Como configurar o BIND como um servidor DNS de rede privada no Ubuntu 18.04

Published on January 9, 2020

DNS Networking Ubuntu 18.04

By Mitchell Anicas and Justin Ellingwood

Português

Introdução

Uma parte importante do gerenciamento da configuração e infraestrutura de servidores inclui a manutenção de maneira fácil de verificar as interfaces de rede e endereços IP por nome, através da configuração de um Sistema de Nome de Domínio (DNS). Ao usar os nomes de domínio totalmente qualificados (FQDNs), ao invés de endereços IP para especificar os endereços de rede, facilita-se a configuração de serviços e aplicativos e aumenta-se a capacidade de manutenção dos arquivos de configuração. Configurar seu próprio DNS para sua rede privada é uma ótima maneira de melhorar o gerenciamento dos seus servidores.

Neste tutorial, veremos como configurar um servidor DNS interno usando o software de servidor de nomes BIND (BIND9) no Ubuntu 18.04, que pode ser usado pelos seus servidores para resolver nomes de host e endereços IP privados. Isso fornece uma maneira central de gerenciamento dos seus nomes de host e endereços IP privados internos, o que é indispensável quando seu ambiente se expande para mais de alguns poucos hosts.

A versão CentOS deste tutorial pode ser encontrada aqui.

Pré-requisitos

Para completar este tutorial, você precisará das seguinte infraestrutura. Crie cada servidor **no mesmo datacenter** com o modo de **rede privada habilitado**:

- Um servidor Ubuntu 18.04 para servir como o servidor DNS primário, o **ns1**
- (Recomendado) Um segundo servidor Ubuntu 18.04 para servir como um servidor DNS secundário, o **ns2**
- Servidores adicionais no mesmo datacenter que usarão seus servidores DNS

Em cada um desses servidores, configure o acesso administrativo por um usuário `sudo` e um firewall seguindo nosso guia de configuração inicial do servidor Ubuntu 18.04.

Se você não estiver familiarizado com os conceitos do DNS, é recomendável que você leia pelo menos as três primeiras partes da nossa Introdução ao gerenciamento do DNS.

Exemplo de infraestrutura e objetivos

Para os fins deste artigo, vamos assumir o seguinte:

- Temos dois servidores que serão designados como nossos servidores de nome DNS. Vamos nos referir a eles como **ns1** e **ns2** neste guia.
- Temos dois servidores de cliente adicionais que irão usar a infraestrutura DNS que criamos. Vamos chamá-los **host1** e **host2** neste guia. Você pode adicionar quantos quiser para sua infraestrutura.
- Todos esses servidores existem no mesmo datacenter. Vamos assumir que este datacenter chama-se **nyc3**.
- Todos esses servidores têm o modo de rede privada habilitado (e estão na sub-rede `10.128.0.0/16`. É provável que você tenha que ajustar isso para seus servidores).
- Todos os servidores estão conectados a um projeto executado em "example.com". Como nosso sistema DNS será totalmente interno e privado, você não precisa comprar um nome de domínio. No entanto, usar um domínio que você possui pode ajudar a evitar conflitos com domínios de encaminhamento público.

Com essas suposições, decidimos que é sensato usar um esquema de nomeação que usa "nyc3.example.com" para se referir à nossa sub-rede ou zona privada. Portanto, o Nome de domínio totalmente qualificado (FQDN) privado do **host1** será **host1.nyc3.example.com**. Consulte a tabela a seguir com os detalhes relevantes:

Host	Função	FQDN privado	Endereço IP privado
ns1	Servidor DNS primário	ns1.nyc3.example.com	10.128.10.11
ns2	Servidor DNS secundário	n2.nyc3.example.com	10.128.20.12
host1	Host genérico 1	host1.nyc3.example.com	10.128.100.101
host2	Host genérico 2	host2.nyc3.example.com	10.128.200.102

> **Nota:** A sua configuração existente será diferente, mas os nomes dos exemplos e endereços IP serão usados para demonstrar como configurar um servidor DNS para fornecer um DNS interno funcional. Você consegue adaptar essa configuração ao seu ambiente com facilidade, pela substituição dos nomes de host e endereços IP privados pelos seus. Não é necessário usar o nome regional do datacenter no seu esquema de nomeação, mas usamos ele aqui para denotar que esses hosts pertencem a uma rede privada de um datacenter particular. Se você usar vários datacenters, é possível configurar um DNS interno dentro de cada datacenter respectivo.

Ao final deste tutorial, teremos um servidor DNS primário, **ns1**, e opcionalmente um servidor DNS secundário, **ns2**, que servirá como backup.

Vamos começar pela instalação do nosso servidor DNS primário, o ns1.

Como instalar o BIND nos servidores DNS

> **Nota:** As passagens que estiverem destacadas em vermelho são importantes! Normalmente, elas serão usadas para denotar algo que precisa ser substituído pelas suas próprias configurações ou que deve ser modificado ou adicionado a um arquivo de configuração. Por exemplo, se você ver algo como `host1.nyc3.example.com`, substitua-o pelo FQDN do seu próprio servidor. De forma similar, se você ver `host1_private_IP`, substitua-o pelo endereço IP privado do seu próprio servidor.

Em ambos os servidores DNS, **ns1** e **ns2**, atualize o cache de pacotes `apt` digitando:

```
ns$ sudo apt-get update
```

Copy

Agora, instale o BIND:

```
ns$ sudo apt-get install bind9 bind9utils bind9-doc
```

Como configurar o Bind para o modo IPv4

Antes de continuar, vamos colocar o BIND no modo IPv4, já que nossa rede privada usa exclusivamente o IPv4. Nos dois servidores, edite o arquivo de configuração padrão `bind9` digitando:

```
ns$ sudo nano /etc/default/bind9
```

Copy

Adicione "-4" ao final do parâmetro `OPTIONS`. Ele deve se parecer com o seguinte:

/etc/default/bind9

```
. . .
OPTIONS="-u bind -4"
```

Salve e feche o arquivo quando você terminar.

Reinicie o BIND para implementar as alterações:

```
ns$ sudo systemctl restart bind9
```

Copy

Agora que o BIND está instalado, vamos configurar o servidor DNS primário.

Como configurar o servidor DNS primário

A configuração do BIND consiste em vários arquivos, que estão incluídos no arquivo de configuração principal, o `named.conf`. Estes nomes de arquivos começam com `named` porque este é o nome do processo que o BIND executa (abreviação de "domain name daemon"). Vamos começar configurando o arquivo de opções.

Como configurar o arquivo de opções

No **ns1**, abra o arquivo `named.conf.options` para edição:

```
ns1$ sudo nano /etc/bind/named.conf.options
```

Copy

Acima do bloco `options` existente, crie um bloco ALC (lista de controle de acesso) *new* chamado "confiáveis". É aqui que vamos definir uma lista de clientes para os quais consultas recursivas DNS serão permitidas (ou seja, seus servidores que estão no mesmo datacenter que o **ns1**). Usando nosso exemplo de endereço IP privado, serão adicionados o **ns1**, **ns2**, **host1** e **host2** à nossa lista de clientes confiáveis:

/etc/bind/named.conf.options — 1 of 3

```
acl "trusted" {
        10.128.10.11;    # ns1 - can be set to localhost
        10.128.20.12;    # ns2
        10.128.100.101;  # host1
        10.128.200.102;  # host2
};

options {

        . . .
```

Agora que temos nossa lista de clientes DNS confiáveis, queremos editar o bloco `options`. Atualmente, o início do bloco se parece com o seguinte:

/etc/bind/named.conf.options — 2 of 3

```
        . . .
};

options {
        directory "/var/cache/bind";
        . . .
}
```

Abaixo da diretriz `directory`, adicione as linhas de configuração destacadas (e substitua no endereço IP do **ns1** apropriado) para que fique dessa forma:

/etc/bind/named.conf.options — 3 of 3

```
        . . .

};

options {
        directory "/var/cache/bind";

        recursion yes;                 # enables resursive queries
        allow-recursion { trusted; };  # allows recursive queries from "trusted" clients
        listen-on { 10.128.10.11; };   # ns1 private IP address - listen on private network o
        allow-transfer { none; };      # disable zone transfers by default

        forwarders {
                8.8.8.8;
                8.8.4.4;
        };

        . . .
};
```

Quando você terminar, salve e feche o arquivo `named.conf.options`. A configuração acima especifica que apenas seus próprios servidores (os "confiáveis") poderão consultar seu servidor DNS para domínios externos.

Em seguida, vamos configurar o arquivo local para especificar nossas zonas de DNS.

Como configurar o arquivo local

No **ns1**, abra o arquivo `named.conf.local` para edição:

```
ns1$ sudo nano /etc/bind/named.conf.local
```

Copy

Com exceção de alguns comentários, o arquivo deve estar vazio. Aqui, vamos especificar nossa zona de encaminhamento e nossa zona inversa. As **zonas de DNS** designam um escopo específico para o gerenciamento e definição dos registros de DNS. Como todos nossos domínios estarão dentro do sub-domínio "nyc3.example.com", usaremos ele como nossa zona de encaminhamento. Como os endereços IP privados dos nossos servidores estão no espaço de IP `10.128.0.0/16`, uma zona inversa será configurada para que possamos definir pesquisas inversas dentro desse intervalo.

Adicione a zona de encaminhamento com as linhas a seguir, substituindo o nome da zona pelo seu próprio e o **endereço IP privado do servidor DNS secundário** na diretriz `allow-transfer`:

/etc/bind/named.conf.local — 1 of 2

```
zone "nyc3.example.com" {
    type master;
    file "/etc/bind/zones/db.nyc3.example.com"; # zone file path
    allow-transfer { 10.128.20.12; };           # ns2 private IP address - secondary
};
```

Supondo que nossa sub-rede privada seja `10.128.0.0/16`, adicione a zona reversa com as linhas a seguir (**note que nosso nome da zona reversa inicia com "128.10", que é a reversão do octeto reverso de "10.128"**):

/etc/bind/named.conf.local — 2 of 2

```
    . . .
};

zone "128.10.in-addr.arpa" {
    type master;
    file "/etc/bind/zones/db.10.128";  # 10.128.0.0/16 subnet
```

```
    allow-transfer { 10.128.20.12; };  # ns2 private IP address - secondary
};
```

Se seus servidores se estendem por várias sub-redes privadas mas estão no mesmo datacenter, certifique-se de especificar uma zona adicional e um arquivo de zona para cada sub-rede distinta. Quando terminar de adicionar todas as suas zonas desejadas, salve e saia do arquivo `named.conf.local`.

Agora que nossas zonas estão especificadas em BIND, precisamos criar os arquivos correspondentes da zona de encaminhamento e da zona reversa.

Como criar o arquivo da zona de encaminhamento

O arquivo da zona de encaminhamento está onde definimos os registros DNS para pesquisas de encaminhamentos de DNS. Isso é, quando o DNS receber um nome de consulta, "host1.nyc3.example.com", por exemplo, ele olhará no arquivo da zona de encaminhamento para resolver o endereço IP privado correspondente do **host1.**

Vamos criar o diretório onde nossos arquivos de zona irão permanecer. De acordo com nossa configuração **named.conf.local**, esse local deve ser o `/etc/bind/zones`:

```
ns1$ sudo mkdir /etc/bind/zones
```
Copy

Vamos basear nosso arquivo da zona de encaminhamento no arquivo de zona amostral `db.local`. Copie-o para o local correto com os seguintes comandos:

```
ns1$ sudo cp /etc/bind/db.local /etc/bind/zones/db.nyc3.example.com
```
Copy

Agora, vamos editar nosso arquivo da zona de encaminhamento:

```
ns1$ sudo nano /etc/bind/zones/db.nyc3.example.com
```
Copy

Inicialmente, ele se parecerá com o seguinte:

/etc/bind/zones/db.nyc3.example.com — original

```
$TTL    604800
@       IN      SOA     localhost. root.localhost. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      localhost.      ; delete this line
@       IN      A       127.0.0.1       ; delete this line
@       IN      AAAA    ::1             ; delete this line
```

Primeiro, vamos editar o registro do SOA. Substitua o primeiro "localhost" pelo FQDN do **ns1** e então substitua "root.localhost" por "admin.nyc3.example.com". Toda vez que você editar um arquivo de zona, será necessário aumentar o valor **serial** antes de reiniciar o processo `named`. Vamos incrementá-lo para "3". Agora, ele deve se parecer com isso:

/etc/bind/zones/db.nyc3.example.com — updated 1 of 3

```
@       IN      SOA     ns1.nyc3.example.com. admin.nyc3.example.com. (
                              3         ; Serial

                              . . .
```

Em seguida, delete os três registros ao final do arquivo (depois do registro do SOA). Se não tiver certeza sobre quais linhas excluir, elas estão marcadas acima com um comentário "delete this line".

Ao final do arquivo, adicione os registros do servidor do seu nome com as linhas a seguir (substitua os nomes pelos seus próprios). Note que a segunda coluna especifica que esses registros são "NS":

/etc/bind/zones/db.nyc3.example.com — updated 2 of 3

```
. . .

; name servers - NS records
    IN      NS      ns1.nyc3.example.com.
    IN      NS      ns2.nyc3.example.com.
```

Agora, adicione os registros A para seus hosts que pertencem a esta zona. Isso inclui qualquer servidor cujo nome queremos que termine com ".nyc3.example.com" (substitua os nomes e endereços IP privados). Usando nossos nomes de exemplo e endereços IP privados, vamos adicionar registros A para o **ns1**, **ns2**, **host1** e **host2** desta forma:

/etc/bind/zones/db.nyc3.example.com — updated 3 of 3

```
. . .

; name servers - A records
ns1.nyc3.example.com.          IN      A       10.128.10.11
ns2.nyc3.example.com.          IN      A       10.128.20.12

; 10.128.0.0/16 - A records
host1.nyc3.example.com.        IN      A      10.128.100.101
host2.nyc3.example.com.        IN      A      10.128.200.102
```

Salve e feche o arquivo `db.nyc3.example.com`.

Nosso arquivo de exemplo final da zona de encaminhamento se parece com o seguinte:

/etc/bind/zones/db.nyc3.example.com — updated

```
$TTL    604800
@       IN      SOA     ns1.nyc3.example.com. admin.nyc3.example.com. (
                      3     ; Serial
                 604800     ; Refresh
                  86400     ; Retry
                2419200     ; Expire
                 604800 )   ; Negative Cache TTL
;
; name servers - NS records
     IN      NS      ns1.nyc3.example.com.
     IN      NS      ns2.nyc3.example.com.

; name servers - A records
ns1.nyc3.example.com.          IN      A       10.128.10.11
ns2.nyc3.example.com.          IN      A       10.128.20.12

; 10.128.0.0/16 - A records
host1.nyc3.example.com.        IN      A      10.128.100.101
host2.nyc3.example.com.        IN      A      10.128.200.102
```

Agora, vamos seguir para o(s) arquivo(s) da zona reversa.

Como criar o(s) arquivo(s) da zona reversa

Os arquivos da zona reverso estão onde definimos os registros DNS PTR para pesquisas de DNS reverso. Isso é, quando o DNS recebe uma consulta pelo endereço IP, "10.128.100.101", por exemplo, ele olhará no(s) arquivo(s) da zona reversa para resolver o FQDN correspondente, sendo ele, o "host1.nyc3.example.com" neste caso.

No **ns1**, para cada zona reversa especificada no arquivo `named.conf.local`, crie um arquivo de zona reversa. Vamos basear nosso(s) arquivo(s) de zona reversa no arquivo de zona amostral `db.127`. Copie-o para o local correto com os seguintes comandos (subtituindo o nome do arquivo de destino para que ele corresponda à definição da sua zona reversa):

```
ns1$ sudo cp /etc/bind/db.127 /etc/bind/zones/db.10.128
```

Copy

Edite o arquivo de zona reversa que corresponde à(s) zona(s) reversa(s) definida(s) em `named.conf.local`:

```
ns1$ sudo nano /etc/bind/zones/db.10.128
```

Copy

Inicialmente, ele se parecerá com o seguinte:

/etc/bind/zones/db.10.128 — original

```
$TTL    604800
@       IN      SOA     localhost. root.localhost. (
                              1         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      localhost.      ; delete this line
1.0.0   IN      PTR     localhost.      ; delete this line
```

De maneira similar ao arquivo de zona de encaminhamento, edite o registro do SOA e aumente o valor **serial**. Ela deve se parecer com isto:

/etc/bind/zones/db.10.128 — updated 1 of 3

```
@       IN      SOA     ns1.nyc3.example.com. admin.nyc3.example.com. (
                              3         ; Serial

                              . . .
```

Agora, delete os dois registros ao final do arquivo (depois do registro do SOA). Se não tiver certeza sobre quais linhas excluir, elas estão marcadas acima com um comentário “delete this line”.

Ao final do arquivo, adicione os registros do servidor do seu nome com as linhas a seguir (substitua os nomes pelos seus próprios). Note que a segunda coluna especifica que esses registros são “NS”:

/etc/bind/zones/db.10.128 — updated 2 of 3

```
. . .

; name servers - NS records
      IN      NS      ns1.nyc3.example.com.
      IN      NS      ns2.nyc3.example.com.
```

Então, adicione os registros `PTR` para todos os seus servidores cujos endereços IP estão na sub-rede do arquivo de zona que está editando. No nosso exemplo, isso inclui todos os nossos hosts porque eles estão todos na sub-rede `10.128.0.0/16`. Note que a primeira coluna consiste nos dois últimos octetos dos endereços IP privados dos seus servidores em **reversed order**. Certifique-se de substituir os nomes e endereços IP privados para corresponder aos seus servidores:

/etc/bind/zones/db.10.128 — updated 3 of 3

```
. . .

; PTR Records
11.10   IN      PTR     ns1.nyc3.example.com.    ; 10.128.10.11
12.20   IN      PTR     ns2.nyc3.example.com.    ; 10.128.20.12
101.100 IN      PTR     host1.nyc3.example.com.  ; 10.128.100.101
102.200 IN      PTR     host2.nyc3.example.com.  ; 10.128.200.102
```

Salve e feche o arquivo de zona reversa (repita essa seção caso precise adicionar mais arquivos de zona reversa).

Nosso arquivo de exemplo final de zona reversa se parece com o seguinte:

/etc/bind/zones/db.10.128 — updated

```
$TTL    604800
@       IN      SOA     nyc3.example.com. admin.nyc3.example.com. (
                              3         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
```

```
                        604800 )       ; Negative Cache TTL
; name servers
      IN      NS      ns1.nyc3.example.com.
      IN      NS      ns2.nyc3.example.com.

; PTR Records
11.10    IN      PTR     ns1.nyc3.example.com.    ; 10.128.10.11
12.20    IN      PTR     ns2.nyc3.example.com.    ; 10.128.20.12
101.100  IN      PTR     host1.nyc3.example.com.  ; 10.128.100.101
102.200  IN      PTR     host2.nyc3.example.com.  ; 10.128.200.102
```

Agora que terminamos de editar nossos arquivos, podemos verificá-los à procura de erros.

Verificando a sintaxe de configuração do BIND

Execute o comando a seguir para verificar a sintaxe dos arquivos `named.conf*`:

```
ns1$ sudo named-checkconf
```
Copy

Se seus arquivos de configuração nomeados não tiverem erros de sintaxe, você retornará ao seu prompt do shell e não verá nenhuma mensagem de erro. Se houver problemas com seus arquivos de configuração, reveja a mensagem de erro e a seção "Como configurar o servidor DNS primário", e então tente o `named-checkconf` novamente.

O comando `named-checkzone` pode ser usado para verificar a correção dos arquivos da sua zona. Seu primeiro argumento especifica um nome de zona e o segundo especifica o arquivo da zona correspondente, sendo que ambos estão definidos em `named.conf.local`.

Por exemplo, para verificar a configuração da zona de encaminhamento "nyc3.example.com", execute o seguinte comando (mude os nomes para que correspondam à sua zona de encaminhamento e arquivo):

```
$ sudo named-checkzone nyc3.example.com db.nyc3.example.com
```
Copy

E para verificar a configuração da zona reversa "128.10.in-addr.arpa", execute o seguinte comando (mude os números para que correspondam à sua zona reversa e arquivo):

```
$ sudo named-checkzone 128.10.in-addr.arpa /etc/bind/zones/db.10.128
```
Copy

Quando todos os arquivos de configuração e zona estiverem livres de erros, você está pronto para reiniciar o serviço BIND.

Reiniciando o BIND

Reinicie o BIND:

```
ns1$ sudo systemctl restart bind9
```
Copy

Se você tiver o firewall UFW configurado, libere o acesso para o BIND digitando:

```
ns1$ sudo ufw allow Bind9
```
Copy

Seu servidor de DNS primário agora está configurado e pronto para responder às consultas do DNS. Vamos seguir em frente para a criação do servidor DNS secundário.

Configurando o servidor DNS secundário

Na maioria dos ambientes, é uma boa ideia configurar um servidor DNS secundário que responda aos pedidos caso o primário fique indisponível. Felizmente, o servidor DNS secundário é muito mais fácil de configurar.

No **ns2**, edite o arquivo `named.conf.options`:

```
ns2$ sudo nano /etc/bind/named.conf.options
```
Copy

Ao topo do arquivo, adicione o ACL com os endereços IP privados de todos os seus servidores confiáveis:

/etc/bind/named.conf.options — updated 1 of 2 (secondary)

```
acl "trusted" {
        10.128.10.11;   # ns1
        10.128.20.12;   # ns2 - can be set to localhost
        10.128.100.101;  # host1
        10.128.200.102;  # host2
};

options {

        . . .
```

Abaixo da diretriz `directory`, adicione as seguintes linhas:

/etc/bind/named.conf.options — updated 2 of 2 (secondary)

```
        recursion yes;
        allow-recursion { trusted; };
        listen-on { 10.128.20.12; };      # ns2 private IP address
        allow-transfer { none; };          # disable zone transfers by default

        forwarders {
                8.8.8.8;
                8.8.4.4;
        };
```

Salve e feche o arquivo `named.conf.options`. Este arquivo deve se parecer exatamente com o arquivo `named.conf.options` do **ns1**, exceto por precisar ser configurado para escutar o endereço IP privado do **ns2**.

Agora, edite o arquivo `named.conf.local`:

```
ns2$ sudo nano /etc/bind/named.conf.local
```

Copy

Definas as zonas subordinadas que correspondam às zonas mestras no servidor DNS primário. Note que como o tipo é “subordinado”, o arquivo não contém um caminho e há uma diretriz `masters` que deve ser configurada para o endereço IP privado do servidor DNS primário. Se você definiu várias zonas inversas no servidor DNS primário, certifique-se de adicionar todas elas aqui:

/etc/bind/named.conf.local — updated (secondary)

```
zone "nyc3.example.com" {
    type slave;
    file "db.nyc3.example.com";
    masters { 10.128.10.11; };  # ns1 private IP
};

zone "128.10.in-addr.arpa" {
    type slave;
    file "db.10.128";
    masters { 10.128.10.11; };  # ns1 private IP
};
```

Agora salve e feche o arquivo `named.conf.local`.

Execute o comando a seguir para verificar a validade dos seus arquivos de configuração:

```
ns2$ sudo named-checkconf
```

Copy

Assim que for aprovado, reinicie o BIND:

```
ns2$ sudo systemctl restart bind9
```

Copy

Permita conexões DNS ao servidor pela alteração das regras do firewall UFW:

```
ns2$ sudo ufw allow Bind9
```
Copy

Agora você tem servidores DNS primários e secundários para resoluções de nome e endereço IP da rede privada. Agora, você precisa configurar os seus servidores de cliente para usar os seus servidores DNS privados.

Configurando os clientes DNS

Antes que todos os seus servidores ACL "confiáveis" possam consultar seus servidores DNS, você precisa configurar cada um deles para usar o **ns1** e o **ns2** como servidores de nomes. Este processo varia dependendo do SO, mas para a maioria das distribuições do Linux, envolve a adição dos seus servidores de nomes ao arquivo `/etc/resolv.conf`.

Clientes Ubuntu 18.04

No Ubuntu 18.04, a rede é configurada com o Netplan, uma abstração que permite que você escreva configurações padronizadas de rede e aplique-as para softwares backend de rede incompatíveis. Para configurar o DNS, precisamos escrever um arquivo de configuração do Netplan.

Primeiramente, encontre o dispositivo associado à sua rede privada consultando a sub-rede privada com o comando `ip address`:

```
$ ip address show to 10.128.0.0/16
```
Copy

```
Output
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qle
    inet 10.128.100.101/16 brd 10.128.255.255 scope global eth1
           valid_lft forever preferred_lft forever
```

Neste exemplo, a interface privada é a `eth1`.

Em seguida, crie um novo arquivo em `/etc/netplan` chamado `00-private-nameservers.yaml`:

```
$ sudo nano /etc/netplan/00-private-nameservers.yaml
```
Copy

Cole lá dentro o seguinte conteúdo. Será necessário modificar a interface da rede privada, os endereços dos seus servidores DNS **ns1** e **ns2** e da zona do DNS:

> **Nota:** o Netplan usa o formato de serialização de dados YAML para seus arquivos de configuração. Como o YAML usa recuos e espaços em branco para definir sua estrutura de dados, certifique-se de que sua definição utilize recuos consistentes para evitar erros.

/etc/netplan 00-private-nameservers.yaml

```
network:
    version: 2
    ethernets:
        eth1:                                                          # Private netv
            nameservers:
                addresses:
                - 10.128.10.11                          # Private IP for ns1
                - 10.132.20.12                          # Private IP for ns2
                search: [ nyc3.example.com ]    # DNS zone
```

Salve e feche o arquivo quando você terminar.

Em seguida, faça o Netplan tentar usar o novo arquivo de configuração utilizando o `netplan try`. Se houver problemas que causem uma perda de rede, o Netplan irá retroceder automaticamente as mudanças após um tempo limite:

```
$ sudo netplan try
```

Copy

```
Output
Warning: Stopping systemd-networkd.service, but it can still be activated by:
  systemd-networkd.socket
Do you want to keep these settings?


Press ENTER before the timeout to accept the new configuration


Changes will revert in 120 seconds
```

Se a contagem regressiva estiver atualizando corretamente ao fim, a nova configuração é, ao menos, funcional o suficiente para não interromper sua conexão via protocolo SSH. Pressione **ENTER** para aceitar a nova configuração.

Agora, verifique o resolvedor DNS do sistema para determinar se sua configuração de DNS foi aplicada:

```
$ sudo systemd-resolve --status
```
Copy

Role para baixo até ver a seção da sua interface de rede privada. Você deve ver os endereços IP privados dos seus servidores DNS listados primeiro, seguidos de alguns valores de retorno. Seu domínio deve estar no "DNS Domain":

```
Output
. . .
Link 3 (eth1)
      Current Scopes: DNS
       LLMNR setting: yes
MulticastDNS setting: no
      DNSSEC setting: no
    DNSSEC supported: no
         DNS Servers: 10.128.10.11
                      10.128.20.12
                      67.207.67.2
                      67.207.67.3
          DNS Domain: nyc3.example.com
. . .
```

Seu cliente agora deve estar configurado para usar seus servidores DNS internos.

Clientes Ubuntu 16.04 e Debian

Nos servidores Linux Ubuntu 16.04 e Debian, você pode editar o arquivo `/etc/network/interfaces`:

```
$ sudo nano /etc/network/interfaces
```
Copy

Encontre lá dentro a linha `dns-nameservers` e anexe no início os seus próprios servidores de nomes na frente da lista que atualmente está lá. Abaixo dessa linha, adicione uma opção `dns-search` apontada para o domínio base da sua infraestrutura. No nosso caso, seria "nyc3.example.com":

/etc/network/interfaces

```
. . .

dns-nameservers 10.128.10.11  10.128.20.12 8.8.8.8
dns-search nyc3.example.com

. . .
```

Salve e feche o arquivo quando você terminar.

Agora, reinicie seus serviços de rede, aplicando as novas mudanças com os comandos a seguir. Certifique-se de substituir o `eth0` pelo nome da sua interface de rede:

```
$ sudo ifdown --force eth0 && sudo ip addr flush dev eth0 && sudo ifup --force eth0
```
Copy

Isso deve reiniciar sua rede sem interromper sua conexão atual. Se funcionou corretamente, você verá algo similar a isto:

```
Output
RTNETLINK answers: No such process
Waiting for DAD... Done
```

Verifique novamente se suas configurações foram aplicadas digitando:

```
$ cat /etc/resolv.conf
```
Copy

Você deve ver seus servidores de nomes no arquivo `/etc/resolv.conf`, além do seu domínio de busca:

```
Output
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 10.128.10.11
nameserver 10.128.20.12
nameserver 8.8.8.8
search nyc3.example.com
```

Seu cliente agora está configurado para usar seus servidores DNS.

Clientes CentOS

No CentOS, RedHat, e Fedora Linux, edite o arquivo `/etc/sysconfig/network-scripts/ifcfg-eth0`. Pode ser que você precise substituir o `eth0` pelo nome da sua interface de rede primária:

```
$ sudo nano /etc/sysconfig/network-scripts/ifcfg-eth0
```
Copy

Procure as opções `DNS1` e `DNS2` e defina-as para os endereços IP privados dos seus servidores de nomes primários e secundários. Adicione um parâmetro `DOMAIN` junto com o domínio base da sua infraestrutura. Neste guia, seria "nyc3.example.com":

/etc/sysconfig/network-scripts/ifcfg-eth0

```
. . .
DNS1=10.128.10.11
DNS2=10.128.20.12
DOMAIN='nyc3.example.com'
. . .
```

Salve e feche o arquivo quando você terminar.

Agora, reinicie o serviço de rede digitando:

```
$ sudo systemctl restart network
```
Copy

O comando pode ficar suspenso por alguns segundos, mas deve retornar você para o prompt em breve.

Verifique se suas alterações foram aplicadas digitando:

```
$ cat /etc/resolv.conf
```
Copy

Você deve ver seus servidores de nomes e domínio de busca na lista:

/etc/resolv.conf

```
nameserver 10.128.10.11
nameserver 10.128.20.12
search nyc3.example.com
```

Seu cliente agora deve conseguir se conectar aos seus servidores DNS e utilizá-los.

Testando os clientes

Use o `nslookup` para testar se seus clientes podem consultar seus servidores de nomes. Você deve conseguir fazer isso em todos os clientes que configurou e que estão no ACL "confiáveis".

Para clientes CentOS, pode ser necessário instalar o utilitário com:

```
$ sudo yum install bind-utils
```
Copy

Podemos começar executando uma pesquisa direta.

Pesquisa direta

Por exemplo, é possível executar uma pesquisa direta para recuperar o endereço IP do **host1.nyc3.example.com** executando o seguinte comando:

```
$ nslookup host1
```
Copy

A consulta do "host1" expande-se para o "host1.nyc3.example.com" pelo fato da opção `search` estar configurada para o seu sub-domínio privado e as consultas de DNS tentarão procurar naquele sub-domínio antes de procurar o host em outro lugar. O resultado do comando acima se pareceria com o seguinte:

```
Output
Server:     127.0.0.53
Address:    127.0.0.53#53

Non-authoritative answer:
Name:   host1.nyc3.example.com
Address: 10.128.100.101
```

Em seguida, podemos verificar as pesquisas inversas.

Pesquisa inversa

Para testar a pesquisa inversa, consulte o servidor DNS com o endereço IP privado do **host1**:

```
$ nslookup 10.128.100.101
```
Copy

Deverá ver um resultado que se parece com o seguinte:

```
Output
11.10.128.10.in-addr.arpa   name = host1.nyc3.example.com.

Authoritative answers can be found from:
```

Se todos os nomes e endereços IP resolverem os valores corretos, seus arquivos de zona estão configurados corretamente. Se receber valores inesperados, certifique-se de rever os arquivos de zona no seu servidor DNS primário (por exemplo, `db.nyc3.example.com` e `db.10.128`).

Parabéns! Seus servidores DNS internos agora estão configurados corretamente! Agora, vamos falar sobre a manutenção dos seus registros de zona.

Conservando os registros DNS

Agora que você tem um DNS interno funcionando, é preciso conservar seus registros DNS para que eles reflitam com precisão o ambiente do seu servidor.

Como adicionar um Host ao DNS

Sempre que adicionar um host ao seu ambiente (no mesmo datacenter), adicione-o ao DNS. Aqui está uma lista de passos que você precisa seguir:

Servidor de nomes primário

- Arquivo de zona de encaminhamento: adicione um registro “A” para o novo host, incrementando o valor de “Serial”
- Arquivo de zona inversa: adicione um registro “PTR” para o novo host, incrementando o valor de “Serial”
- Adicione o endereço IP privado do seu novo host ao ACL “confiáveis” (`named.conf.options`)

Teste os seus arquivos de configuração:

```
$ sudo named-checkconf
$ sudo named-checkzone nyc3.example.com db.nyc3.example.com
$ sudo named-checkzone 128.10.in-addr.arpa /etc/bind/zones/db.10.128
```

Copy

Então, recarregue o BIND:

```
$ sudo systemctl reload bind9
```

Copy

Seu servidor primário deve estar agora configurado para o novo host.

Servidor de nomes secundário

- Adicione o endereço IP privado do seu novo host ao ACL “confiáveis” (`named.conf.options`)

Verifique a sintaxe de configuração:

```
$ sudo named-checkconf
```

Copy

Então, recarregue o BIND:

```
$ sudo systemctl reload bind9
```

Copy

Seu servidor secundário agora aceitará conexões do novo host.

Configure o novo host para usar o seu DNS

- Configure o `/etc/resolv.conf` para que use seus servidores DNS
- Teste utilizando o `nslookup`

Removendo o host do DNS

Se você remover um host do seu ambiente ou quiser simplesmente removê-lo do DNS, remova todas as coisas que foram adicionadas quando adicionou o servidor ao DNS (ou seja, o inverso dos passos acima).

Conclusão

Agora, é possível consultar as interfaces de rede privadas dos seus servidores por nome ao invés de endereço IP. Isso torna mais fácil a configuração dos serviços e aplicativos porque você já não precisa se lembrar dos endereços IP privados e os arquivos serão mais fáceis de ler e entender. Além disso, é possível agora alterar suas configurações para que apontem para um novo servidor em um único lugar, o seu servidor DNS, ao invés de precisar editar uma variedade de arquivos de configuração distribuídos, facilitando a manutenção.

Assim que tiver seu DNS interno configurado, e os seus arquivos de configuração estiverem usando FQDNs privados para especificar conexões de rede, é **fundamental** que seus servidores DNS estejam devidamente conservados. Se ambos ficarem indisponíveis, seus serviços e aplicativos que dependem deles deixam de funcionar corretamente. É por isso que é recomendável configurar o seu DNS com pelo menos um servidor secundário, além de manter backups funcionais de todos eles.