연결된 이메일 주소를 변경하거나 영역 **TTL(Time to live)** 값을 변경하려는 경우입니다. 기본적으로 모든 사용자는 영역을 수정할 수 있습니다.

사전 요구 사항

- **RHOSP** 프로젝트는 수정하려는 영역을 소유해야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. 영역의 이름과 변경할 영역 속성을 지정하여 영역을 수정합니다.

   --email <email_address>

   영역에 대한 소유자(소유자)에 대한 유효한 이메일 주소입니다.

   --ttl <seconds>

   (Time To Live) DNS client-(예: 확인자, 웹 브라우저), 운영 체제가 업데이트되었는지 확인하기 전에 레코드를 캐시할 수 있는 기간(초)입니다.

   --description <string> | --no-description

   영역의 목적을 설명하는 문자열입니다.

   --masters <dns-server> [<dns-server> ...]

   기본 인스턴스인 **DNS** 서버의 정규화된 도메인 이름입니다. 다른 **DNS** 서버에서 보조 서버로 동기화할 수 있는 인스턴스입니다.

   예제

   ```
   $ openstack zone set example.com. --ttl 3000
   ```

검증

- 영역을 수정한 후 성공했는지 확인합니다.

예제

```
$ openstack zone show example.com.
```

추가 리소스

- *명령줄 인터페이스 참조에* 설정된 영역
- *명령줄 인터페이스 참조에* **zone** 표시

7.4. 영역 삭제

RHOSP(Red Hat OpenStack Platform) DNS 서비스(**designate**)에서 관리하는 영역을 제거할 수 있습니다. 예를 들어 영역 이름이 변경된 경우 영역을 삭제합니다.

사전 요구 사항

- **RHOSP** 프로젝트는 삭제할 영역을 소유해야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. 영역을 삭제합니다.

예제

```
$ openstack zone delete example.com.
```

검증

- openstack zone list 명령을 실행하여 영역이 더 이상 존재하지 않는지 확인합니다.

추가 리소스

- *명령줄 인터페이스 참조*의 zone delete
- *명령줄 인터페이스 참조*의 영역 목록

7.5. 영역 내보내기

RHOSP(Red Hat OpenStack Platform) DNS 서비스에서 영역 데이터를 내보내는 것은 기본적으로 DNS 서비스가 내부적으로 저장하는 영역 내보내기 리소스를 생성하는 것으로 구성됩니다. 예를 들어 designate://v2/zones/tasks/exports/e75aef2c-b562-4cd9-a426-4a73f6cb82be/export 입니다. 영역 내보내기 데이터 리소스를 생성한 후 해당 콘텐츠에 액세스할 수 있습니다. 영역 데이터 내보내기는 RHOSP 배포를 위한 DNS 정보를 보호하기 위한 전체 백업 전략의 일부입니다.

사전 요구 사항

- RHOSP 프로젝트는 데이터를 내보내는 영역을 소유해야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. 영역을 내보냅니다.

예제

```
$ openstack zone export create example.com.
```

샘플 출력

```
+------------+--------------------------------------+
| Field      | Value                                |
+------------+--------------------------------------+
| created_at | 2022-02-11T02:01:30.000000           |
| id         | e75aef2c-b562-4cd9-a426-4a73f6cb82be |
| location   | None                                 |
| message    | None                                 |
| project_id | cf5a8f5cc5834d2dacd1d54cd0a354b7     |
| status     | PENDING                              |
| updated_at | None                                 |
| version    | 1                                    |
| zone_id    | d8f81db6-937b-4388-bfb3-ba620e6c09fb |
+------------+--------------------------------------+
```

중요

영역 내보내기 리소스를 생성한 후 **DNS** 서비스는 나중에 영역에 적용된 변경 사항으로 리소스를 계속 업데이트합니다.

3. 영역 내보내기 **ID**를 사용하고 영역 내보내기 데이터에 액세스해야 하므로 영역 내보내기 **ID(e75aef2c-b562-4cd9-a426-4a73f6cb82be)**를 기록합니다.

검증

1. **DNS** 서비스에서 영역 내보내기 리소스를 성공적으로 생성했는지 확인합니다.

예제

```
$ openstack zone export show e75aef2c-b562-4cd9-a426-4a73f6cb82be
```

샘플 출력

```
+------------+--------------------------------------------------------------------------------+
| Field      | Value                                                                          |
+------------+--------------------------------------------------------------------------------+
| created_at | 2022-02-11T02:01:30.000000                                                     |
| id         | e75aef2c-b562-4cd9-a426-4a73f6cb82be                                           |
| location   | designate://v2/zones/tasks/exports/e75aef2c-b562-4cd9-a426-
4a73f6cb82be/export |
| message    | None                                                                           |
| project_id | cf5a8f5cc5834d2dacd1d54cd0a354b7                                               |
| status     | COMPLETE                                                                       |
| updated_at | 2022-02-11T02:01:30.000000                                                     |
| version    | 2                                                                              |
| zone_id    | d8f81db6-937b-4388-bfb3-ba620e6c09fb                                           |
+------------+--------------------------------------------------------------------------------+
```

zone export create 명령은 **DNS** 서비스가 기본적으로 내부적으로 저장하는 리소스를 생성

합니다.

2. 이전에 가져온 영역 내보내기 **ID**를 사용하여 영역 내보내기 파일의 콘텐츠에 액세스합니다.

작은 정보

-f value 옵션을 사용하면 **tabulation** 없이 영역 파일의 내용을 출력합니다. 로컬에서 내보낸 영역 파일을 수정하려는 경우 유용할 수 있는 로컬 텍스트 파일로 콘텐츠를 리디렉션한 다음 **DNS** 서비스로 가져와 영역을 업데이트할 수도 있습니다.

예제

```
$ openstack zone export showfile e75aef2c-b562-4cd9-a426-4a73f6cb82be -f value
```

샘플 출력

```
$ORIGIN example.com.
$TTL 3600

example.com.  IN NS ns1.example.com.
example.com.  IN SOA ns1.example.com. admin.example.com. 1624414033 3583 600 86400
3600

www.example.com.  IN A 192.0.2.2
www.example.com.  IN A 192.0.2.1
```

추가 리소스

- 영역 파일 형식:
  - RFC1034, 섹션 3.6

- RFC1035, 섹션 5.1

- *명령줄 인터페이스 참조의* 영역 내보내기 생성

- *명령줄 인터페이스 참조에* 영역 내보내기 표시

- *명령줄 인터페이스 참조 의 영역 내보내기 표시*

7.6. 영역 가져오기

영역 데이터를 **RHOSP(Red Hat OpenStack Platform) DNS** 서비스로 가져오는 것은 **openstack zone import** 명령을 실행하여 **openstack zone data file** 형식을 준수하는 파일에서 **openstack zone import** 명령을 실행하여 **openstack zone export showfile** 명령으로 생성된 데이터와 같이 **DNS** 영역 데이터 파일 형식을 준수하는 것으로 구성됩니다. 데이터를 가져오는 한 가지 이유는 사용자가 영역을 실수로 삭제하는 경우입니다.

사전 요구 사항

- **RHOSP** 프로젝트는 하위 영역을 생성하는 영역을 소유해야 합니다. 그렇지 않으면 영역이 허용되는 **TLD**여야 합니다.

- 가져오는 영역이 아직 존재하지 않아야 합니다.

- 가져오는 영역 데이터에 영역 **TTL(Time to live)** 값이 포함되어야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

```
$ source ~/overcloudrc
```

2. 시스템의 영역을 나열합니다.

```
$ openstack zone list
```

3. 가져올 영역이 이미 존재하는 경우 **openstack zone delete** 명령을 실행하여 먼저 삭제해야 합니다.

예제

```
$ openstack zone delete example.com.
```

4. 시스템의 영역을 나열하여 영역이 더 이상 존재하지 않는지 확인합니다.

```
$ openstack zone list
```

5. 가져올 영역 데이터에 영역 **TTL** 값이 포함되어 있는지 확인합니다.

예제

```
$ cat /home/stack/zone_file
```

샘플 출력

```
$ORIGIN example.com.
$TTL 3000

example.com.  IN NS test.example.com.
```

```
example.com.  IN SOA test.example.com. admin.example.com. 1624415706 9000 500 86000
5000
www.example.com.  IN A 192.0.2.2
test.example.com.  IN NS test.example.com.
```

6. **유효한 영역 데이터 파일을 가져옵니다.**

예제

```
$ openstack zone import create /home/stack/zone_file
```

검증

- **DNS 서비스가 영역을 가져왔는지 확인합니다.**

예제

```
$ openstack recordset list -c name -c type -c records -c status example.com.
```

샘플 출력

```
+------------------+------+-----------------------------------------------------------------------+--------+
| name             | type | records                                                               | status |
+------------------+------+-----------------------------------------------------------------------+--------+
| example.com.     | SOA  | ns1.example.com. admin.example.com. 1624415706 3582 500
86000 3600  | ACTIVE |
| test.example.com. | NS   | test.example.com.                                                     | ACTIVE |
| example.com.     | NS   | ns1.example.com.                                                      | ACTIVE |
| www.example.com.  | A    | 192.0.2.2                                                             | ACTIVE |
+------------------+------+-----------------------------------------------------------------------+--------+
```

추가 리소스

- 영역 파일 형식:
  - RFC1034, 섹션 3.6
  - RFC1035, 섹션 5.1
- *명령줄 인터페이스 참조*에서 영역 가져오기 생성
- *명령줄 인터페이스 참조*의 영역 목록

7.7. 영역 소유권 전송

영역의 소유권을 한 프로젝트에서 다른 프로젝트로 이전할 수 있습니다. 예를 들어, **Finance** 팀은 **wow.example.com.** 영역의 소유권을 해당 프로젝트에서 **Sales** 팀의 프로젝트로 이전하려고 할 수 있습니다.

클라우드 관리자의 개입 없이 영역의 소유권을 이전할 수 있습니다. 그러나 현재 프로젝트 영역 소유자와 수신 프로젝트 멤버 모두 전송에 동의해야 합니다.

사전 요구 사항

- 프로젝트는 전송할 영역을 소유해야 합니다.
- 전송 요청을 생성한 후 수신 프로젝트의 구성원은 전송하는 영역을 수락해야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. **영역의 소유권을 전송할 프로젝트의 ID를 가져옵니다.**

예제

```
$ openstack project list
```

샘플 출력

```
+----------------------------------+--------------------+
| ID                               | Name               |
+----------------------------------+--------------------+
| 7af0acba0486472da2447ff55df4a26d | Finance            |
| 1d12e87fad0d437286c2873b36a12316 | Sales              |
+----------------------------------+--------------------+
```

3. **이전 단계에서 얻은 프로젝트 ID를 사용하여 전송할 영역에 대한 영역 전송 요청을 생성합니다.**

참고

대상 프로젝트 ID를 사용하는 경우 다른 프로젝트에서 영역 전송을 수락할 수 없습니다. 대상 프로젝트 ID를 제공하지 않으면 전송 요청 ID가 있는 모든 프로젝트에서 해당 키가 영역 전송을 수신할 수 있습니다.

예제

wow.example.com. 영역을 **1d12e87fad0d437286c2873b36b12316** 으로 전송하려면 다음을 실행합니다.

```
$ openstack zone transfer request create --target-project-id
1d12e87fad0d437286c2873b36a12316 wow.example.com.
```

샘플 출력

```
+------------------+-----------------------------------------------------+
| Field            | Value                                               |
+------------------+-----------------------------------------------------+
| created_at       | 2022-05-26T22:06:39.000000                          |
| description      | None                                                |
| id               | 63cab5e5-65fa-4480-b26c-c16c267c44b2                |
| key              | BIFJIQWH                                            |
| links            | {'self': 'http://127.0.0.1:60053/v2/zones/tasks/tra |
|                  | nsfer_requests/63cab5e5-65fa-4480-b26c-c16c267c44b2 |
|                  | '}                                                  |
| project_id       | 6265985fc493465db6a978b318a01996                    |
| status           | ACTIVE                                              |
| target_project_id | 1d12e87fad0d437286c2873b36a12316                   |
| updated_at       | None                                                |
| zone_id          | 962f08b4-b671-4096-bf24-8908c9d4af0c                |
| zone_name        | wow.example.com.                                    |
+------------------+-----------------------------------------------------+
```

4. **영역 전송 요청 ID와 해당 키를 가져옵니다.**

예제

```
$ openstack zone transfer request list -c id -c zone_name -c key
```

샘플 출력

```
+--------------------------------------+-----------------+----------+
| id                                   | zone_name       | key      |
+--------------------------------------+-----------------+----------+
| 63cab5e5-65fa-4480-b26c-c16c267c44b2 | wow.example.com. | BIFJIQWH |
+--------------------------------------+-----------------+----------+
```

5. **수신 중인 프로젝트의 멤버에게 영역 전송 요청 ID와 해당 키를 제공합니다.**

6. **수신되는 프로젝트 구성원은 수신 프로젝트에 로그인하고 해당 자격 증명 파일을 가져옵니다.**

예제

```
$ source ~/overcloudrc
```

7. **영역 전송 요청 ID와 해당 키를 사용하여 영역 전송을 수락합니다.**

예제

```
$ openstack zone transfer accept request --transfer-id 63cab5e5-65fa-4480-b26c-
c16c267c44b2 --key BIFJIQWH
```

샘플 출력

```
+-------------------------+------------------------------------------+
| Field                   | Value                                    |
+-------------------------+------------------------------------------+
| created_at              | 2022-05-27T21:37:43.000000               |
| id                      | a4c4f872-c98c-411b-a787-58ed0e2dce11     |
| key                     | BIFJIQWH                                 |
```

```
| links                  | {'self': 'http://127.0.0.1:60053/v2/zones/ta |
|                        | sks/transfer_accepts/a4c4f872-c98c-411b-a787 |
|                        | -58ed0e2dce11', 'zone': 'http://127.0.0.1:60 |
|                        | 053/v2/zones/962f08b4-b671-4096-bf24-8908c9d |
|                        | 4af0c'}                                      |
| project_id             | 1d12e87fad0d437286c2873b36a12316             |
| status                 | COMPLETE                                     |
| updated_at             | 2022-05-27T21:37:43.000000                   |
| zone_id                | 962f08b4-b671-4096-bf24-8908c9d4af0c         |
| zone_transfer_request_id | 63cab5e5-65fa-4480-b26c-c16c267c44b2       |
+------------------------+----------------------------------------------+
```

검증

- **이전 단계의 영역 전송 수락 ID를 사용하여 영역 전송 상태를 확인합니다.**

예제

이 예에서 영역 상태 허용 ID는 a4c4f872-c98c-411b-a787-58ed0e2dce11 입니다.

```
$ openstack zone transfer accept show a4c4f872-c98c-411b-a787-58ed0e2dce11
```

샘플 출력

```
+------------------------+----------------------------------------------+
| Field                  | Value                                        |
+------------------------+----------------------------------------------+
| created_at             | 2022-05-27T21:37:43.000000                   |
| id                     | a4c4f872-c98c-411b-a787-58ed0e2dce11         |
| key                    | None                                         |
| links                  | {'self': 'http://127.0.0.1:60053/v2/zones/ta |
|                        | sks/transfer_accepts/a4c4f872-c98c-411b-a787 |
|                        | -58ed0e2dce11', 'zone': 'http://127.0.0.1:60 |
|                        | 053/v2/zones/962f08b4-b671-4096-bf24-8908c9d |
|                        | 4af0c'}                                      |
| project_id             | 1d12e87fad0d437286c2873b36a12316             |
| status                 | COMPLETE                                     |
| updated_at             | 2022-05-27T21:37:43.000000                   |
| zone_id                | 962f08b4-b671-4096-bf24-8908c9d4af0c         |
| zone_transfer_request_id | 63cab5e5-65fa-4480-b26c-c16c267c44b2       |
+------------------------+----------------------------------------------+
```

추가 리소스

- *명령줄 인터페이스 참조*의 영역 전송 요청 **create** 명령
- *명령줄 인터페이스 참조*의 영역 전송 수락 요청 명령

7.8. 영역 전송 요청 수정

한 프로젝트에서 다른 프로젝트로 영역의 소유권을 전송하는 첫 번째 단계는 영역 전송 요청을 생성하는 것입니다. 영역 전송 요청을 변경하거나 삭제해야 하는 경우 이를 수행할 수 있습니다.

사전 요구 사항

- 프로젝트는 수정 중인 전송 요청이 있는 영역을 소유해야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. 수정 중인 영역 전송 요청의 **ID**를 가져옵니다.

   예제

   ```
   $ openstack zone transfer request list -c id -c zone_name
   ```

샘플 출력

```
+--------------------------------------+-----------------+
| id                                   | zone_name       |
+--------------------------------------+-----------------+
| 63cab5e5-65fa-4480-b26c-c16c267c44b2 | wow.example.com. |
+--------------------------------------+-----------------+
```

3. **이전 단계에서 얻은 영역 전송 요청 ID를 사용하여 설명 및 대상 프로젝트 ID와 같은 영역 전송 요청에서 제한된 필드 세트를 업데이트할 수 있습니다.**

예제

```
$ openstack zone transfer request set --description "wow zone transfer" 63cab5e5-65fa-4480-b26c-c16c267c44b2
```

샘플 출력

```
+-------------------+--------------------------------------------------------+
| Field             | Value                                                  |
+-------------------+--------------------------------------------------------+
| created_at        | 2022-05-26T22:06:39.000000                             |
| description       | wow zone transfer                                      |
| id                | 63cab5e5-65fa-4480-b26c-c16c267c44b2                   |
| key               | BIFJIQWH                                               |
| links             | {'self': 'http://127.0.0.1:60053/v2/zones/tasks/tra |
|                   | nsfer_requests/63cab5e5-65fa-4480-b26c-c16c267c44b2 |
|                   | '}                                                     |
| project_id        | 6265985fc493465db6a978b318a01996                       |
| status            | ACTIVE                                                 |
| target_project_id | 1d12e87fad0d437286c2873b36a12316                       |
| updated_at        | 2022-05-27T20:52:08.000000                             |
```

```
| zone_id          | 962f08b4-b671-4096-bf24-8908c9d4af0c                |
| zone_name        | wow.example.com.                                    |
+------------------+-----------------------------------------------------+
```

4. **2단계에서 얻은 영역 전송 요청 ID를 사용하여 영역 전송 요청을 삭제하여 보류 중인 영역 전송을 취소할 수 있습니다.**

예제

```
$ openstack zone transfer request delete 63cab5e5-65fa-4480-b26c-c16c267c44b2
```

영역 전송 요청 삭제 명령의 출력이 없습니다. 영역 전송 요청 목록 명령을 실행하여 영역 전송 요청이 제거되었는지 확인합니다.

추가 리소스

- **7.7절. "영역 소유권 전송"**
- *명령줄 인터페이스 참조*의 **영역 전송 요청 set 명령**
- *명령줄 인터페이스 참조*의 **영역 전송 요청 삭제 명령**

8장. 레코드 세트 관리

RHOSP(Red Hat OpenStack) DNS 서비스(designate)는 레코드 세트에 대한 데이터를 저장합니다. 레코드 세트는 하나 이상의 DNS 리소스 레코드로 구성됩니다. 영역을 쿼리하여 추가, 수정 및 삭제 외에 해당 레코드 세트를 나열할 수 있습니다.

이 섹션에 포함된 항목은 다음과 같습니다.

8.1. DNS 서비스의 레코드 및 레코드 세트 정보

DNS(Domain Name System)는 리소스 레코드를 사용하여 네임스페이스 내에 영역 데이터를 저장합니다. RHOSP(Red Hat OpenStack) DNS 서비스(designate)의 DNS 레코드는 레코드 세트를 사용하여 관리합니다.

각 DNS 레코드에는 다음 속성이 포함되어 있습니다.

- **name** - DNS 네임스페이스에서 해당 위치를 나타내는 문자열입니다.
- **type** - 레코드 사용 방법을 식별하는 문자 코드 집합입니다. 예를 들어 **A** 는 주소 레코드를 식별하고 **CNAME** 은 표준 이름 레코드를 식별합니다.
- **class** - 레코드의 네임스페이스를 지정하는 문자 코드 집합입니다. 일반적으로 이것은 인터넷의 경우 **IN** 이지만 다른 네임스페이스는 존재합니다.

- **TTL - (Time to live)** 레코드가 유효한 기간(초)입니다.

- **R Data - A** 레코드의 **IP** 주소 또는 **CNAME** 레코드의 다른 레코드 이름과 같은 레코드의 데이터입니다.

각 영역 네임스페이스에는 권한 시작**(SOA)** 레코드가 포함되어야 하며 권한 있는 이름 서버**(NS)** 레코드와 다양한 다른 유형의 레코드가 있을 수 있습니다. **SOA** 레코드는 이 이름 서버가 영역에 대한 정보의 최상의 소스임을 나타냅니다. **NS** 레코드는 영역에 권한이 있는 이름 서버를 식별합니다. 영역의 **SOA** 및 **NS** 레코드는 읽을 수 있지만 수정할 수 없습니다.

필수 **SOA** 및 **NS** 레코드 외에도 가장 일반적인 레코드 유형 세 가지는 주소**(A),** 표준 이름**(CNAME),** 포인터**(PTR)** 레코드입니다. 레코드는 호스트 이름을 **IP** 주소에 매핑합니다. **PTR** 레코드는 **IP** 주소를 호스트 이름에 매핑합니다. **CNAME** 레코드는 별칭의 전체 호스트 이름을 식별합니다.

레코드 세트는 이름과 유형이 동일하지만 잠재적으로 데이터가 다를 수 있는 하나 이상의 **DNS** 레코드를 나타냅니다. 예를 들어 데이터 **192.0.2.1** 및 **192.0.2.2** 를 포함하는 **web.example.com** 이라는 레코드 세트는 두 개의 **IP** 주소에 있는 **web.example.com** 을 호스팅하는 두 개의 웹 서버를 반영할 수 있습니다.

영역 내에 레코드 세트를 생성해야 합니다. 레코드 세트가 포함된 영역을 삭제하면 영역 내의 해당 레코드 세트도 삭제됩니다.

openstack recordset list -c name -c type -c records example.com 명령을 사용하여 **example.com** 영역을 쿼리하여 얻은 이 출력을 고려하십시오.

```
+-----------------+------+----------------------------------------------+
| name            | type | records                                      |
+-----------------+------+----------------------------------------------+
| example.com.    | SOA  | ns1.example.net. admin.example.com. 16200126 |
|                 |      | 16 3599 600 8640 0 3600                      |
|                 |      |                                              |
| example.com.    | NS   | ns1.example.net.                             |
|                 |      |                                              |
| web.example.com. | A    | 192.0.2.1                                    |
|                 |      | 192.0.2.2                                    |
|                 |      |                                              |
| www.example.com. | A    | 192.0.2.1                                    |
+-----------------+------+----------------------------------------------+
```

이 예에서 **example.com. zone**에 대한 권한 있는 이름 서버는 **ns1.example.net., NS** 레코드입니다. 이를 확인하려면 **BIND dig** 툴을 사용하여 **NS** 레코드의 이름 서버를 쿼리할 수 있습니다.

```
$ dig @ns1.example.net example.com. -t NS +short
ns1.example.net.
```

A 레코드 세트를 확인할 수도 있습니다.

```
$ dig @ns1.example.net web.example.com. +short
192.0.2.2
192.0.2.1
$ dig @ns1.example.net www.example.com. +short
192.0.2.1
```

8.2. 레코드 세트 생성

기본적으로 모든 사용자는 **Red Hat OpenStack Platform DNS** 서비스(**designate**) 레코드 세트를 생성할 수 있습니다.

사전 요구 사항

- 프로젝트에서 레코드 세트를 생성하는 영역을 소유해야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. **openstack recordset create** 명령을 사용하여 레코드 세트를 생성합니다. 레코드 세트에는 영역, 이름, 유형 및 데이터가 필요합니다.

   예제

   ```
   $ openstack recordset create --type A --record 192.0.2.1 example.com. www
   ```

참고

FQDN(정규화된 도메인 이름)을 사용하는 경우 후행 점(..)이 필요합니다. 후행 점을 생략하면 결과 레코드 이름에 영역 이름이 복제됩니다(예: www.example.com.example.com).

이전 예에서 사용자가 example.com. 이라는 영역을 생성했습니다. 레코드 세트 이름 ExternalIP 는 FQDN이 아니므로 DNS 서비스 앞에 영역 이름 앞에 추가합니다. 레코드 세트 이름 인수에 FQDN을 사용하여 동일한 결과를 얻을 수 있습니다.

```
$ openstack recordset create --type A --record 192.0.2.1 example.com. www.example.com.
```

3. 문자 문자열(255자)의 최대 길이를 초과하는 TXT 레코드 세트를 구성하려면 레코드 세트를 만들 때 문자열을 여러 작은 문자열로 분할해야 합니다.

이 예제에서 사용자는 255자 최대값보다 두 개의 문자열을 지정하여 410자 문자열을 포함하는 TXT 레코드 세트(_domainkey.example.com)를 생성합니다.

```
$ openstack recordset create --type TXT --record '"210 characters string" "200 characters string"' example.com. _domainkey
```

4. --record 인수를 여러 번 제공하여 레코드 세트 내에서 여러 레코드를 만들 수 있습니다. 여러 --record 인수에 일반적인 사용은 라운드 로빈 DNS입니다.

예제

```
$ openstack recordset create --type A --record 192.0.2.1 --record 192.0.2.2 example.com. web
```

검증

- list 명령을 실행하여 생성한 레코드 세트가 있는지 확인합니다.

예제

```
$ openstack recordset list -c name -c type -c records example.com.
```

샘플 출력

```
+-----------------+------+-----------------------------------------+
| name            | type | records                                 |
+-----------------+------+-----------------------------------------+
| example.com.    | SOA  | ns1.example.net. admin.example.com 162001261 |
|                 |      | 6 3599 600 86400 3600                   |
|                 |      |                                         |
| example.com.    | NS   | ns1.example.net.                        |
|                 |      |                                         |
| web.example.com. | A    | 192.0.2.1 192.0.2.2                     |
|                 |      |                                         |
| www.example.com. | A    | 192.0.2.1                               |
+-----------------+------+-----------------------------------------+
```

추가 리소스

- *명령줄 인터페이스 참조*의 **recordset create** 명령
- *명령줄 인터페이스 참조*의 **recordset list** 명령
- **dig**도움말 페이지

8.3. 레코드 세트 업데이트

기본적으로 모든 사용자는 **Red Hat OpenStack Platform DNS** 서비스(**designate**) 레코드 세트를 업데이트할 수 있습니다.

사전 요구 사항

- 프로젝트에서 레코드 세트를 업데이트하는 영역을 소유해야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. **openstack recordset set** 명령을 사용하여 레코드 세트를 수정합니다.

   예제

   이 예에서 사용자는 레코드 세트 **web.example.com.** 을 업데이트하여 두 개의 레코드를 포함합니다.

   ```
   $ openstack recordset set example.com. web.example.com. --record 192.0.2.5 --record 192.0.2.6
   ```

   참고

   레코드 세트를 업데이트할 때 **ID** 또는 해당 이름으로 식별할 수 있습니다. 해당 이름을 사용하는 경우 **FQDN**(정규화된 도메인 이름)을 사용해야 합니다.

검증

- **list** 명령을 실행하여 수정 사항을 확인합니다.

  예제

  ```
  $ openstack recordset list -c name -c type -c records example.com.
  ```

샘플 출력

```
+------------------+------+-----------------------------------------+
| name             | type | records                                 |
+------------------+------+-----------------------------------------+
| example.com.     | SOA  | ns1.example.net. admin.example.com 162001261 |
|                  |      | 6 3599 600 86400 3600                   |
|                  |      |                                         |
| example.com.     | NS   | ns1.example.net.                        |
|                  |      |                                         |
| web.example.com. | A    | 192.0.2.5 192.0.2.6                     |
|                  |      |                                         |
| www.example.com. | A    | 192.0.2.1                               |
+------------------+------+-----------------------------------------+
```

추가 리소스

- *명령줄 인터페이스 참조*의 **recordset create** 명령
- *명령줄 인터페이스 참조*의 **recordset list** 명령

8.4. 레코드 세트 삭제

기본적으로 모든 사용자는 **Red Hat OpenStack Platform DNS 서비스(designate)** 레코드 세트를 삭제할 수 있습니다.

사전 요구 사항

- 프로젝트에서 레코드 세트를 삭제하는 영역을 소유해야 합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. **openstack recordset delete** 명령을 사용하여 레코드 세트를 삭제합니다.

예제

이 예에서 사용자는 **example. com.** 영역에서 레코드 세트 **web.example.com**을 삭제하고 있습니다.

```
$ openstack recordset delete example.com. web.example.com.
```

검증

- **list** 명령을 실행하여 삭제를 확인합니다.

예제

```
$ openstack recordset list -c name -c type -c records example.com.
```

샘플 출력

```
+-----------------+------+---------------------------------------------+
| name            | type | records                                     |
+-----------------+------+---------------------------------------------+
| example.com.    | SOA  | ns1.example.net. admin.example.com 162001261 |
|                 |      | 6 3599 600 86400 3600                       |
|                 |      |                                             |
| example.com.    | NS   | ns1.example.net.                            |
|                 |      |                                             |
| www.example.com. | A    | 192.0.2.1                                   |
+-----------------+------+---------------------------------------------+
```

추가 리소스

- *명령줄 인터페이스 참조*의 **recordset delete** 명령
- *명령줄 인터페이스 참조*의 **recordset list** 명령

9장. 포인터 레코드 관리 (PTR)

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)를 구성하는 단계는 역방향 조회라고도 하는 IP address-to-domain-name-lookups를 설정하는 것입니다. DNS 리소스인 포인터(PTR) 레코드는 address-to-name 매핑 데이터를 포함하며 역방향 조회 영역에 저장됩니다. DNS 서비스를 사용하면 유동 IP 주소에 대한 역방향 조회를 관리할 수도 있습니다.

이 섹션에 포함된 항목은 다음과 같습니다.

9.1. PTR 레코드 기본 사항

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)에서 포인터(PTR) 레코드를 사용하여 단일 IP 또는 일련의 IP 주소에서 FQDN(정규화된 도메인 이름)으로 이름 매핑(반복 매핑)을 생성합니다. DNS(Domain Name System)는 주소를 이름으로 검색하므로 IP 주소 이름이 포함된 PTR 레코드를 생성합니다. 특정 규칙에 따라 이 이름을 형성합니다. IP 주소를 되돌리고 IPv4 주소의 경우 in-addr.arpa, IPv6 주소의 경우 ip6.arpa 를 추가합니다.

예를 들어 my-server.example.com 의 IP 주소가 198.51.100.42 인 경우 역방향 조회 영역인 42.100.51.198.in-addr.arpa 에서 해당 노드의 이름을 지정합니다. 표준 FQDN(정규화된 도메인 이름)과 마찬가지로 역방향 IP 주소가 왼쪽에서 오른쪽으로 이동하는 대로 IP 주소를 역순으로 표시하므로 IP 주소의 이름을 역순으로 쉽게 표시할 수 있습니다.

DNS 서비스는 PTR 레코드의 내용을 역방향 조회 영역이라는 특수 영역에 씁니다. 이 영역의 유일한 용도는 address-to-name 조회를 제공하는 것입니다. PTR 레코드에는 표준 FQDN과 유사하게 구성된 데이터가 포함되어 있으므로 다른 영역을 위임하는 것과 동일한 방식으로 역방향 조회 영역의 하위 영역을 위임할 수 있습니다. 이전 예에서 198.51.100.42 호스트는 198.in-addr.arpa 영역의 노드이며 이 영역은 네트워크 관리자 198.51.100.0/8 에게 위임될 수 있습니다.

사용자의 RHOSP 프로젝트에서 IP 주소가 포함된 영역을 소유해야 하므로 DNS 서비스는 표준 IP 주소와 다른 부동 IP 주소에 대한 PTR 레코드를 관리합니다. 역방향 이름 조회와 관련된 대부분의 사용 사례에서는 이 요구 사항을 쉽게 충족할 수 있습니다. 표준 IP 주소에 대한 역방향 조회를 관리할 때 다른 DNS 리소스 레코드 유형을 관리할 때 openstack recordset 명령을 사용합니다.

그러나 유동 IP 주소로 작업할 때는 여러 프로젝트에서 유동 IP 주소 풀을 공유하는 것이 일반적입니다. 공유 주소 풀의 프로젝트 소유권 문제를 해결하려면 openstack ptr record 명령 유동 IP의 역방향 조회를 관리할 때 다른 명령을 사용해야 합니다.

추가 리소스

- 9.3절. “PTR 레코드 생성”
- 9.5절. “부동 IP 주소에 대한 PTR 레코드 설정”

9.2. 역방향 조회 영역 생성

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)를 올바르게 구성하려면 역방향 조회 영역이 있어야 합니다. 역방향 조회 영역에는 address-to-name 조회를 수행하는 데 필요한 PTR 레코드가 포함되어 있습니다. 이 규칙에 따라 역방향 조회 영역의 이름을 지정해야 합니다. < backward_IP_address>.in-addr.arpa 는 IPv4 주소의 경우 < backward_IP_address>.ip6.arpa 입니다.

일반적으로 RHOSP 배포의 영역을 서브넷 계획에 맞춥니다. 예를 들어 외부 네트워크의 /24 서브넷이 있는 경우 PTR 레코드를 포함하도록 /24 서브넷 역방향 조회 영역을 생성합니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. **openstack zone create** 명령을 사용하고 필요한 인수를 지정하여 역방향 조회 영역을 생성합니다.

--email <email_address>

영역에 대한 소유자(소유자)에 대한 유효한 이메일 주소입니다.

<name>

규칙을 준수하는 역방향 조회 영역의 이름입니다. **< backward_IP_address>.in-addr.arpa** 는 **IPv4** 주소의 경우, **< backward_IP_address>.ip6.arpa** 입니다.

예제

이 예에서 역방향 조회 영역은 **198.51.100.42** 주소에 대해 하나의 **PTR** 레코드용으로 설계되었습니다.

```
$ openstack zone create --email admin@example.com \
  42.100.51.198.in-addr.arpa.
```

샘플 출력

```
+----------------+-----------------------------------------+
| Field          | Value                                   |
+----------------+-----------------------------------------+
| action         | CREATE                                  |
| attributes     |                                         |
| created_at     | 2022-02-02T17:32:47.000000              |
| description    | None                                    |
| email          | admin@example.com                       |
| id             | f5546034-b27e-4326-bf9d-c53ed879f7fa    |
| masters        |                                         |
| name           | 42.100.51.198.in-addr.arpa.             |
| pool_id        | 794ccc2c-d751-44fe-b57f-8894c9f5c842    |
| project_id     | 123d51544df443e790b8e95cce52c285        |
| serial         | 1591119166                              |
| status         | PENDING                                 |
| transferred_at | None                                    |
| ttl            | 3600                                    |
| type           | PRIMARY                                 |
```

```
| updated_at     | None                                 |
| version        | 1                                    |
+----------------+--------------------------------------+
```

예제

198.51.100.0/24 서브넷의 역방향 영역의 경우 영역을 생성합니다.

```
$ openstack zone create --email admin@example.com \
  100.51.198.in-addr.arpa.
```

샘플 출력

```
+----------------+--------------------------------------+
| Field          | Value                                |
+----------------+--------------------------------------+
| action         | CREATE                               |
| attributes     |                                      |
| created_at     | 2022-02-02T17:40:23.000000           |
| description    | None                                 |
| email          | admin@example.com                    |
| id             | 5669caad86a04256994cdf755df4d3c1     |
| masters        |                                      |
| name           | 100.51.198.in-addr.arpa.             |
| pool_id        | 794ccc2c-d751-44fe-b57f-8894c9f5c842 |
| project_id     | 123d51544df443e790b8e95cce52c285     |
| serial         | 1739276248                           |
| status         | PENDING                              |
| transferred_at | None                                 |
| ttl            | 3600                                 |
| type           | PRIMARY                              |
| updated_at     | None                                 |
| version        | 1                                    |
+----------------+--------------------------------------+
```

검증

1. **생성한 역방향 조회 영역이 있는지 확인합니다.**

```
$ openstack zone list -c id -c name -c status
```

샘플 출력

```
+--------------------------------------+----------------------------+--------+
| id                                   | name                       | status |
+--------------------------------------+----------------------------+--------+
| f5546034-b27e-4326-bf9d-c53ed879f7fa | 42.100.51.198.in-addr.arpa. | ACTIVE |
+--------------------------------------+----------------------------+--------+
```

2. **address-to-name** 매핑을 완료하려면 **IP** 주소가 포함된 영역이 있어야 합니다. 정방향 영역이 없는 경우 지금 해당 영역을 생성합니다.

추가 리소스

- 영역 생성
- *명령줄 인터페이스 참조*에서 **zone create**

9.3. PTR 레코드 생성

RHOSP(Red Hat OpenStack Platform) DNS 서비스(**designate**)에서 **PTR** 레코드를 생성하여 역방향 조회(**address-to-name mappings**)를 활성화합니다. 역방향 조회를 활성화하는 것은 **RHOSP** 배포에서 **DNS** 서비스를 올바르게 구성하는 과정의 일부입니다.

사전 요구 사항

- **RHOSP** 프로젝트는 **PTR** 레코드를 생성하는 영역을 소유해야 합니다.
- **PTR** 레코드를 저장할 역방향 조회 영역입니다. 자세한 내용은 **9.2**절. “역방향 조회 영역 생성”의 내용을 참조하십시오.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. **openstack recordset create** 명령을 사용하고 이러한 필수 인수를 지정하여 **PTR** 레코드를 생성합니다.

--record <domain_name>

역방향 조회를 수행할 때 반환해야 하는 도메인 이름 대상입니다.

--type PTR

생성 중인 레코드 **PTR** 입니다.

<zone_name>

레코드가 있는 영역의 이름, 역방향 조회 영역입니다.

<record_name>

PTR 레코드의 이름입니다.

레코드 이름은 **<zone_name>**과 일치하거나 영역의 멤버여야 합니다. 예를 들어 역방향 조회 영역 **100.51.198.in-addr.arpa**의 경우 유효한 **PTR** 레코드 이름 **1. 100.51.198.in-addr.arpa. , 2.100.51.198.in-addr.arpa., 198.51.100.0/24** 서브넷의 기타 역방향 **IP** 주소입니다.

예제

```
openstack recordset create --record www.example.com. --type PTR \
42.100.51.198.in-addr.arpa. 42.100.51.198.in-addr.arpa.
```

샘플 출력

```
+-------------+--------------------------------------+
| Field       | Value                                |
+-------------+--------------------------------------+
| action      | CREATE                               |
| created_at  | 2022-02-02T19:55:50.000000           |
| description | None                                 |
| id          | ca604f72-83e6-421f-bf1c-bb4dc1df994a |
| name        | 42.100.51.198.in-addr.arpa.          |
| project_id  | 123d51544df443e790b8e95cce52c285     |
| records     | www.example.com.                     |
| status      | PENDING                              |
| ttl         | 3600                                 |
| type        | PTR                                  |
| updated_at  | None                                 |
| version     | 1                                    |
| zone_id     | f5546034-b27e-4326-bf9d-c53ed879f7fa |
| zone_name   | 42.100.51.198.in-addr.arpa.          |
+-------------+--------------------------------------+
```

검증

- 역방향 조회를 수행하여 **IP** 주소**(198.51.100.42)**가 도메인 이름**(www.example.com)**에 매핑되었는지 확인합니다**.**

예제

이 예에서 **203.0.113.5** 는 배포의 **DNS** 서버 중 하나입니다**.**

```
$ dig @203.0.113.5 -x 198.51.100.42 +short
```

샘플 출력

```
www.example.com.
```

추가 리소스

추가 리소스

- *명령줄 인터페이스 참조*에서 **recordset 생성**
- **dig** 명령 도움말 페이지.

9.4. 여러 PTR 레코드 생성

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)에서 보다 광범위하게 정의된 역방향 조회 영역을 사용하여 대규모 서브넷에 많은 **PTR** 레코드를 추가할 수 있습니다.

사전 요구 사항

- **RHOSP** 프로젝트는 **PTR** 레코드를 생성하는 영역을 소유해야 합니다.
- 보다 광범위하게 정의된 **PTR** 레코드를 저장하는 역방향 조회 영역입니다. 예를 들어 **198.51.100.0/24** 역방향 조회 영역인 **100.51.198.in-addr-arpa.** 자세한 내용은 **9.2절. "역방향 조회 영역 생성"**의 내용을 참조하십시오.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. **openstack recordset create** 명령을 사용하고 이러한 필수 인수를 지정하여 **PTR** 레코드를 생성합니다.

   --record <domain_name>

   조회의 도메인 이름입니다.

--type PTR

생성 중인 레코드 **PTR** 입니다.

<zone_name>

레코드가 있는 역방향 조회 영역의 이름입니다.

<record_name>

PTR 레코드의 이름입니다.

레코드 이름은 **<zone_name>**과 일치하거나 영역의 멤버여야 합니다. 예를 들어 역방향 조회 영역 **100.51.198.in-addr.arpa**의 경우 유효한 **PTR** 레코드 이름 **1. 100.51.198.in-addr.arpa. , 2.100.51.198.in-addr.arpa., 198.51.100.0/24** 서브넷의 기타 역방향 **IP** 주소입니다.

예제

이 예에서 역방향 조회 영역은 보다 광범위하게 정의됩니다. 예를 들어 **100.51.198.0/24** 역방향 조회 영역인 **100.51.198.in-addr-arpa:**

```
$ openstack recordset create --record cats.example.com. --type PTR \
--ttl 3600 100.51.198.in-addr.arpa. 3.100.51.198.in-addr.arpa.
```

샘플 출력

```
+-------------+--------------------------------------+
| Field       | Value                                |
+-------------+--------------------------------------+
| action      | CREATE                               |
| created_at  | 2022-02-02T20:10:54.000000           |
| description | None                                 |
| id          | c843729b-7aaf-4f99-a40a-d9bf70edf271 |
| name        | 3.100.51.198.in-addr.arpa.           |
| project_id  | 123d51544df443e790b8e95cce52c285     |
| records     | cats.example.com.                    |
| status      | PENDING                              |
| ttl         | 3600                                 |
| type        | PTR                                  |
| updated_at  | None                                 |
| version     | 1                                    |
| zone_id     | e9fd0ced-1d3e-43fa-b9aa-6d4b7a73988d |
| zone_name   | 100.51.198.in-addr.arpa.             |
+-------------+--------------------------------------+
```

검증

1. 역방향 조회를 수행하여 IP 주소(198.51.100.3)가 도메인 이름(cats.example.com)에 매핑되었는지 확인합니다.

   예제

   이 예에서 203.0.113.5 는 배포의 DNS 서버 중 하나입니다.

   ```
   $ dig @203.0.113.5 -x 198.51.100.3 +short
   ```

   샘플 출력

   ```
   cats.example.com.
   ```

2. 역방향 조회를 수행하여 다른 IP 주소(198.51.100.0/24)가 도메인 이름(example.com)에 매핑되었는지 확인합니다.

   예제

   이 예에서 203.0.113.5 는 배포의 DNS 서버 중 하나입니다.

   ```
   $ dig @203.0.113.5 -x 198.51.100.10 +short
   ```

   샘플 출력

   ```
   example.com.
   ```

추가 리소스

- *명령줄 인터페이스 참조*에서 **recordset 생성**
- **dig** 명령 도움말 페이지.

9.5. 부동 IP 주소에 대한 PTR 레코드 설정

RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate)에서 유동 **IP** 주소에 대한 **PTR** 레코드를 생성하여 역방향 조회를 허용할 수 있습니다.

사전 요구 사항

- 하나 이상의 유동 **IP**가 정의되어 있습니다.
- **PTR** 레코드를 생성할 유동 **IP**의 역방향 조회 영역입니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. **PTR** 레코드를 삭제할 유동 **IP** 주소의 **ID**를 확인합니다. 이 정보는 이후 단계에서 필요합니다.

   ```
   $ openstack floating ip list -c ID -c "Floating IP Address"
   ```

   샘플 출력

   ```
   +--------------------------------------+---------------------+
   ```

```
| ID                                   | Floating IP Address |
+--------------------------------------+---------------------+
| 5c02c519-4928-4a38-bd10-c748c200912f | 192.0.2.11          |
| 89532684-13e1-4af3-bd79-f434c9920cc3 | 192.0.2.12          |
| ea3ebc6d-a146-47cd-aaa8-35f06e1e8c3d | 192.0.2.13          |
+--------------------------------------+---------------------+
```

3. **유동 IP를 호스팅하는 neutron 인스턴스의 RHOSP 리전 이름을 확인합니다. 이 정보는 이후 단계에서 필요합니다.**

```
$ openstack endpoint list -c ID -c Region -c "Service Name"
```

샘플 출력

```
+----------------------------------+-----------+--------------+
| ID                               | Region    | Service Name |
+----------------------------------+-----------+--------------+
| 16526452effd467a915155ceccf79dae | RegionOne | placement    |
| 21bf826a62a14456a61bd8f39648e849 | RegionOne | keystone     |
| 9cb1956999c54001a39d11ea14e037a1 | RegionOne | nova         |
| bdeec4e2665d4605bb89e16a8b1bc50d | RegionOne | glance       |
| ced05a1c03ab44caa1a351ace95429e6 | RegionOne | neutron      |
| e79e3113ea544d039b3a6378e60bdf3f | RegionOne | nova         |
| f91ee44123954b6c82162dcd2d4fc965 | RegionOne | designate    |
+----------------------------------+-----------+--------------+
```

4. **openstack ptr record set 명령을 사용하고 필요한 인수를 지정하여 PTR 레코드를 생성합니다.**

<floating_IP_ID>

<region_name>:<floating_IP_ID> 형식의 부동 IP ID입니다.

<ptrd_name>

역방향 조회를 수행할 때 반환해야 하는 도메인 이름 대상입니다.

예제

```
$ openstack ptr record set RegionOne:5c02c519-4928-4a38-bd10-c748c200912f
ftp.example.com.
```

샘플 출력

```
+-------------+-------------------------------------------------+
| Field       | Value                                           |
+-------------+-------------------------------------------------+
| action      | CREATE                                          |
| address     | 192.0.2.11                                      |
| description | None                                            |
| id          | RegionOne:5c02c519-4928-4a38-bd10-c748c200912f |
| ptrdname    | ftp.example.com.                                |
| status      | PENDING                                         |
| ttl         | 3600                                            |
+-------------+-------------------------------------------------+
```

검증

- **역방향 조회를 수행하여 유동 IP 주소(192.0.2.11)가 도메인 이름(ftp.example.com)에 매핑되었는지 확인합니다.**

  예제

  이 예에서 203.0.113.5 는 배포의 DNS 서버 중 하나입니다.

  ```
  $ dig @203.0.113.5 -x 192.0.2.11 +short
  ```

  샘플 출력

  ```
  ftp.example.com.
  ```

추가 리소스

- *명령줄 인터페이스 참조에* 설정된 **PTR** 레코드
- **dig** 명령 도움말 페이지.

9.6. 유동 IP 주소에 대한 PTR 레코드 설정 해제

RHOSP(Red Hat OpenStack Platform) DNS 서비스(**designate**)에서 유동 **IP** 주소와 연결된 **PTR** 레코드를 제거할 수 있습니다.

사전 요구 사항

- 유동 **IP**의 **PTR** 레코드입니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. **PTR** 레코드를 삭제할 유동 **IP** 주소의 **ID**를 확인합니다. 이 정보는 이후 단계에서 필요합니다.

   ```
   $ openstack floating ip list -c ID -c "Floating IP Address"
   ```

   샘플 출력

```
+--------------------------------------+--------------------+
| ID                                   | Floating IP Address |
+--------------------------------------+--------------------+
| 5c02c519-4928-4a38-bd10-c748c200912f | 192.0.2.11         |
| 89532684-13e1-4af3-bd79-f434c9920cc3 | 192.0.2.12         |
| ea3ebc6d-a146-47cd-aaa8-35f06e1e8c3d | 192.0.2.13         |
+--------------------------------------+--------------------+
```

3. **RHOSP** 리전의 이름을 확인합니다. 이 정보는 이후 단계에서 필요합니다.

```
$ openstack endpoint list -c ID -c Region -c "Service Name"
```

샘플 출력

```
+----------------------------------+-----------+--------------+
| ID                               | Region    | Service Name |
+----------------------------------+-----------+--------------+
| 16526452effd467a915155ceccf79dae | RegionOne | placement    |
| 21bf826a62a14456a61bd8f39648e849 | RegionOne | keystone     |
| 9cb1956999c54001a39d11ea14e037a1 | RegionOne | nova         |
| bdeec4e2665d4605bb89e16a8b1bc50d | RegionOne | glance       |
| ced05a1c03ab44caa1a351ace95429e6 | RegionOne | neutron      |
| e79e3113ea544d039b3a6378e60bdf3f | RegionOne | nova         |
| f91ee44123954b6c82162dcd2d4fc965 | RegionOne | designate    |
+----------------------------------+-----------+--------------+
```

4. **openstack ptr record unset** 명령을 사용하고 필요한 인수를 지정하여 **PTR** 레코드를 삭제합니다.

<floating_IP_ID>

<region>:<floating_IP_ID> 형식의 부동 **IP ID**입니다.

예제

```
$ openstack ptr record unset RegionOne:5c02c519-4928-4a38-bd10-c748c200912f
```

검증

- **PTR 레코드를 제거했는지 확인합니다.**

```
$ openstack ptr record list
```

추가 리소스

- *명령줄 인터페이스 참조*에서 **PTR 레코드 설정되지 않음**

10장. DNS 서비스 문제 해결

Red Hat OpenStack Platform DNS 서비스(designate) 로그를 검토하고 몇 가지 간단한 명령을 사용하면 서비스가 올바르게 실행되고 있는지 확인할 수 있습니다. 이러한 작업은 DNS 서비스 문제 해결의 첫 번째 단계입니다.

이 섹션에 포함된 항목은 다음과 같습니다.

- 10.1절. “DNS 서비스 및 BIND 로그”
- 10.2절. “DNS 서비스 풀 구성 내보내기”
- 10.3절. “사용 가능한 DNS 서비스 엔드 포인트 나열”

10.1. DNS 서비스 및 BIND 로그

Red Hat OpenStack Platform DNS 서비스(designate) 로그를 검토하면 문제를 해결할 때 유용할 수 있습니다.

DNS 서비스 로그는 /var/log/containers/designate 에 있습니다. 각 구성 요소 서비스에 대해 하나의 로그가 있습니다.

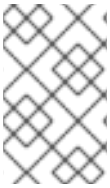
- central.log
- designate-api.log
- mdns.log
- producer.log
- worker.log

Red Hat이 RHOSP DNS 서비스와 함께 제공하는 BIND 서버의 로그는 /var/log/containers/designate/designate-bind:에 있습니다.

- central.log
- designate-api.log

10.2. DNS 서비스 풀 구성 내보내기

DNS 풀 구성 사본을 사용하여 RHOSP(Red Hat OpenStack Platform) DNS 서비스(designate) 문제를 해결할 수 있습니다.



참고

RHOSP 17.1에서는 여러 풀이 지원되지 않습니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

예제

```
$ source ~/overcloudrc
```

2. designate-central 컨테이너에서 designate-manage pool 명령을 실행합니다. 먼저 컨트롤 플레인 VM 인스턴스의 IP 주소를 가져옵니다.

예제

이 예에서 컨트롤러 이름은 overcloud-controller-0 입니다.

```
$ CTRL_IP=$(openstack server list -f value -c Networks \
--name overcloud-controller-0 | sed 's/ctlplane=//')
```

3. **컨트롤러 노드 중 하나에 로그인하고 현재 실행 중인 DNS 서비스 풀 구성을 콘솔에 표시합니다.**

```
$ ssh tripleo-admin@${CTRL_IP} sudo podman exec -i -u root \
designate-central designate-manage pool show_config
```

샘플 출력

```
Pool Configuration:
--------------------
also_notifies: []
attributes: {}
description: Default Pool
id: 794ccc2c-d751-44fe-b57f-8894c9f5c842
name: default
nameservers:
- host: 192.0.2.111
  port: 53
- host: 192.0.2.109
  port: 53
- host: 192.0.2.131
  port: 53
ns_records:
- hostname: ns2.example.com.
  priority: 2
- hostname: ns1.example.com.
  priority: 1
- hostname: ns3.example.com.
  priority: 3
targets:
- description: BIND9 Server 3
  masters:
  - host: 192.0.2.137
    port: 16002
  - host: 192.0.2.137
    port: 16001
  - host: 192.0.2.137
    port: 16000
  options:
    host: 192.0.2.111
    port: '53'
    rndc_config_file: /etc/designate/private/bind3.conf
    rndc_host: 192.0.2.111
    rndc_port: '953'
  type: bind9
- description: BIND9 Server 2
  masters:
```

```
    - host: 192.0.2.137
      port: 16001
    - host: 192.0.2.137
      port: 16002
    - host: 192.0.2.137
      port: 16000
    options:
      host: 192.0.2.131
      port: '53'
      rndc_config_file: /etc/designate/private/bind2.conf
      rndc_host: 192.0.2.131
      rndc_port: '953'
    type: bind9
  - description: BIND9 Server 1
    masters:
    - host: 192.0.2.137
      port: 16002
    - host: 192.0.2.137
      port: 16001
    - host: 192.0.2.137
      port: 16000
    options:
      host: 192.0.2.109
      port: '53'
      rndc_config_file: /etc/designate/private/bind1.conf
      rndc_host: 192.0.2.109
      rndc_port: '953'
    type: bind9
```

4. 현재 풀 구성을 파일로 내보내려면 **designate-manage pool generate_file** 명령을 사용합니다.

예제

```
$ sudo podman exec -i designate-manage pool generate_file \
--file ~/my_dns_service_config.yaml
```

작은 정보

podman cp 명령을 사용하여 컨테이너에서 로컬 시스템으로 파일을 복사합니다.

10.3. 사용 가능한 DNS 서비스 엔드 포인트 나열

Red Hat OpenStack Platform DNS 서비스(designate) 끝점 및 해당 상태를 결정하는 것은 문제를 해결할 때 유용할 수 있습니다.

프로세스

1. 자격 증명 파일을 가져옵니다.

   예제

   ```
   $ source ~/overcloudrc
   ```

2. RHOSP 서비스 끝점을 나열합니다.

   ```
   $ openstack endpoint list -c "Service Name" -c Enabled -c URL
   ```

   샘플 출력

   ```
   +--------------+---------+-----------------------------------------------+
   | Service Name | Enabled | URL                                           |
   +--------------+---------+-----------------------------------------------+
   | swift        | True    | http://198.51.100.61:8080                     |
   | designate    | True    | http://203.0.113.103:9001                     |
   | heat-cfn     | True    | http://192.0.2.137:8000/v1                    |
   | designate    | True    | http://192.0.2.137:9001                       |
   | placement    | True    | http://203.0.113.103:8778/placement           |
   | cinderv3     | True    | http://203.0.113.103:8776/v3/%(tenant_id)s    |
   | heat         | True    | http://203.0.113.103:8004/v1/%(tenant_id)s    |
   | heat-cfn     | True    | http://203.0.113.103:8000/v1                  |
   | nova         | True    | http://203.0.113.103:8774/v2.1                |
   | heat         | True    | http://192.0.2.137:8004/v1/%(tenant_id)s      |
   | glance       | True    | http://203.0.113.103:9292                     |
   | heat         | True    | http://203.0.113.103:8004/v1/%(tenant_id)s    |
   | glance       | True    | http://203.0.113.103:9292                     |
   | neutron      | True    | http://203.0.113.103:9696                     |
   | nova         | True    | http://192.0.2.137:8774/v2.1                  |
   | cinderv3     | True    | http://192.0.2.137:8776/v3/%(tenant_id)s      |
   | placement    | True    | http://203.0.113.103:8778/placement           |
   ```

```
| keystone     | True    | http://192.168.24.17:35357                    |
| neutron      | True    | http://192.0.2.137:9696                       |
| nova         | True    | http://203.0.113.103:8774/v2.1                |
| heat-cfn     | True    | http://203.0.113.103:8000/v1                  |
| cinderv3     | True    | http://203.0.113.103:8776/v3/%(tenant_id)s    |
| glance       | True    | http://192.0.2.137:9292                       |
| placement    | True    | http://192.0.2.137:8778/placement             |
| swift        | True    | http://198.51.100.61:8080/v1/AUTH_%(tenant_id)s |
| swift        | True    | http://192.0.2.137:8080/v1/AUTH_%(tenant_id)s |
| designate    | True    | http://203.0.113.103:9001                     |
| keystone     | True    | http://192.0.2.137:5000                       |
| neutron      | True    | http://203.0.113.103:9696                     |
| keystone     | True    | http://203.0.113.103:5000                     |
+--------------+---------+-----------------------------------------------+
```