

Conseils et astuces Nexus (NX-OS)

Contenu

[Introduction](#)

[Conversion rapide hexadécimale/déc](#)

[Exécution de plusieurs CLI en une seule opération](#)

[Utilisation des variables CLI](#)

[Personnaliser les paramètres de terminal](#)

[Retarder l'exécution des commandes selon les besoins](#)

[Accès à Routing-Context/VRF](#)

[Exécution des CLI pour les VDC respectifs lorsqu'ils sont attachés à LC : N7k spécifique](#)

[Basculement entre le noyau Linux et NX-OS sans quitter le plug-in de débogage](#)

[Chargement de la configuration directement via VSH](#)

[Utilisation des différences : répertorie les différences entre les compteurs.](#)

[Utilisation de l'interface CLI Watch - Spécifique à N9k depuis 7.0\(3\)I2\(x\)](#)

[Vérifier l'utilisation du processeur en interne](#)

[Version de build interne](#)

Introduction

Ce document décrit quelques astuces et astuces qui peuvent améliorer l'expérience de dépannage avec NX-OS.

Conversion rapide hexadécimale/déc

Hex signifie Hexa Decimal , Dec pour Decimal

```
NX-OS# dec 0x89
137
NX-OS# hex 137
0x89
```

Exécution de plusieurs CLI en une seule opération

CLI signifie Command Line Interface

```
N7k-LabSW# show clock ; show switchname ; show license host-id
19:10:59.016 UTC Mon Apr 04 2016
N7k-LabSW
License hostid: VDH=TBM14354170
```

Fonctionne également pour la configuration :

```
N7k-LabSW# conf t ; hostname N7k-LabSW-DEFAULT ; end
Enter configuration commands, one per line. End with CNTL/Z.
N7k-LabSW-DEFAULT#
```

Temps nécessaire à une interface de ligne de commande :

```
N7k-LabSW# show clock ; show run | ex .* ; show clock
07:56:08.709 UTC Tue Apr 05 2016
07:56:08.834 UTC Tue Apr 05 2016
```

Utilisation des variables CLI

```
N7k-LabSW# show cli variables
VSH Variable List (* = session vars)
-----
```

```
SWITCHNAME="N7k-LabSW"
TIMESTAMP="2016-04-05-08.34.18"
```

```
N7k-LabSW# tac-pac bootflash:${SWITCHNAME}-${TIMESTAMP}-show-tech-all.gz
N7k-LabSW# dir bootflash: | i show-tech-all
 15091722   Apr 05 08:41:27 2016  N7k-LabSW-2016-04-05-08.35.57-show-tech-all.gz
```

Vous pouvez également créer vos propres variables :

[Référence CCO](#)

Personnaliser les paramètres de terminal

1) Éviter les délais d'expiration de la session de terminal :

```
N7k-LabSW# show terminal | i Timeout
Session Timeout: 30 minutes
```

```
N7k-LabSW# terminal session-timeout ?
<0-525600> Enter timeout in minutes, 0 to disable
```

```
N7k-LabSW# terminal session-timeout 0
N7k-LabSW# show terminal | i Timeout
Session Timeout: 0 minutes
```

2) Définir la largeur/longueur du terminal d'affichage :

S'avère pratique lorsque vous ne voulez pas que la sortie se casse sur une nouvelle ligne pour les sorties longue largeur

```
N7k-LabSW# show terminal | i Width
Length: 25 lines, Width: 98 columns
```

```
N7k-LabSW# terminal width 300
N7k-LabSW# show terminal | i Width
Length: 25 lines, Width: 300 columns
```

Retarder l'exécution des commandes selon les besoins

```
N7k-LabSW# show clock ; sleep 10 ; show clock
```

19:27:07.435 UTC Mon Apr 04 2016

19:27:17.445 UTC Mon Apr 04 2016

Accès à Routing-Context/VRF

VRF est l'acronyme de Virtual Routing and Forwarding

Exécutez des CLI pour un VRF spécifique sans ajouter de VRF <> à chaque fois à la commande :

```
N7k-LabSW# routing-context vrf management
N7k-LabSW%management# sh ip route
IP Route Table for VRF "management"
 '*' denotes best ucast next-hop
 *** denotes best mcast next-hop
 '[x/y]' denotes [preference/metric]
 '%<string>' in via output denotes VRF <string>

0.0.0.0/0, ubest/mbest: 1/0
   *via 10.48.61.129, [1/0], 1w4d, static
10.48.61.128/26, ubest/mbest: 1/0, attached
   *via 10.48.61.150, mgmt0, [0/0], 1w4d, direct
10.48.61.150/32, ubest/mbest: 1/0, attached
   *via 10.48.61.150, mgmt0, [0/0], 1w4d, local
```

```
N7k-LabSW%management#
N7k-LabSW%management# routing-context vrf default
N7k-LabSW# sh ip route
IP Route Table for VRF "default"
 '*' denotes best ucast next-hop
 *** denotes best mcast next-hop
 '[x/y]' denotes [preference/metric]
 '%<string>' in via output denotes VRF <string>
```

Exécution des CLI pour les VDC respectifs lorsqu'ils sont attachés à LC : N7k spécifique

VDC signifie Virtual Device Context

LC signifie carte de ligne

N7K signifie Nexus 7000

Une fois attachés à LC, les CLI sont exécutés par rapport au VDC par défaut, ou doivent spécifier le numéro de VDC pour la commande associée.

À la place, vous pouvez exécuter les CLI dans l'espace # du VDC, tout comme le contexte de routage.

Exemple :

```
N7k-LabSW# attach module 4
Attaching to module 4 ...
To exit type 'exit', to abort type '$.'
module-4# show hardware mac address-table
FE | Valid| PI|  BD |      MAC      |  Index| Stat| SW  |  Modi| Age| Tmr| GM| Sec| TR| NT| RM|
```


N5600-Lab

```
Linux(debug)# vsh -c "show module"
```

Mod	Ports	Module-Type	Model	Status
1	52	Nexus 56128P Supervisor	N5K-C56128P-SUP	active *

Mod	Sw	Hw	World-Wide-Name(s) (WWN)
1	7.1(1)N1(1)	1.1	--

<snipped>

Chargement de la configuration directement via VSH

VSH signifie Virtual Shell

```
NEXUS-LABSWITCH# echo "hostname LAB" | vsh
```

LAB#

```
Mon Apr 4 20:51:14 2016:type=update:id=10.61.236.97@pts/1:user=admin:cmd=echo "hostname LAB" | vsh (SUCCESS)
Mon Apr 4 20:51:14 2016:type=start:id=vsh.9326:user=admin:cmd=
Mon Apr 4 20:51:14 2016:type=update:id=vsh.9326:user=admin:cmd=configure terminal ; hostname LAB (SUCCESS)
```

```
LAB# echo "hostname NEXUS-LABSWITCH" | vsh
```

NEXUS-LABSWITCH#

```
Mon Apr 4 20:51:23 2016:type=update:id=10.61.236.97@pts/1:user=admin:cmd=echo "hostname NEXUS-LABSWITCH" | vsh (SUCCESS)
Mon Apr 4 20:51:23 2016:type=start:id=vsh.9390:user=admin:cmd=
Mon Apr 4 20:51:23 2016:type=update:id=vsh.9390:user=admin:cmd=configure terminal ; hostname NEXUS-LABSWITCH (SUCCESS)
```

```
N5600-Lab# show run | i hostname | sed "s/N5600-Lab/N56128/" | vsh
```

N56128#

```
N56128# show accounting log | last 3
```

```
Mon Apr 4 21:23:23 2016:type=start:id=vsh.6762:user=admin:cmd=
Mon Apr 4 21:23:23 2016:type=update:id=vsh.6762:user=admin:cmd=configure terminal ; hostname N56128 (SUCCESS)
Mon Apr 4 21:23:23 2016:type=stop:id=vsh.6762:user=admin:cmd=
```

```
N5600-Lab# echo "hostname N56128" > bootflash:CLI
```

N5600-Lab#

```
N5600-Lab# show file CLI
```

hostname N56128

```
N5600-Lab# show file CLI | vsh
```

N56128#

Utilisation des différences : répertorie les différences entre les compteurs.

Diff signifie Différence

```
N5600-Lab# show int mgmt0 | diff
```

```
N5600-Lab# show int mgmt0 | diff
```

11,12c11,12

```
< 9393 input packets 829764 bytes
```

```
<141 unicast packets 3715 multicast packets
---
> 9395 input packets 830217 bytes
>142 unicast packets 3716 multicast packets
15,16c15,16
< 214 output packets 47396 bytes
<27 unicast packets 181 multicast packets
---
> 215 output packets 47640 bytes
>27 unicast packets 182 multicast packets
```

Utilisation de l'interface CLI Watch - Spécifique à N9k depuis 7.0(3)I2(x)

Stands N9K pour Nexus 9000

```
BRU-N9K3-1# watch differences interval 1 show int mgmt0 counter
```

```
Every 1.0s: vsh -c "show int mgmt0
counter"
```

Mon Apr 4 23:42:27 2016

[CCO reference](#)

Vérifier l'utilisation du processeur en interne

CPU signifie unité centrale de traitement

Cette commande fournit une sortie toutes les 5 secondes (peut être modifiée) quel processus consomme le processeur. C'est la même sortie que la commande `top` linux :

```
NEXUS# show system internal processes cpu 5 | no-more
top - 12:41:55 up 117 days, 17:00, 5 users, load average: 0.44, 0.50, 0.55 Tasks: 2883 total, 2
running, 1022 sleeping, 0 stopped, 1859 zombie Cpu(s): 1.0%us, 0.9%sy, 0.0%ni, 98.0%id, 0.0%wa,
0.0%hi, 0.1%si, 0.0%st Mem: 32744992k total, 14563132k used, 18181860k free, 61308k buffers
Swap: 0k total, 0k used, 0k free, 4091160k cached PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+
COMMAND 18465 admin 20 0 5200 3108 1148 R 16.6 0.0 0:00.17 top 6389 root 20 0 699m 53m 14m S 3.3
0.2 113:49.07 netstack 18473 root 20 0 97940 3804 2796 R 3.3 0.0 0:00.02 vsh 10 root 15 -5 0 0 0
S 1.7 0.0 9:10.85 ksoftirqd/2 4876 root 15 -5 369m 27m 6564 S 1.7 0.1 67:33.91 sysmgr 9581 svc-
isan 20 0 446m 41m 11m S 1.7 0.1 21:16.15 isis_l2mp 1 root 20 0 1980 652 572 S 0.0 0.0 3:02.73
init 2 root 15 -5 0 0 0 S 0.0 0.0 0:00.03 kthreadd 3 root RT -5 0 0 0 S 0.0 0.0 0:00.49
migration/0 4 root 15 -5 0 0 0 S 0.0 0.0 22:48.66 ksoftirqd/0 5 root -2 -5 0 0 0 S 0.0 0.0
0:00.00 watchdog/0 6 root RT -5 0 0 0 S 0.0 0.0 0:00.65 migration/1 7 root 15 -5 0 0 0 S 0.0 0.0
2:27.70 ksoftirqd/1 top - 12:42:01 up 117 days, 17:00, 5 users, load average: 0.37, 0.48, 0.54
Tasks: 2887 total, 1 running, 1027 sleeping, 0 stopped, 1859 zombie Cpu(s): 1.0%us, 0.9%sy,
0.0%ni, 98.0%id, 0.0%wa, 0.0%hi, 0.1%si, 0.0%st Mem: 32744992k total, 14562360k used, 18182632k
free, 61308k buffers Swap: 0k total, 0k used, 0k free, 4091160k cached PID USER PR NI VIRT RES
SHR S %CPU %MEM TIME+ COMMAND 18502 admin 20 0 5200 3112 1148 R 18.5 0.0 0:00.17 top 6389 root
20 0 699m 53m 14m S 1.7 0.2 113:49.08 netstack 6402 root -2 0 366m 13m 8660 S 1.7 0.0 1032:41
clx 9015 root 20 0 482m 21m 11m S 1.7 0.1 225:45.72 pm 10066 svc-isan 20 0 403m 22m 9.9m S 1.7
0.1 268:31.28 ospf 10487 svc-isan 20 0 354m 9992 6596 S 1.7 0.0 194:52.11 ecp 1 root 20 0 1980
652 572 S 0.0 0.0 3:02.73 init 2 root 15 -5 0 0 0 S 0.0 0.0 0:00.03 kthreadd 3 root RT -5 0 0 0
S 0.0 0.0 0:00.49 migration/0 4 root 15 -5 0 0 0 S 0.0 0.0 22:48.66 ksoftirqd/0 5 root -2 -5 0 0
0 S 0.0 0.0 0:00.00 watchdog/0 6 root RT -5 0 0 0 S 0.0 0.0 0:00.65 migration/1 7 root 15 -5 0 0
0 S 0.0 0.0 2:27.70 ksoftirqd/1
```

Version de build interne

N7k/N9k

```
N7k-LabSW# show version internal build-identifier
Kickstart image file: bootflash:///n7000-s2-kickstart.6.2.10.bin : S102
System image file: bootflash:///n7000-s2-dk9.6.2.10.bin : S102
```

```
BRU-N9K3-1# show version internal build-identifier
nxos image file: bootflash:///nxos.7.0.3.I2.2a.bin : S30
```

N5k/6k :

```
N5600-Lab# show platform fwm info global | i FwM
    FwM build Details: Built at Sat Apr 18 11:11:18 PDT 2015 on nuo-sw-build40 in directory
    /auto/n5kiluka/daily_build/iplus/nexus/522/src/build by buildsa)
```