Konfiguration von Chiffren, MACs und Kex-Algorithmen auf Nexus-Plattformen

Inhalt

Einleitung

Dieses Dokument beschreibt die Schritte zum Hinzufügen (oder) Entfernen von Ciphers, MACs und Kex Algorithms in Nexus-Plattformen.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie die Grundlagen von Linux und Bash verstehen.

Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf folgenden Hardware- und Software-Versionen:

- Nexus 3000 und 9000 NX-OS 7.0(3)I7(10)
- Nexus 3000 und 9000 NX-OS 9.3(13)
- Nexus 9000 NX-OS 10.2(7)
- Nexus 9000 NX-OS 10.3(5)
- Nexus 7000 NX-OS 8.4(8)
- Nexus 5600 NX-OS 7.3(14)N1(1)

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

Hintergrundinformationen

In manchen Fällen können Sicherheits-Scans schwache Verschlüsselungsmethoden finden, die von Nexus-Geräten verwendet werden. In diesem Fall sind Änderungen an der dcos_sshd_config Datei auf den Switches erforderlich, um diese unsicheren Algorithmen zu entfernen.

Verfügbare Chiffren, MACs und Kex-Algorithmen prüfen

Um zu bestätigen, welche Chiffren, MACs und Kex-Algorithmen eine Plattform verwendet, und dies von einem externen Gerät aus zu überprüfen, können Sie folgende Optionen verwenden:

Option 1: Verwenden der CMD-Leitung vom PC

Öffnen Sie eine CMD-Zeile auf einem PC, der das Nexus-Gerät erreichen kann, und verwenden Sie den Befehl ssh -vvv <hostname> .

<#root>

C:\Users\xxxxx>ssh -vvv <hostname>
--------- snipped ----------
debug2: peer server KEXINIT proposal
debug2:

KEX algorithms: diffie-hellman-group1-sha1,diffie-hellman-group14-sha1,diffie-hellman-group-exchange-sha

debug2: host key algorithms: ssh-rsa
debug2: ciphers ctos: aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,aes192-cbc,aes256-cbc
debug2:

ciphers stoc: aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,aes192-cbc,aes256-cbc <--- encryption algorith

debug2: MACs ctos: hmac-sha1
debug2:

MACs stoc: hmac-sha1 <--- mac algorithms

debug2: compression ctos: none,zlib@openssh.com
debug2:

compression stoc: none,zlib@openssh.com <--- compression algorithms

Option 2: Zugreifen auf die Datei "dcos_sshd_config" mithilfe der **Feature-Bash-Shell**

Dies gilt für:

- N3K mit 7. X, 9. X, 10. X
- Alle N9K-Codes
- N7K mit 8.2 und höher

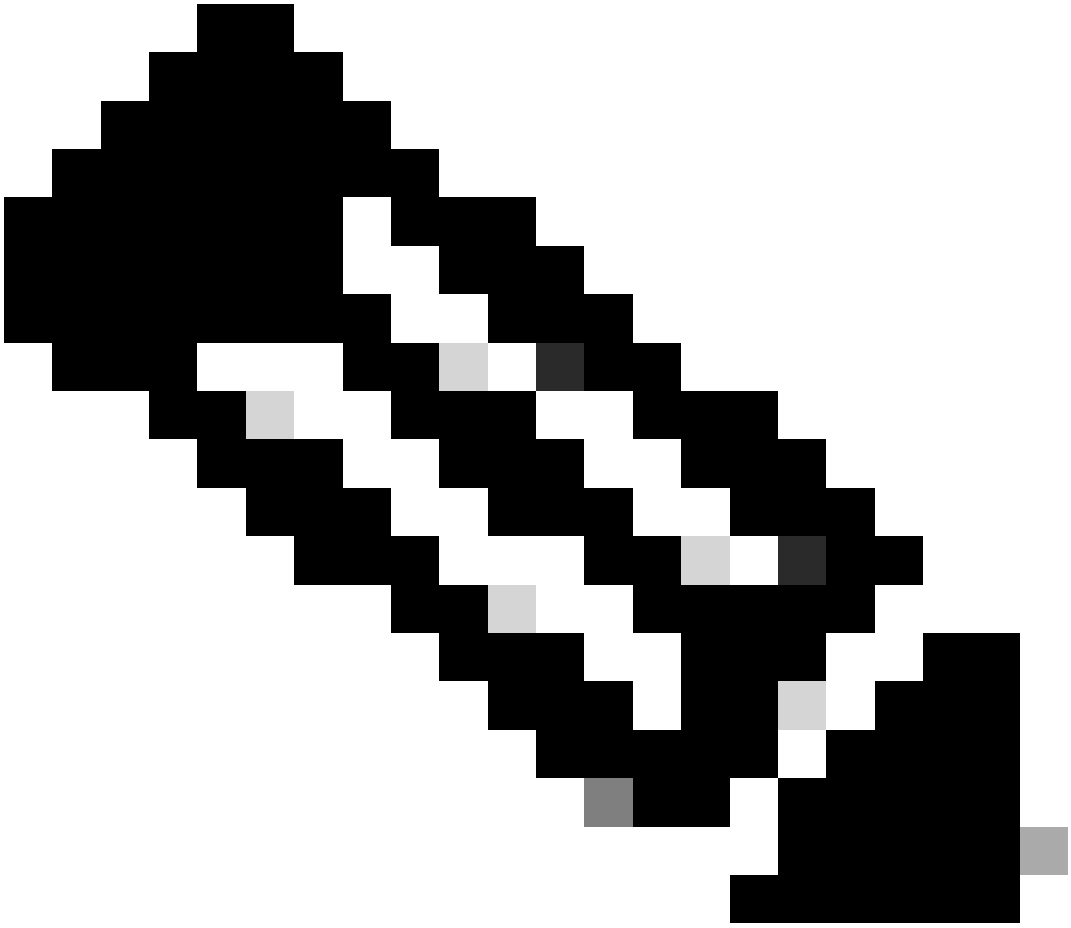
Schritte:

- Aktivieren Sie die bash-shell-Funktion, und wechseln Sie in den bash-Modus:

```
switch(config)# feature bash-shell
switch(config)#
switch(config)# run bash
bash-4.3$
```

2. Überprüfen Sie den Inhalt der dcos_sshd_config Datei:

```
bash-4.3$ cat /isan/etc/dcos_sshd_config
```



Hinweis: Sie können egrep verwenden, um bestimmte Posten anzuzeigen: cat /isan/etc/dcos_sshd_config | grep MAC

Option 3. Zugreifen auf die Datei "dcos_sshd_config" mit einer **Dplug-Datei**

Dies gilt für:

- N3Ks mit 6. X ohne Zugriff auf die bash-shell

    - Alle N5K- und N6K-Codes
- N7Ks mit 6. X und 7. X-Codes

Schritte:

1. Öffnen Sie ein TAC-Ticket, um die dplug-Datei zu erhalten, die mit der auf dem Switch ausgeführten NXOS-Version übereinstimmt.

2. Laden Sie die dplug-Datei auf bootflash hoch und erstellen Sie eine Kopie davon.

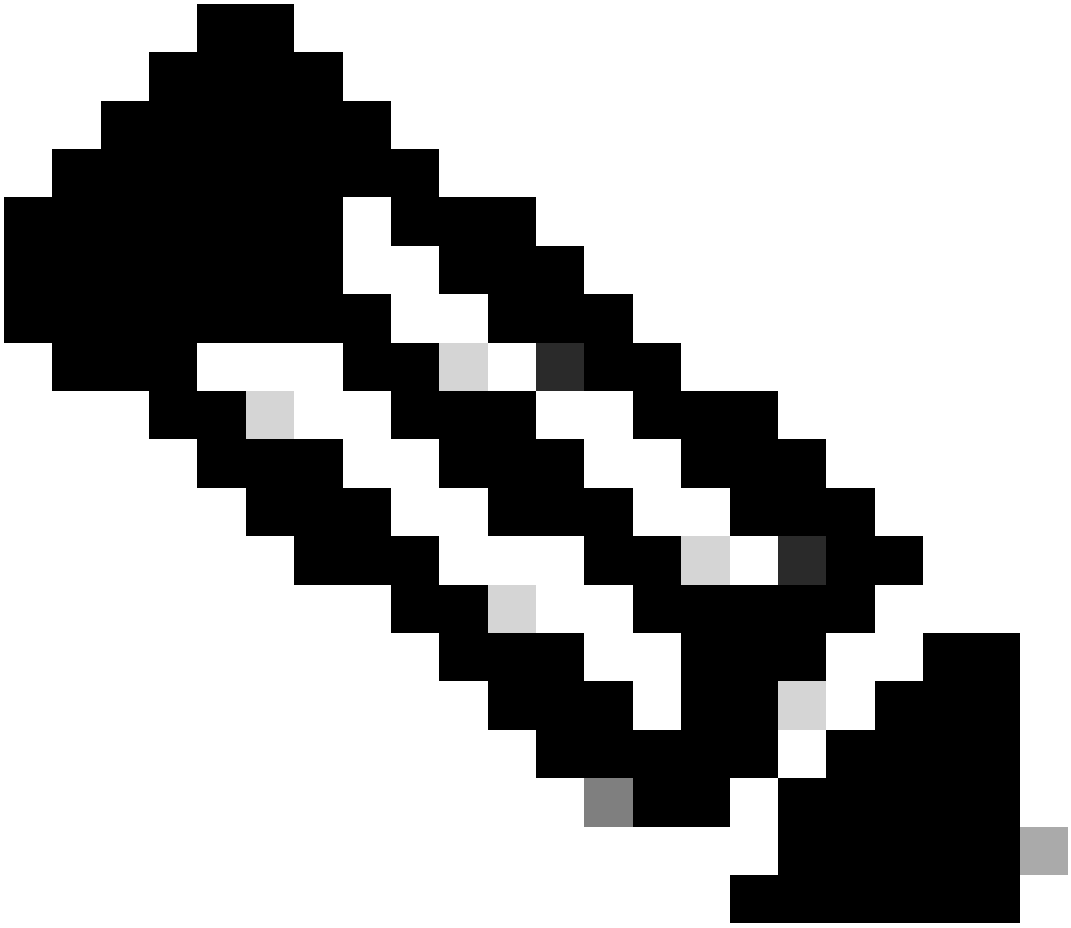
<#root>

switch# copy bootflash:

nuova-or-dplug-mzg.7.3.8.N1.1

bootflash:

dp



Hinweis: Eine Kopie ("dp") der ursprünglichen dplug-Datei wird im bootflash erstellt, sodass nur die Kopie entfernt wird, nachdem der dplug geladen wurde und die ursprüngliche dplug-Datei für weitere Läufe im bootflash verbleibt.

3. Laden Sie die Kopie des dplug über den load Befehl.

<#root>

n5k-1# load bootflash:dp
Loading plugin version 7.3(8)N1(1)
##
Warning: debug-plugin is for engineering internal use only!

For security reason, plugin image has been deleted.

```
############################################################
Successfully loaded debug-plugin!!!
Linux(debug)#
Linux(debug)#
```

2. Datei überprüfendcos_sshd_config.

```
Linux(debug)# cat /isan/etc/dcos_sshd_config
```

Lösung

Schritt 1: Exportieren der Datei "dcos_sshd_config"

1. Senden Sie eine Kopie der dcos_sshd_config Datei an bootflash:

```
Linux(debug)# cd /isan/etc/
Linux(debug)# copy dcos_sshd_config /bootflash/dcos_sshd_config
Linux(debug)# exit
```

2. Vergewissern Sie sich, dass die Kopie im Bootflash ist:

```
switch(config)# dir bootflash: | i ssh
7372 Mar 24 02:24:13 2023 dcos_sshd_config
```

3. Auf einen Server exportieren:

```
switch# copy bootflash: ftp:
Enter source filename: dcos_sshd_config
Enter vrf (If no input, current vrf 'default' is considered): management
Enter hostname for the ftp server: <hostname>
Enter username: <username>
Password:
***** Transfer of file Completed Successfully *****
Copy complete, now saving to disk (please wait)...
Copy complete.
```

4. Nehmen Sie die notwendigen Änderungen an der Datei vor und importieren Sie zurück nach bootflash.

Schritt 2: Importieren der Datei "dcos_sshd_config"

1. Laden Sie die geänderte dcos_sshd_config Datei in den Boot-Flash.

```
switch# copy ftp: bootflash:
Enter source filename: dcos_sshd_config_modified.txt
Enter vrf (If no input, current vrf 'default' is considered): management
Enter hostname for the ftp server: <hostname>
Enter username: <username>
Password:
***** Transfer of file Completed Successfully *****
Copy complete, now saving to disk (please wait)...
Copy complete.
switch#
```

Schritt 3: Ersetzen Sie die ursprüngliche Datei "dcos_sshd_config" durch die Kopie.

Manueller Prozess (bei Neustarts nicht beständig) - alle Plattformen

Durch Ersetzen der vorhandenen dcos_sshd_config Datei unter /isan/etc/ mit einer modifizierten dcos_sshd_config Datei im bootflash. Dieser Prozess ist bei Neustarts nicht persistent

- Eine geänderte ssh config Datei in den Bootflash hochladen:

```
switch# dir bootflash: | i ssh
7372 Mar 24 02:24:13 2023 dcos_sshd_config_modified
```

2. Überschreiben Sie im bash- oder Linux(debug)#-Modus die vorhandene dcos_sshd_config Datei mit der Datei im bootflash:

```
bash-4.3$ sudo su
bash-4.3# copy /bootflash/dcos_sshd_config_modified /isan/etc/dcos_sshd_config
```

3. Bestätigen Sie, dass die Änderungen erfolgreich waren:

```
bash-4.3$ cat /isan/etc/dcos_sshd_config
```

Automatisierter Prozess - N7K

Durch die Verwendung eines EEM-Skripts, das ausgelöst wird, wenn das Protokoll "VDC_MGR-2-VDC_ONLINE" nach einem Neuladen hochgeladen wird. Wenn der EEM ausgelöst wird, wird ein py-Skript ausgeführt, das die vorhandene dcos_sshd_config Datei unter /isan/etc/ durch eine geänderte dcos_sshd_config Datei im Bootflash ersetzt. Dies gilt nur für NX-OS-Versionen, die "feature bash-shell" unterstützen.

- Eine geänderte SSH-Konfigurationsdatei in den Bootflash hochladen:

<#root>

switch# dir bootflash: | i ssh
7404 Mar 03 16:10:43 2023

dcos_sshd_config_modified_7k

switch#

2. Erstellen Sie ein Py-Skript, das die Änderungen auf die dcos_sshd_config Datei anwendet. Stellen Sie sicher, dass Sie die Datei mit der Erweiterung "py" speichern.

<#root>

#!/usr/bin/env python
import os
os.system("sudo usermod -s /bin/bash root")
os.system("sudo su -c \"cp

/bootflash/dcos_sshd_config_modified_7

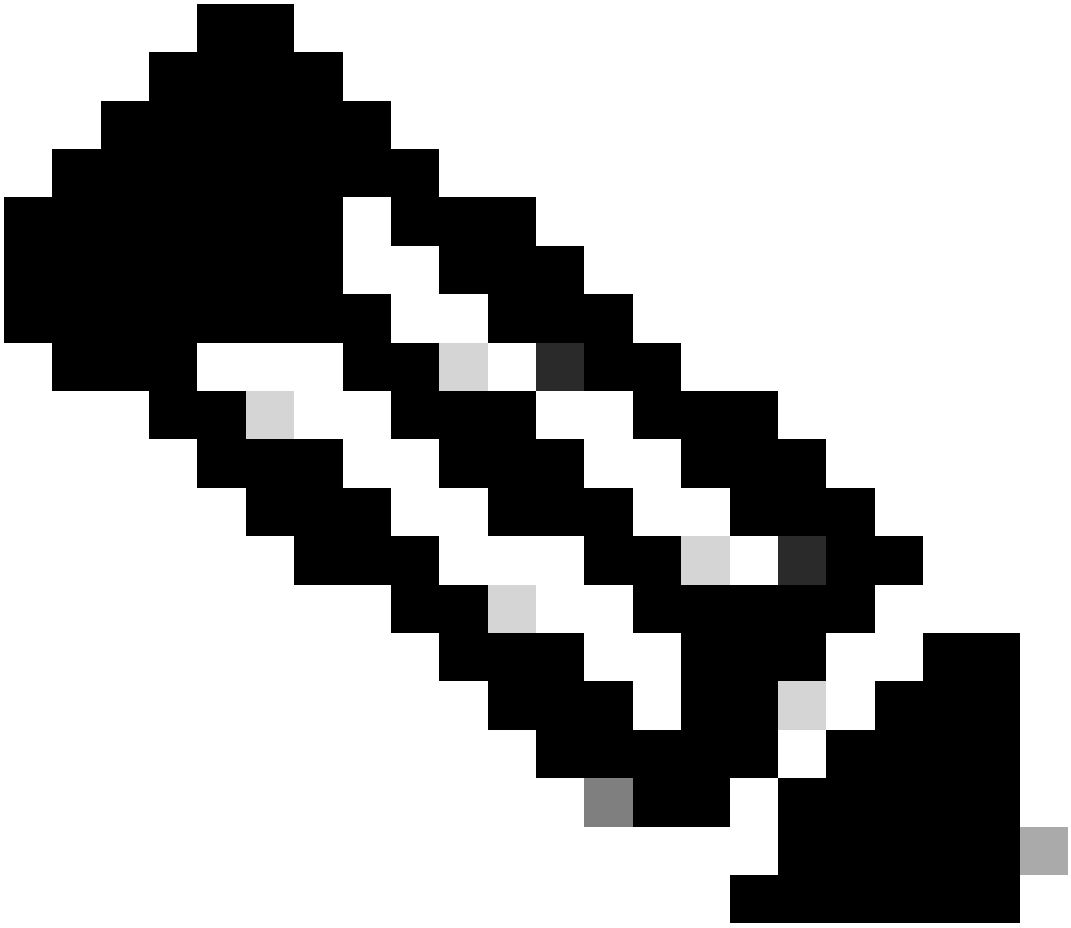
k /isan/etc/dcos_sshd_config\"")

3. Laden Sie das Python-Skript auf bootflash hoch.

<#root>

switch# dir bootflash:///scripts
175 Mar 03 16:11:01 2023

ssh_workaround_7k.py



Hinweis: Python-Skripts sind auf allen Plattformen fast identisch, mit Ausnahme von N7K, das einige zusätzliche Zeilen enthält, um den Cisco Bug zu beheben ID CSCva14865.

4. Stellen Sie sicher, dass der dcos_sshd_config Dateiname aus dem Skript und bootflash (Schritt 1.) identisch sind:

```
<#root>

switch# dir bootflash: | i ssh
      7404    Mar 03 16:10:43 2023

dcos_sshd_config_modified_7k


switch#
```

<#root>

switch# show file bootflash:///

scripts/ssh_workaround_7k.py

```
#!/usr/bin/env python
import os
os.system("sudo usermod -s /bin/bash root")
os.system("sudo su -c \"cp /
```

bootflash/dcos_sshd_config_modified_7k

```
 /isan/etc/dcos_sshd_config\"")
switch#
```

4. Führen Sie das Skript einmal aus, sodass die dcos_sshd_config Datei geändert wird.

<#root>

switch#

source ssh_workaround_7k.py

```
switch#
```

5. Konfigurieren Sie ein EEM-Skript, sodass das py-Skript bei jedem Neustart des Switches und bei jedem erneuten Hochfahren ausgeführt wird.

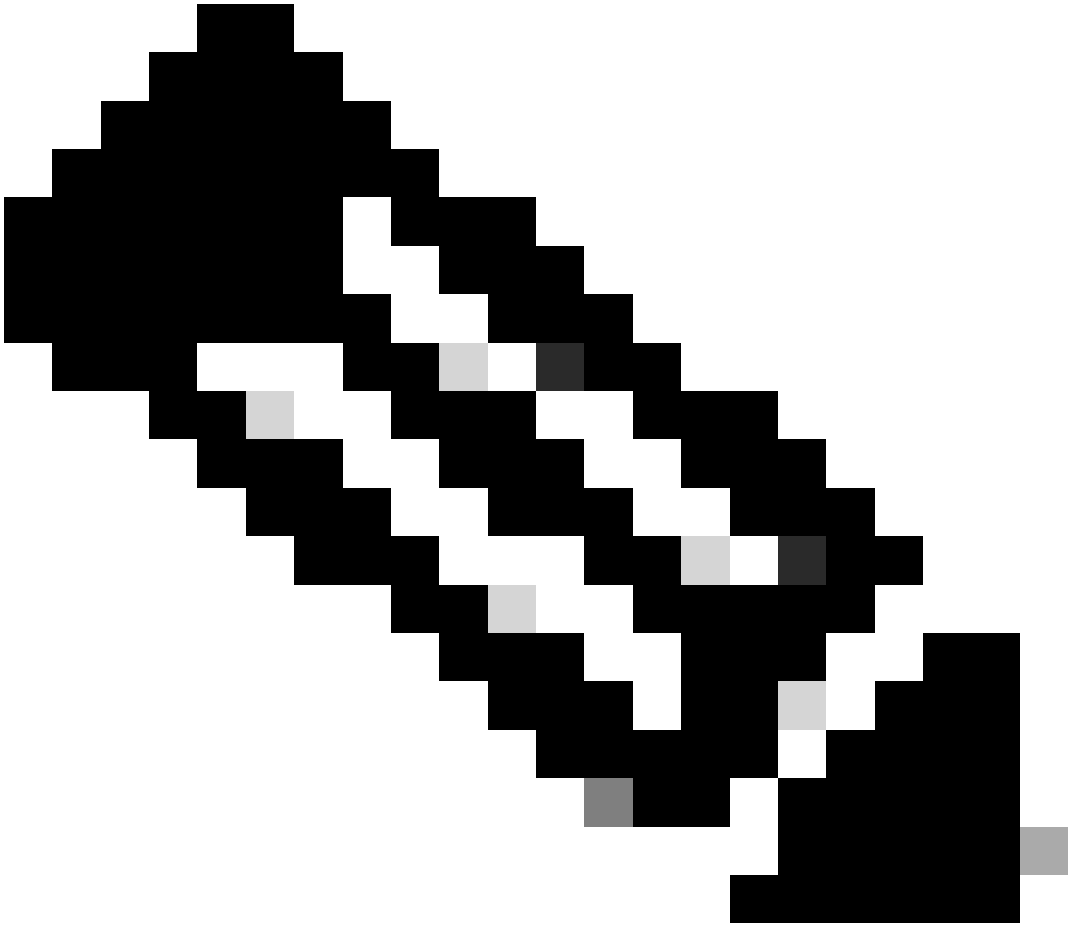
EEM N7K:

<#root>

event manager applet SSH_workaround
  event syslog pattern "vdc 1 has come online"
  action 1.0 cli command

"source ssh_workaround_7k.py"

```
  action 2 syslog priority alerts msg "SSH Workaround implemented"
```



Hinweis: Die EEM-Syntax kann je nach NXOS-Version variieren (einige Versionen erfordern "CLI" und andere "CLI-Befehl"). Stellen Sie deshalb sicher, dass die EEM-Befehle ordnungsgemäß ausgeführt werden.

Automatisierter Prozess - N9K, N3K

- Laden Sie eine geänderte SSH-Konfigurationsdatei in den Bootflash hoch.

<#root>

switch# dir | i i ssh

7732 Jun 18 16:49:47 2024 dcos_sshd_config
7714 Jun 18 16:54:20 2024

dcos_sshd_config_modified

switch#

2. Erstellen Sie ein Py-Skript, das die Änderungen auf die dcos_sshd_config Datei anwendet. Stellen Sie sicher, dass Sie die Datei mit der Erweiterung "py" speichern.

<#root>

#!/usr/bin/env python
import os
os.system("sudo su -c \"cp

/bootflash/dcos_sshd_config_modified

/isan/etc/dcos_sshd_config\"")

3. Laden Sie das Python-Skript auf bootflash hoch.

<#root>

switch# dir | i i .py
127 Jun 18 17:21:39 2024

ssh_workaround_9k.py

switch#

4. Stellen Sie sicher, dass der dcos_sshd_config Dateiname vom Skript und vom Bootflash (Schritt 1.) identisch sind:

<#root>

switch# dir | i i ssh
7732 Jun 18 16:49:47 2024 dcos_sshd_config
7714 Jun 18 16:54:20 2024

dcos_sshd_config_modified

127 Jun 18 17:21:39 2024 ssh_workaround_9k.py
switch#

<#root>

switch# sh file bootflash:ssh_workaround_9k.py
#!/usr/bin/env python
import os
os.system("sudo su -c \"cp

/bootflash/dcos_sshd_config_modified

 /isan/etc/dcos_sshd_config\"")
switch#

4. Führen Sie das Skript einmal aus, sodass die dcos_sshd_config Datei geändert wird.

<#root>

switch#

python bootflash:ssh_workaround_9k.py

5. Konfigurieren Sie ein EEM-Skript, sodass das PY-Skript bei jedem Neustart des Switches und bei jedem erneuten Einschalten ausgeführt wird.

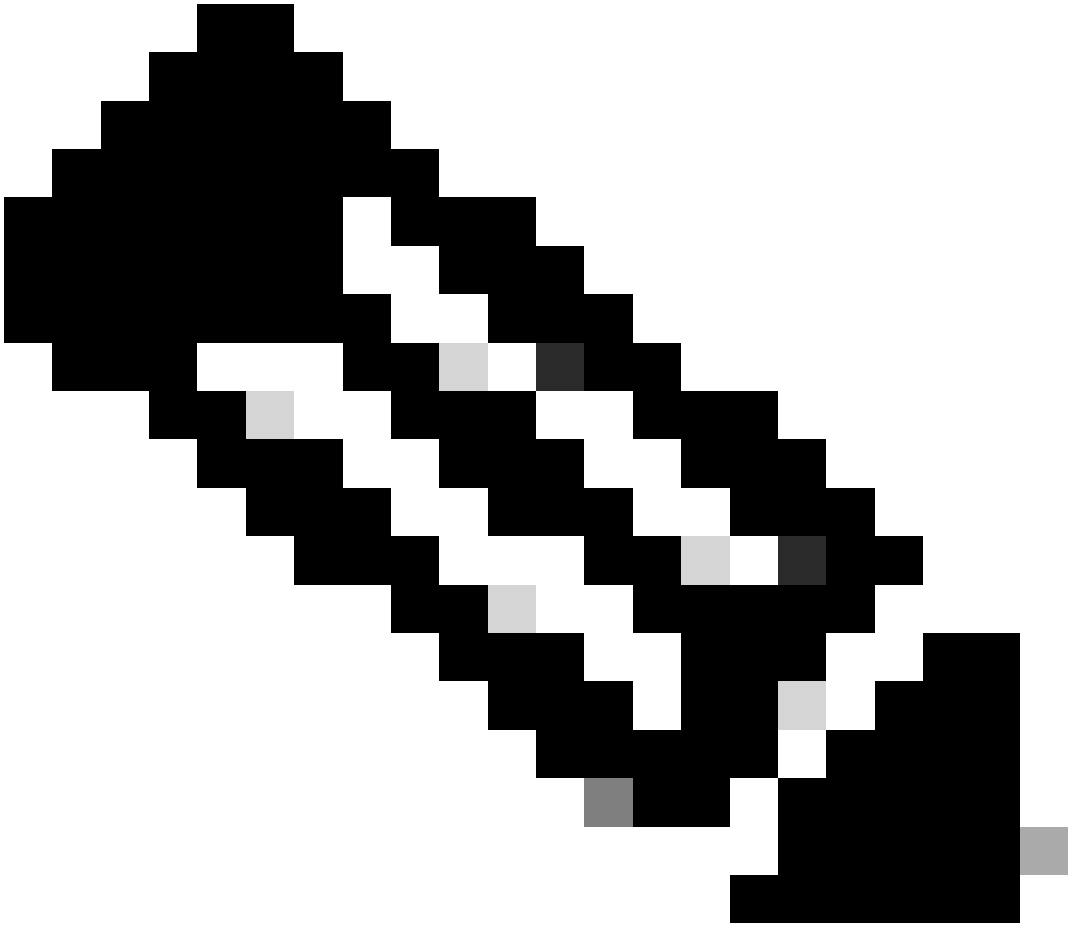
EEM N9K und N3K:

<#root>

event manager applet SSH_workaround
 event syslog pattern "vdc 1 has come online"
 action 1.0 cli

python bootflash:ssh_workaround_9k.py

  action 2 syslog priority alerts msg SSH Workaround implemented



Hinweis: Die EEM-Syntax kann je nach NXOS-Version variieren (einige Versionen erfordern "CLI" und andere "CLI-Befehl"). Stellen Sie deshalb sicher, dass die EEM-Befehle ordnungsgemäß ausgeführt werden.

Automatisierter Prozess - N5K, N6K

Eine modifizierte dplug-Datei wurde mit der Cisco Bug-ID CSCvr23488 erstellt, um diese Kex Algorithms zu entfernen:

- diffie-hellman-group-exchange-sha256
- diffie-hellman-group-exchange-sha1

- diffie-hellman-group1-sha1

Die über die Cisco Bug-ID CSCvr23488 bereitgestellten Debugdateien unterscheiden sich von den Dateien, die für den Zugriff auf die Linux Shell verwendet werden. Öffnen Sie ein TAC-Ticket, um die modifizierte dplug von der Cisco Bug-ID CSCvr23488 zu erhalten.

- Überprüfen Sie die dcos_sshd_config Standardeinstellungen:

<#root>

C:\Users\user>ssh -vvv admin@<hostname>
---- snipped ----
debug2: peer server KEXINIT proposal
debug2:

KEX algorithms: ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-

<--- kex algorithms
debug2:

host key algorithms: ssh-rsa

debug2: ciphers ctos: aes128-ctr,aes192-ctr,aes256-ctr
debug2:

ciphers stoc: aes128-ctr,aes192-ctr,aes256-ctr

<--- encryption algorithms
debug2: MACs ctos: hmac-sha1
debug2:

MACs stoc: hmac-sha1

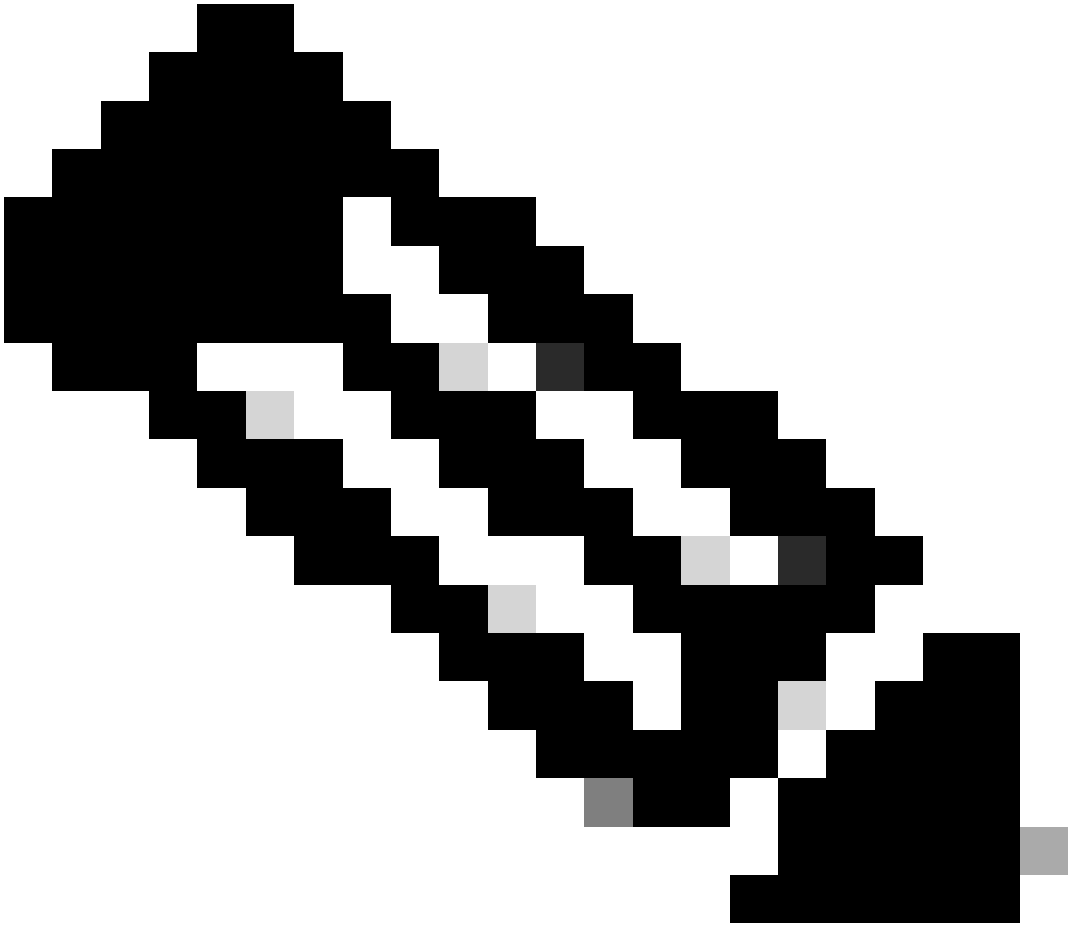
<--- mac algorithms
debug2: compression ctos: none,zlib@openssh.com
debug2:

compression stoc: none,zlib@openssh.com

<--- compression algorithms

2. Erstellen Sie eine Kopie der geänderten dplug-Datei.

switch# copy bootflash:nuova-or-dplug-mzg.7.3.14.N1.1_CSCvr23488.bin bootflash:dp



Hinweis: Eine Kopie ("dp") der ursprünglichen dplug-Datei wird im bootflash erstellt, sodass nur die Kopie entfernt wird, nachdem der dplug geladen wurde und die ursprüngliche dplug-Datei für weitere Läufe im bootflash verbleibt.

---

3. Wenden Sie die dplug-Datei aus der Cisco Bug-ID CSCvr2348 manuell an:

```
switch# load bootflash:dp2
Loading plugin version 7.3(14)N1(1)
################################################################
  Warning: debug-plugin is for engineering internal use only!
  For security reason, plugin image has been deleted.
################################################################
Successfully loaded debug-plugin!!!
```

Workaround for CSCvr23488 implemented
switch#

4. Überprüfen Sie die neuen dcos_sshd_config Einstellungen:

<#root>

C:\Users\user>ssh -vvv admin@<hostname>
    ---- snipped ----
debug2: peer server KEXINIT proposal
debug2:

KEX algorithms: diffie-hellman-group14-sha1,ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521

debug2: host key algorithms: ssh-rsa
debug2: ciphers ctos: aes128-ctr,aes192-ctr,aes256-ctr
debug2:

ciphers stoc: aes128-ctr,aes192-ctr,aes256-ctr

debug2: MACs ctos: hmac-sha1
debug2:

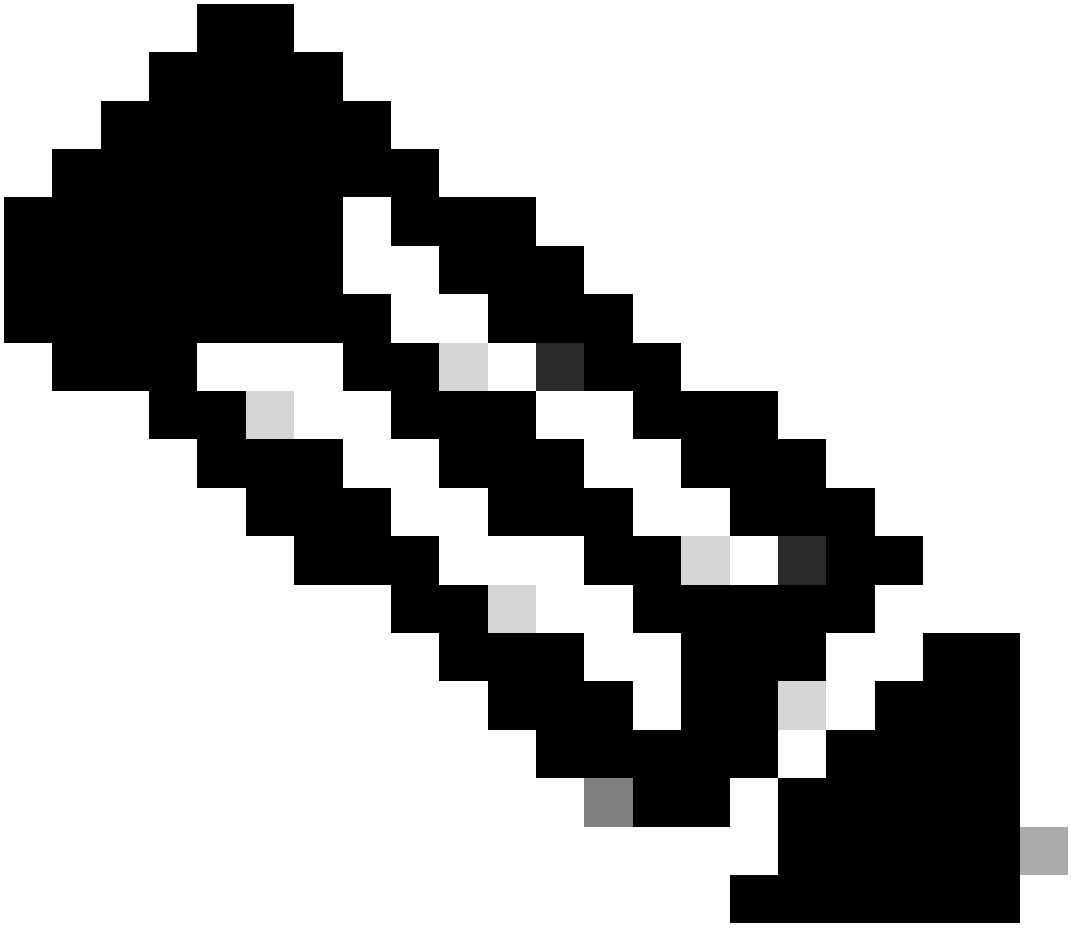
MACs stoc: hmac-sha1

debug2: compression ctos: none,zlib@openssh.com
debug2:

compression stoc: none,zlib@openssh.com

5. Halten Sie diese Änderung über Neustarts hinweg mit einem EEM-Skript aufrecht:

event manager applet CSCvr23488_workaround
event syslog pattern "VDC_MGR-2-VDC_ONLINE"
action 1 cli command "copy bootflash:nuova-or-dplug-mzg.7.3.14.N1.1_CSCvr23488.bin bootflash:dp"
action 2 cli command "load bootflash:dp"
action 3 cli command "conf t ; no feature ssh ;feature ssh"
action 4 syslog priority alerts msg "CSCvr23488 Workaround implemented"



Anmerkung:

- Nachdem der geänderte dplug angewendet wurde, muss die SSH-Funktion auf dieser Plattform zurückgesetzt werden.
- Stellen Sie sicher, dass die dplug-Datei im bootflash vorhanden ist und dass der EEM mit dem richtigen dplug-Dateinamen konfiguriert ist. Der dplug-Dateiname kann je nach Version des Switches variieren. Ändern Sie daher das Skript nach Bedarf.
- Aktion 1 erstellt eine Kopie der ursprünglichen dplug-Datei im Bootflash in eine andere Datei namens "dp", sodass die ursprüngliche dplug-Datei nach dem Laden nicht gelöscht wird.

Überlegungen zur Plattform

N5K/N6K

- MAC (Message Authentication Code) kann auf diesen Plattformen nicht durch Änderung der Datei dcos_sshd_config geändert werden. Die einzige unterstützte MAC ist hmac-sha1.

N7K

- Für das Ändern von MACs ist ein 8.4-Code erforderlich. Weitere Informationen finden Sie unter Cisco Bug ID CSCwc26065.
- "Sudo su" ist in 8.x standardmäßig nicht verfügbar. Referenz Cisco Bug-ID: CSCva14865. Wenn der Befehl ausgeführt wird, wird dieser Fehler beobachtet:

```
<#root>

F241.06.24-N7706-1(config)# feature bash-shell
F241.06.24-N7706-1(config)# run bash
bash-4.3$ sudo su

Cannot execute /isanboot/bin/nobash: No such file or directory   <---

bash-4.3$
```

Geben Sie Folgendes ein, um dies zu überwinden:

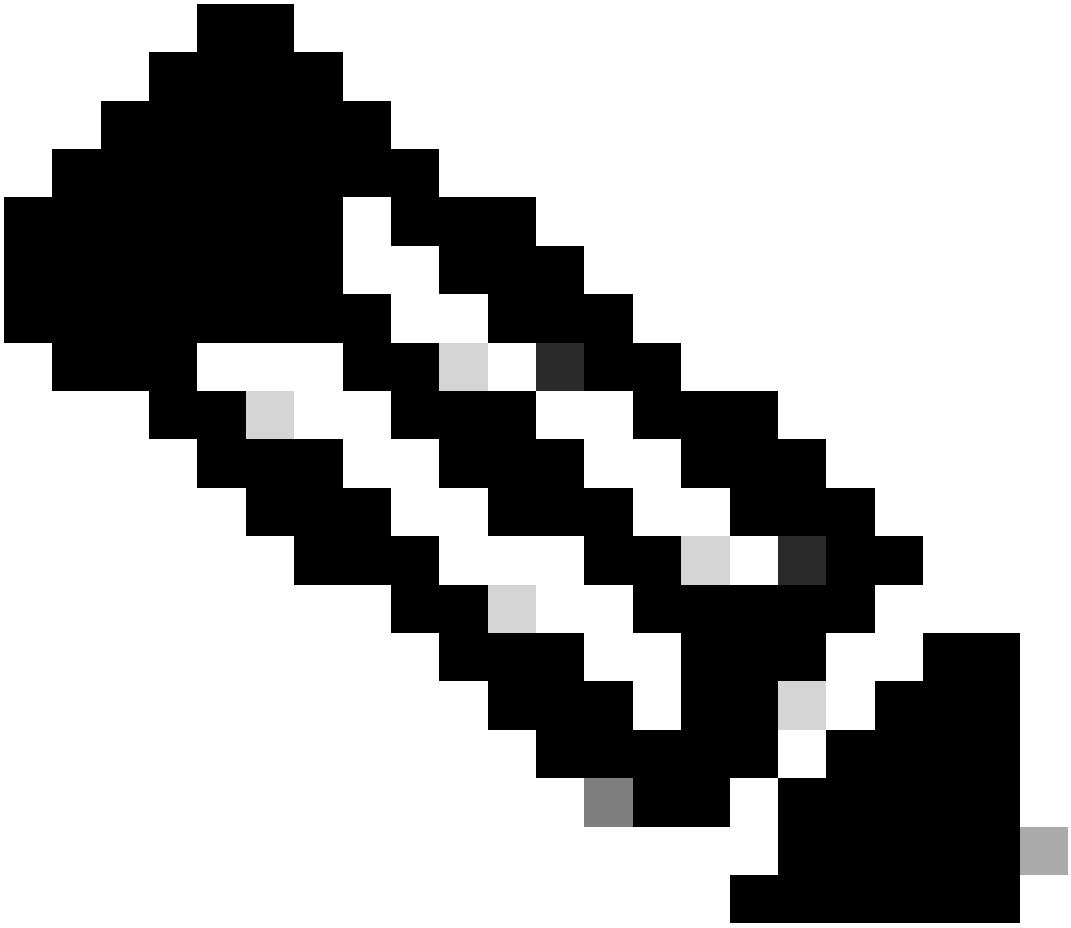
```
<#root>

bash-4.3$

sudo usermod -s /bin/bash root
```

Danach funktioniert "sudo su":

```
bash-4.3$ sudo su
bash-4.3#
```



Hinweis: Diese Änderung besteht nicht nach einem erneuten Laden.

- Es gibt eine separate dcos_sshd_config Datei für jeden VDC. Falls SSH-Parameter in einem anderen VDC geändert werden müssen, stellen Sie sicher, dass die entsprechende dcos_sshd_config Datei geändert wird.

<#root>

N7K# run bash
bash-4.3$ cd /isan/etc/
bash-4.3$ ls -la | grep ssh
-rw-rw-r-- 1 root root 7564 Mar 27 13:48

dcos_sshd_config

```
<--- VDC 1
-rw-rw-r-- 1 root root 7555 Mar 27 13:48
```

dcos_sshd_config.2

```
<--- VDC 2
-rw-rw-r-- 1 root root 7555 Mar 27 13:48
```

dcos_sshd_config.3

```
<--- VDC 3
```

N9K

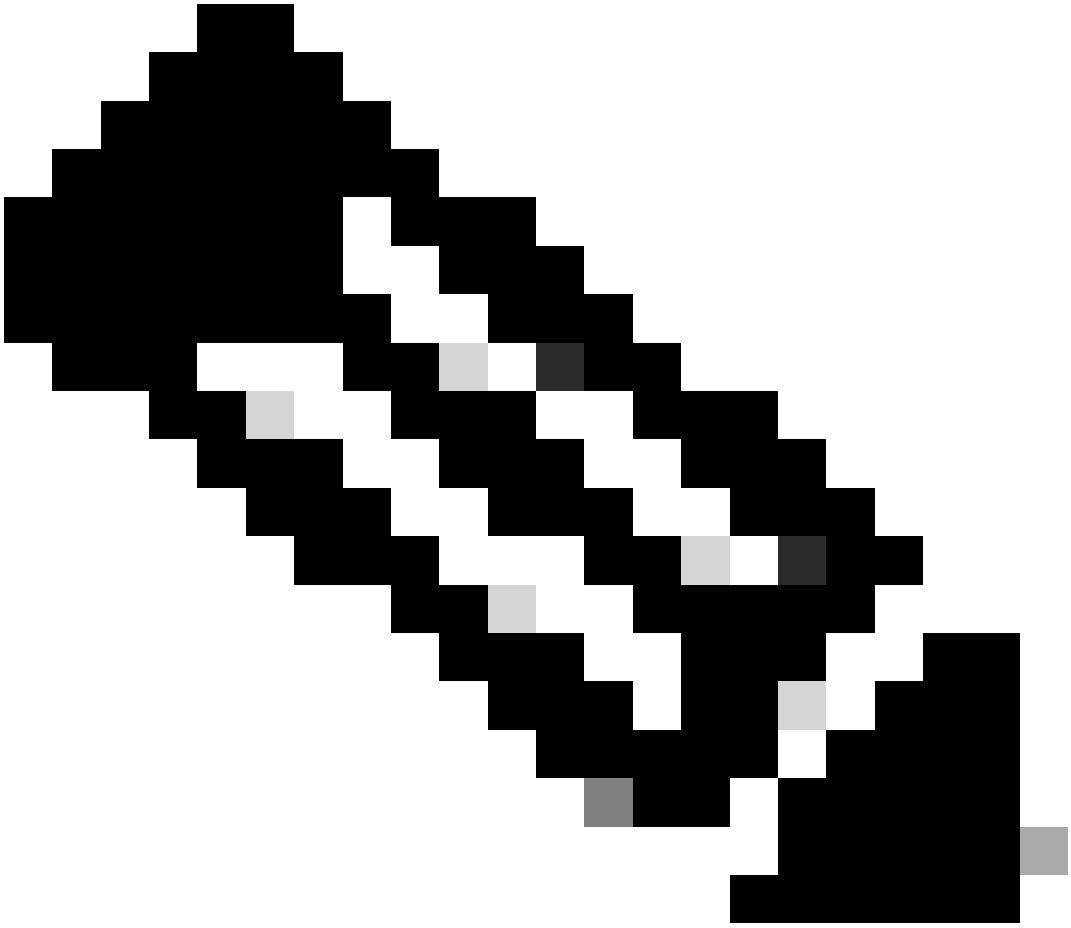
- Änderungen an der dcos_sshd_config Datei bleiben bei Neustarts auf Nexus-Plattformen nicht erhalten. Wenn Änderungen dauerhaft sein müssen, kann ein EEM verwendet werden, um die Datei bei jedem Start des Switches zu ändern. Die Erweiterung auf N9K ändert dies ab 10.4. Weitere Informationen finden Sie unter Cisco Bug ID CSCwd82985.

N7K, N9K, N3K

Es gibt weitere Ciphers, MACs und KexAlgorithms, die bei Bedarf hinzugefügt werden können:

```
<#root>

switch(config)# ssh kexalgos all
switch(config)# ssh macs all
switch(config)# ssh ciphers all
```



Hinweis: Diese Befehle sind auf dem Nexus 7000 mit Version 8.3(1) und höher verfügbar. Für die Nexus 3000-/9000-Plattform ist der Befehl ab Version 7.0(3)I7(8) verfügbar. (Dieser Befehl steht auch allen 9.3(x)-Versionen zur Verfügung. Siehe Cisco Nexus Serie 9000 NX-OS - Sicherheitskonfigurationshandbuch, Version 9.3(x))

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.