

# Configuration DNS du shell bash NX-OS

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Configuration DNS du shell Bash](#)

[Vérification DNS du shell Bash](#)

[Étape 1. Vérifiez l'utilisation d'un espace de noms de réseau valide à tester.](#)

[Étape 2. Vérifiez le fonctionnement de la résolution DNS à l'aide du nom d'hôte de l'hôte de test.](#)

[/etc/resolv.conf Format de fichier](#)

[Exemples](#)

[Informations connexes](#)

## Introduction

Ce document décrit les étapes utilisées pour configurer les serveurs DNS dans Bash afin de permettre la résolution des noms d'hôte DNS en adresses IP.

Les périphériques des gammes Cisco Nexus 3000 et 9000 permettent d'accéder au système Linux sous-jacent de NX-OS via Bash (Bourne-again SHell). Bash permet la gestion et la surveillance du système via un environnement Linux. Pour plus d'informations sur Bash on NX-OS, reportez-vous au [chapitre Bash du Guide de programmabilité NX-OS de la gamme Cisco Nexus 9000](#).

Il peut être nécessaire de traduire des noms de domaine humains en adresses IP numériques tout en effectuant des tâches normales dans l'interpréteur de commandes Bash. Ces tâches incluent l'utilisation des utilitaires `curl` ou `wget` pour accéder aux ressources à partir d'un serveur Web ou pour télécharger des images Docker à l'aide de la commande `docker pull`.

## Conditions préalables

### Conditions requises

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

**Note:** Le shell Bash doit être activé sur votre périphérique Cisco Nexus. Reportez-vous à la section « Accéder à Bash » du chapitre Bash du [Guide de programmabilité NX-OS de la gamme Cisco Nexus 9000](#) pour obtenir des instructions sur l'activation du shell Bash.

### Components Used

Les informations contenues dans ce document sont basées sur les versions de matériel et de

logiciel suivantes :

- Plate-forme Nexus 9000 à partir de la version 6.1(2)I2(1) de NX-OS
- Plate-forme Nexus 3000 à partir de la version 6.0(2)U4(1) de NX-OS

Les informations contenues dans ce document ont été créées à partir des périphériques dans un environnement de laboratoire spécifique. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

## Configuration DNS du shell Bash

L'environnement Linux accessible via le shell Bash utilise le fichier `/etc/resolv.conf` pour stocker la configuration DNS, tout comme la plupart des autres systèmes d'exploitation de type Unix.

1. Connectez-vous au shell Bash en tant qu'utilisateur racine via la commande `run bash sudo su -`.

```
Nexus# run bash sudo su -
root@Nexus#whoami
root
```

2. Affichez le contenu actuel du fichier `/etc/resolv.conf`. Dans cet exemple, le fichier est vide.

```
root@Nexus#cat /etc/resolv.conf
root@Nexus#
```

3. Ouvrez le fichier `/etc/resolv.conf` pour le modifier à l'aide de l'éditeur de texte `vi`.

```
root@Nexus#vi /etc/resolv.conf
```

4. Appuyez sur la touche `i` pour passer en mode INSERT, puis saisissez la configuration souhaitée. Reportez-vous à la section `/etc/resolv.conf Format de fichier` de ce document pour plus d'informations sur le format de configuration dans le fichier `/etc/resolv.conf`.

5. Une fois le fichier modifié, appuyez sur la touche `ESC` pour quitter le mode INSERT, puis entrez `:x` pour enregistrer toutes les modifications apportées au fichier et le fermer.

## Vérification DNS du shell Bash

Une fois les modifications apportées à la configuration DNS du shell Bash, vérifiez que les modifications aboutissent à une résolution de nom de domaine réussie. La méthode la plus simple pour tester la résolution de noms de domaine consiste à utiliser l'utilitaire `ping` en utilisant un nom d'hôte de domaine comme cible. Ce document explique comment vérifier une configuration DNS valide à l'aide d'un hôte de test de `test.cisco.com` et de serveurs DNS de 192.168.2.1 et 192.168.2.2.

**Étape 1. Vérifiez l'utilisation d'un espace de noms de réseau valide à tester.**

Par défaut, l'interpréteur de commandes Bash utilise l'espace de noms réseau **par défaut**, sauf instruction contraire. Les espaces de noms réseau sont logiquement équivalents aux VRF NX-OS, et la commande `ip netns` affiche une liste des espaces de noms disponibles pour le shell Bash, comme illustré ci-dessous :

```
root@Nexus#ip netns
EXAMPLE-VRF (id: 2)
management (id: 1)
default (id: 0)
```

Un espace de noms de réseau valide à tester est un espace de noms qui possède une connectivité IP aux serveurs de noms DNS configurés dans le fichier `/etc/resolv.conf`, ainsi qu'une connectivité IP à l'adresse IP à laquelle votre hôte de test résout.

On peut utiliser la commande `ip netns exec {namespace} {desirable-command}` pour exécuter une commande `{desirable-command}` dans l'espace de noms `{namespace}`. Vous pouvez également exécuter le shell Bash dans le contexte d'un espace de noms spécifique à l'aide de la commande `ip netns exec {namespace} bash`. L'ancienne méthodologie est utilisée dans l'exemple ci-dessus, il est vérifié que l'espace de noms **de gestion** a une connectivité IP avec l'adresse IP appartenant à l'hôte `test.cisco.com` (192.168.2.100) et aux deux serveurs DNS (192.168.2.1) ..

```
root@Nexus#ip netns exec management ping 192.168.2.100 -c 5
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
64 bytes from 192.168.2.100: icmp_seq=1 ttl=59 time=0.277 ms
64 bytes from 192.168.2.100: icmp_seq=2 ttl=59 time=0.284 ms
64 bytes from 192.168.2.100: icmp_seq=3 ttl=59 time=0.280 ms
64 bytes from 192.168.2.100: icmp_seq=4 ttl=59 time=0.274 ms
64 bytes from 192.168.2.100: icmp_seq=5 ttl=59 time=0.297 ms

--- 192.168.2.100 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4001ms
rtt min/avg/max/mdev = 0.274/0.282/0.297/0.017 ms
```

```
root@Nexus#ip netns exec management ping 192.168.2.1 -c 5
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data.
64 bytes from 192.168.2.1: icmp_seq=1 ttl=59 time=0.277 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=59 time=0.284 ms
64 bytes from 192.168.2.1: icmp_seq=3 ttl=59 time=0.280 ms
64 bytes from 192.168.2.1: icmp_seq=4 ttl=59 time=0.274 ms
64 bytes from 192.168.2.1: icmp_seq=5 ttl=59 time=0.297 ms

--- 192.168.2.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4001ms
rtt min/avg/max/mdev = 0.274/0.282/0.297/0.017 ms
```

```
root@Nexus#ip netns exec management ping 192.168.2.2 -c 5
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=59 time=0.277 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=59 time=0.284 ms
64 bytes from 192.168.2.2: icmp_seq=3 ttl=59 time=0.280 ms
64 bytes from 192.168.2.2: icmp_seq=4 ttl=59 time=0.274 ms
64 bytes from 192.168.2.2: icmp_seq=5 ttl=59 time=0.297 ms

--- 192.168.2.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4001ms
rtt min/avg/max/mdev = 0.274/0.282/0.297/0.017 ms
```

## Étape 2. Vérifiez le fonctionnement de la résolution DNS à l'aide du nom d'hôte de l'hôte de test.

Utilisez l'utilitaire `ping` avec une cible du nom d'hôte de l'hôte de test. Si des réponses ICMP sont reçues de l'hôte de test et que l'adresse IP contenue dans les réponses ICMP correspond à l'adresse IP à laquelle le nom d'hôte doit être résolu, la résolution DNS fonctionne dans le shell Bash.

Cet exemple montre comment l'utilitaire `ping` est utilisé dans l'espace de noms de **gestion** pour vérifier la résolution DNS correcte. Notez comment le nom d'hôte de domaine de **test.cisco.com** se résout à **192.168.2.100**, qui est l'adresse IP à laquelle nous nous attendons à ce que ce nom d'hôte se résolve.

```
root@Nexus#ip netns exec management ping test.cisco.com -c 5
PING test.cisco.com (192.168.2.100) 56(84) bytes of data.
64 bytes from test.cisco.com (192.168.2.100): icmp_seq=1 ttl=59 time=0.617 ms
64 bytes from test.cisco.com (192.168.2.100): icmp_seq=2 ttl=59 time=0.341 ms
64 bytes from test.cisco.com (192.168.2.100): icmp_seq=3 ttl=59 time=0.310 ms
64 bytes from test.cisco.com (192.168.2.100): icmp_seq=4 ttl=59 time=0.379 ms
64 bytes from test.cisco.com (192.168.2.100): icmp_seq=5 ttl=59 time=0.296 ms

--- test.cisco.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 0.296/0.388/0.617/0.119 ms
```

Si l'espace de noms utilisé pour tester la résolution DNS a une connectivité IP à Internet, on peut envoyer une requête ping à **cisco.com** pour vérifier que les noms de domaine externes peuvent être résolus en plus des noms de domaine internes. Ceci est particulièrement important si l'on a besoin d'utiliser des utilitaires comme `curl` et `wget` contre les serveurs Web publics. Cet exemple montre comment l'utilitaire `ping` peut être utilisé dans l'espace de noms de **gestion** (qui dispose d'une connectivité IP à Internet) pour vérifier la résolution DNS externe correcte.

```
root@Nexus#ip netns exec management ping cisco.com -c 5
PING cisco.com (72.163.4.161) 56(84) bytes of data.
64 bytes from www1.cisco.com (72.163.4.161): icmp_seq=1 ttl=239 time=29.2 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_seq=2 ttl=239 time=29.2 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_seq=3 ttl=239 time=29.3 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_seq=4 ttl=239 time=29.2 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_seq=5 ttl=239 time=29.2 ms

--- cisco.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 29.261/29.283/29.335/0.111 ms
```

## **/etc/resolv.conf** Format de fichier

Certains paramètres de configuration courants sont expliqués ici. Veuillez à modifier tous les paramètres de configuration pour qu'ils correspondent à votre environnement.

- **domain {domain-name.tld}** - Définit un nom de domaine par défaut {domain-name.tld} à ajouter aux noms d'hôtes qui ne se terminent pas par un point. Il ne peut y avoir qu'une seule entrée de `domain` dans le fichier **/etc/resolv.conf**.
- **search {domain-name-1.tld} [domain-name-2.tld...]** - Définit une liste de noms de domaine délimitée par des espaces ({domain-name-1.tld} et, éventuellement, [domain-name-2.tld]) à

ajouter aux noms d'hôte.

**Note:** Les entrées `domain` et `recherche` s'excluent mutuellement - une seule peut être utilisée à la fois. Si les deux entrées sont incluses dans le fichier `/etc/resolv.conf`, l'entrée qui apparaît en dernier dans le fichier est utilisée.

- **nameserver {address-1}** - Définit une adresse IP `{address-1}` pour un serveur DNS vers lequel les requêtes de résolution DNS sont transmises. Plusieurs entrées `nameserver` sont autorisées dans un seul fichier, jusqu'à un maximum de trois.

## Exemples

Cet exemple montre le contenu du fichier `/etc/resolv.conf` où le domaine par défaut de l'environnement est `cisco.com` et les serveurs DNS de l'environnement ont des adresses IP `192.168.2.1` et `192.168.2.2`. Dans ce scénario, si le shell Bash doit résoudre l'adresse IP d'un périphérique avec un nom d'hôte `foo`, il ajoute `cisco.com` à la fin du nom d'hôte de sorte que le nom de domaine complet (FDQN) de l'hôte soit `foo.cisco.com`.

```
domain cisco.com
nameserver 192.168.2.1
nameserver 192.168.2.2
```

L'exemple suivant montre le contenu du fichier `/etc/resolv.conf` dans lequel les noms de domaine `cisco.com` ou `bar.com` peuvent être utilisés pour résoudre les noms d'hôte DNS. Les serveurs DNS de l'environnement ont les adresses IP `192.168.2.1` et `192.168.2.2`. Dans ce scénario, si le shell Bash doit résoudre l'adresse IP d'un périphérique avec un nom d'hôte `foo`, il tente de résoudre `foo.cisco.com` d'abord, puis tente de résoudre `foo.bar.com` ensuite si la résolution pour `foo.cisco.com` échoue.

```
search cisco.com bar.com
nameserver 192.168.2.1
nameserver 192.168.2.2
```

## Informations connexes

- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 9000, version 9.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 9000, version 7.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 9000, version 6.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 3000, version 9.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 3000, version 7.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 3000, version 6.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 3500, version 9.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 3500, version 7.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 3500, version 6.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 3600, version 9.x](#)
- [Guide de programmabilité NX-OS de la gamme Cisco Nexus 3600, version 7.x](#)
- [Programmabilité et automatisation avec Cisco Open NX-OS](#)