# 在Nexus平台上配置密碼、MAC和Kex演算法

## 目錄


簡介

必要條件

- 需求
- 採用元件

背景資訊

- 檢視可用的密碼、MAC和Kex演算法
  - 選項 1.從PC使用CMD線
  - 選項 2.使用Feature Bash-Shell存取「dcos_sshd_config」檔案
  - 選項 3.使用Dplug檔案存取「dcos_sshd_config」檔案

解決方案

- 步驟1.匯出「dcos_sshd_config」檔案
- 步驟2.匯入「dcos_sshd_config」檔案
- 步驟 3.用副本替換原始「dcos_sshd_config」檔案
  - 手動流程（在重新啟動期間不持續）-所有平台
  - 自動化流程- N7K
  - 自動化流程- N9K、N3K
  - 自動化流程- N5K、N6K

平台考量

- N5K/N6K
- N7K
- N9K
- N7K、N9K、N3K


## 簡介

本文檔介紹在Nexus平台上增加（或）刪除Cipher、MAC和Kex演算法的步驟。

## 必要條件

### 需求

思科建議您瞭解Linux和Bash的基本知識。

### 採用元件

本文件中的資訊是以下列硬體與軟體版本為依據：

- Nexus 3000和9000 NX-OS 7.0(3)I7(10)
- Nexus 3000和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)

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

# 背景資訊

有時，安全掃描可能會發現Nexus裝置使用的加密方法較弱。如果發生這種情況，需要對交換機上的dcos_sshd_config檔案進行更改，以刪除這些不安全的演算法。

檢視可用的密碼、MAC和Kex演算法

要確認平台使用的密碼、MAC和Kex演算法，並從外部裝置檢查這些資訊，您可以使用以下選項：

選項 1.從PC使用CMD線

在可以訪問Nexus裝置的PC上打開CMD線並使用命令 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**

選項 2.使用特徵**Bash-Shell**訪問「dcos_sshd_config」檔案

這適用於：

- N3K運行7。X，9。X，10。X
- 所有N9K代碼
- 運行8.2及更高版本的N7K

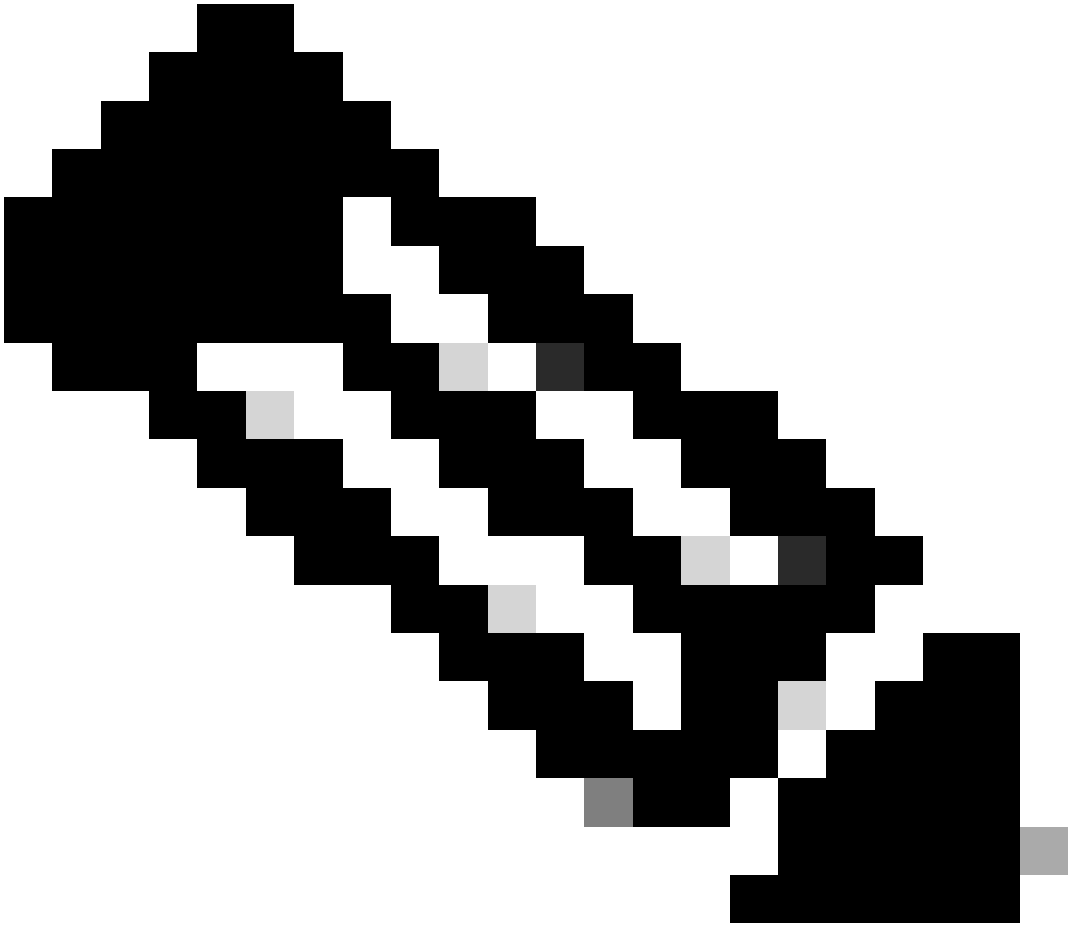
步驟：

- 啟用bash-shell功能並進入bash模式：

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

2. 複查dcos_sshd_config檔案中的內容：

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



**備註**：您可以使用egrep來檢視特定明細行： cat /isan/etc/dcos_sshd_config | grep MAC

---

**選項 3.使用Dplug檔案存取「dcos_sshd_config」檔案**

這適用於：

- N3Ks運行6。無法存取bash-shell的X

- 所有N5K和N6K代碼

- N7Ks運行6。X和7。X代碼

步驟：

1.打開TAC案例以獲取與交換機上運行的NXOS版本匹配的dplug檔案。

2. 將dplug檔案上傳至bootflash並建立其復本。

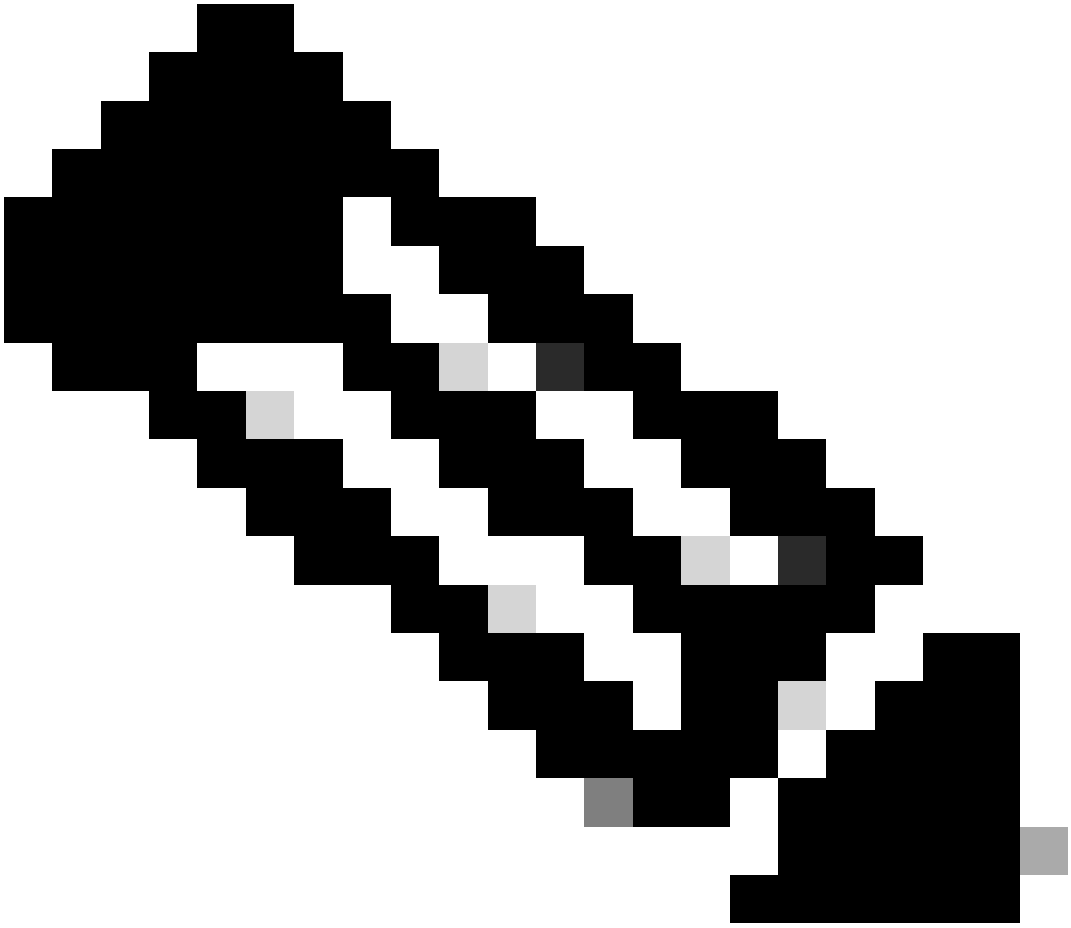
<#root>

switch# copy bootflash:

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

bootflash:

**dp**



**附註**：原始dplug檔案的副本(「dp」)建立在bootflash中，如此一來，只有副本會在載入dplug後移除，而原始dplug檔案仍保留在bootflash中以供後續執行。

3. 透過load 命令載入外掛的副本。

<#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. 複查dcos_sshd_config檔案。

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

**解決方案**

**步驟 1. 匯出「dcos_sshd_config」檔案**

1. 將dcos_sshd_config檔案的一個副本傳送到bootflash：

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

2. 確認副本位於bootflash：

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

3. 匯出至伺服器：

```
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. 對檔案進行必要的變更，並匯入回bootflash。

**步驟 2.** 匯入「dcos_sshd_config」檔案

1. 將修改後的dcos_sshd_config文件上傳到引導快閃記憶體中。

```
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#
```

**步驟 3.用副本替換原始「dcos_sshd_config」檔案**

手動流程（在重新啟動期間不持續） -所有平台

將/isan/etc/下現有的dcos_sshd_config檔案替換為bootflash中修改過的檔案dcos_sshd_config。 此程式在重新啟動後不會持續執行

- 將修改後的ssh config文件上傳到bootflash：

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

2. 在bash或Linux(debug)#模式下，用bootflash中的檔案覆蓋現有dcos_sshd_config檔案：

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

3. 確認變更成功：

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

**自動化流程- N7K**

使用重新載入後日誌「VDC_MGR-2-VDC_ONLINE」啟動時觸發的EEM指令碼。如果EEM被觸發，則會運行一個py指令碼，並用位於bootflash中的修改檔案替換/isan/etc/下的現有dcos_sshd_config檔案dcos_sshd_config。 這僅適用於支援「feature bash-shell」的NX-OS版本。

- 將修改後的ssh配置檔案上傳到bootflash：

<#root>

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

**dcos_sshd_config_modified_7k**

switch#

2. 建立套用變更至dcos_sshd_config檔案的py命令檔。請確定儲存副檔名為「py」的檔案。

<#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**

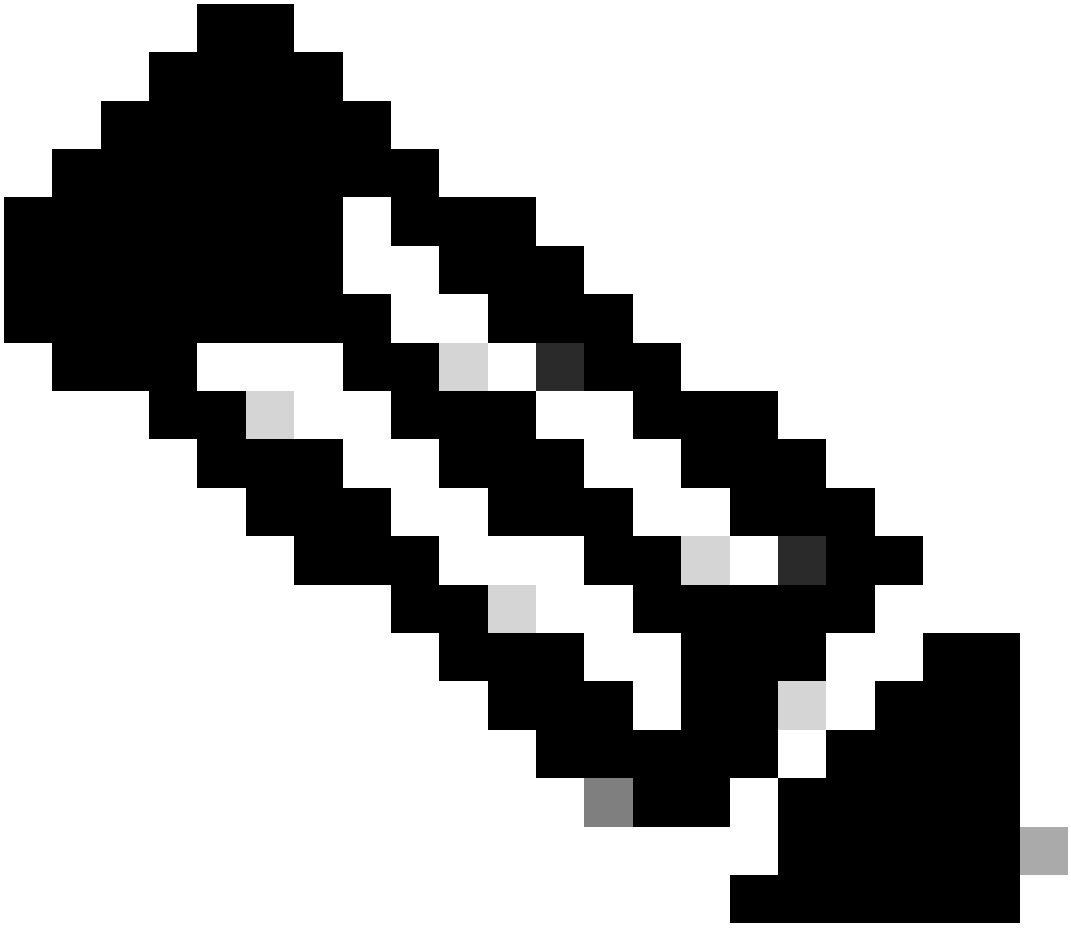
k /isan/etc/dcos_sshd_config\"")

3. 上傳Python指令碼到bootflash。

<#root>

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

**ssh_workaround_7k.py**



**注意**：Python指令碼在所有平台上幾乎都是相同的，但N7K除外，它包含一些用於克服思科漏洞ID CSCva14865的其他行。

---

4. 確保指令碼和bootflash（步驟1）中的檔名相同dcos_sshd_config：

<#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. 執行一次命令檔，以便變更檔案dcos_sshd_config。

<#root>

switch#

**source ssh_workaround_7k.py**

```
switch#
```

5. 配置EEM指令碼，以便每次重新啟動交換機並重新啟動時都運行py指令碼。

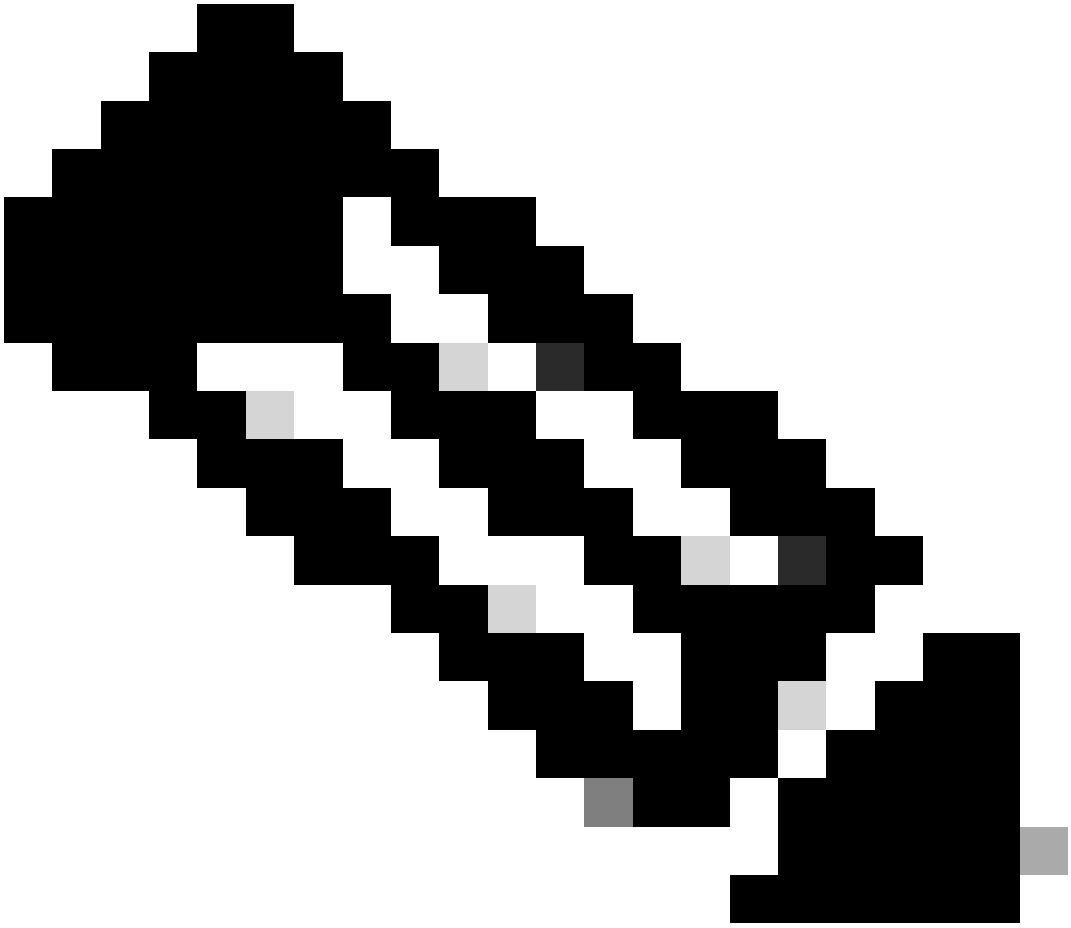
EEM N7K：

<#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"
```



**注意**：EEM語法可能因不同的NXOS版本而異（某些版本需要「CLI」而其它版本需要「CLI命令」），因此請確保正確使用EEM命令。

---

**自動化流程- N9K、N3K**

- 將修改後的SSH配置檔案上傳到bootflash。

<#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. 建立套用變更至dcos_sshd_config檔案的py命令檔。請確定儲存副檔名為「py」的檔案。

<#root>

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

**/bootflash/dcos_sshd_config_modified**

/isan/etc/dcos_sshd_config\"")

3. 上傳python指令碼到bootflash。

<#root>

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

**ssh_workaround_9k.py**

switch#

dcos_sshd_config 4. 確保指令碼和bootflash（步驟1）中的檔名相同：

<#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. 執行一次命令檔，以便變更檔案dcos_sshd_config。

<#root>

switch#

**python bootflash:ssh_workaround_9k.py**

5. 配置EEM指令碼，以便每次重新啟動交換機並重新啟動時都運行py指令碼。

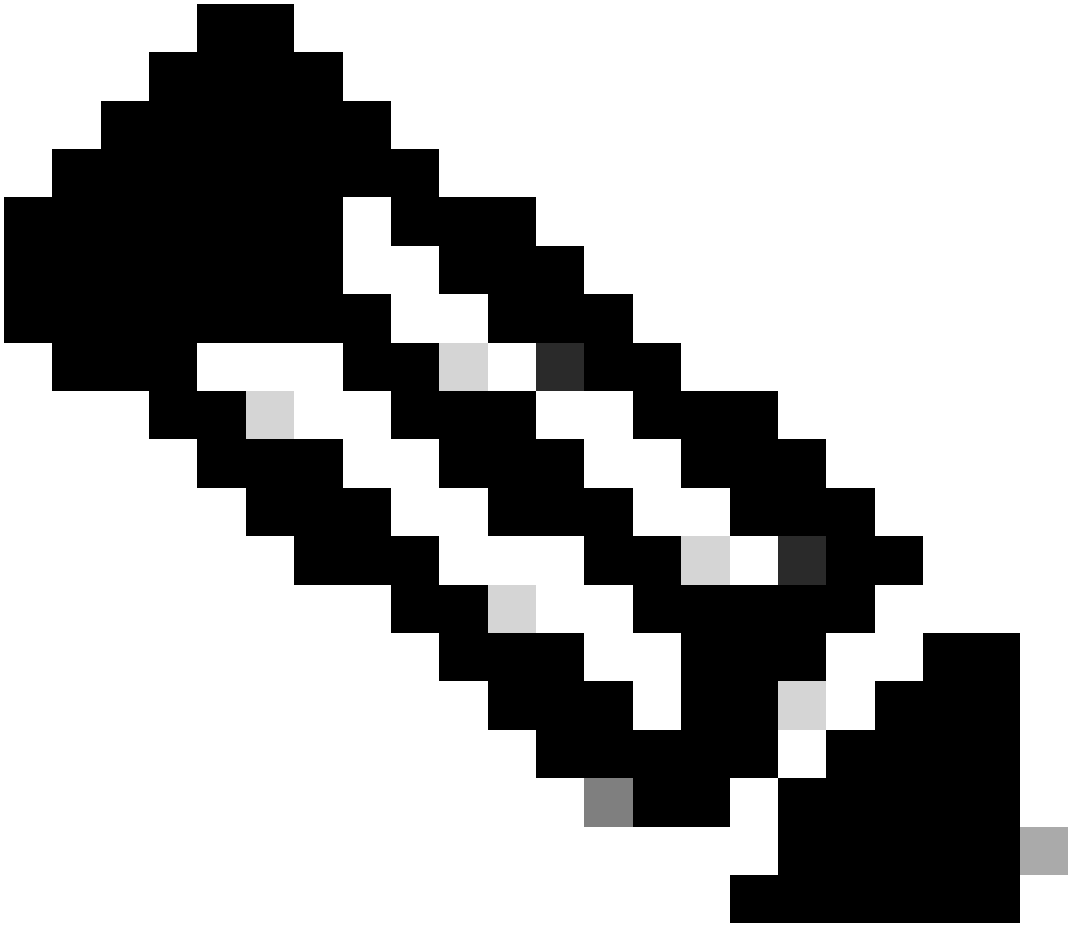
EEM N9K和N3K：

<#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



**注意**：EEM語法可能因不同的NXOS版本而異（某些版本需要「CLI」而其它版本需要「CLI命令」），因此請確保正確使用EEM命令。

---

**自動化流程- N5K、N6K**

透過思科漏洞ID CSCvr23488建立了修改後的dplug檔案，以刪除以下Kex演算法：

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

- diffie-hellman-group1-sha1

透過思科漏洞ID CSCvr23488提供的dpug檔案與用於訪問Linux Shell的dpug檔案不同。打開TAC支援案例以從思科漏洞ID CSCvr23488獲取修改後的dplug。

- 驗證預設dcos_sshd_config設置：

<#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**

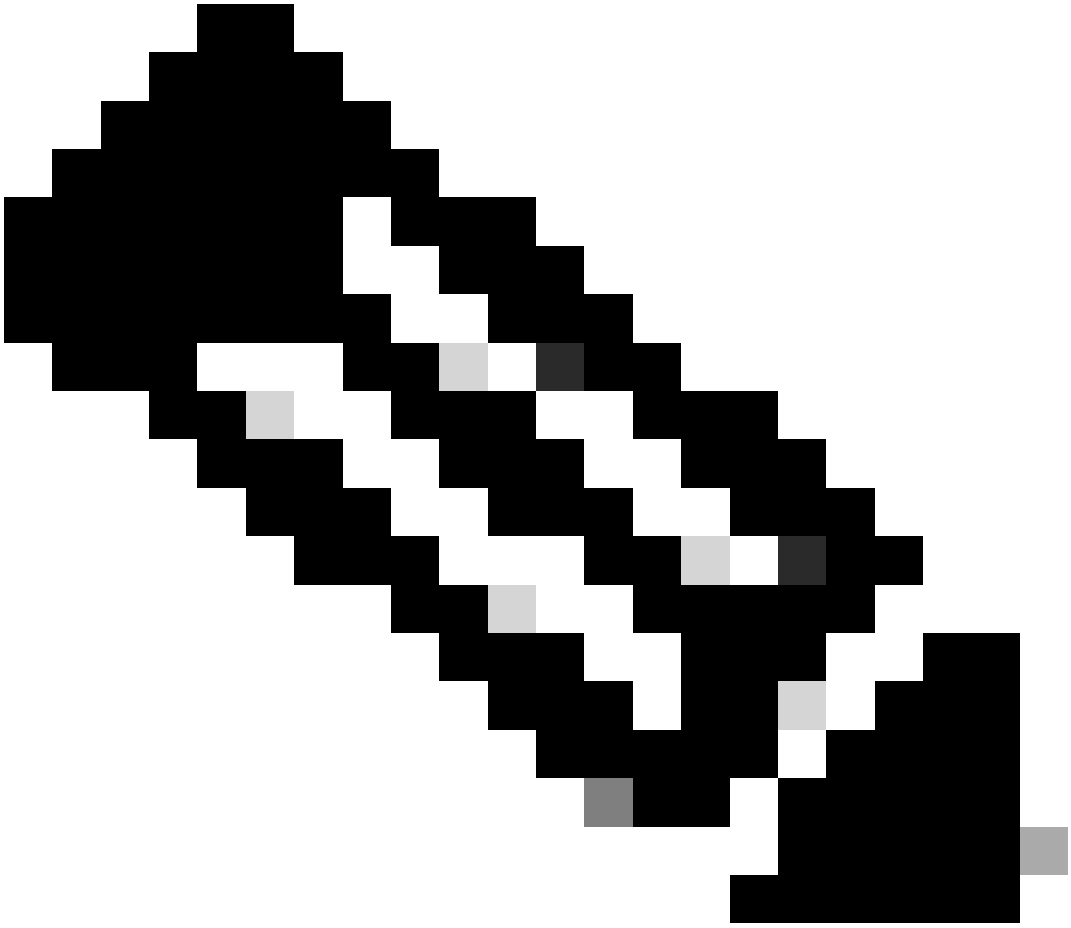
<--- mac algorithms
debug2: compression ctos: none,zlib@openssh.com
debug2:

**compression stoc: none,zlib@openssh.com**

<--- compression algorithms

2. 建立修改後的dplug檔案的副本。

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



**附註**：原始dplug檔案的副本(「dp」)建立在bootflash中，如此一來，只有副本會在載入dplug後移除，而原始dplug檔案仍會保留在bootflash中，以供後續執行之用。

---

3. 手動應用思科漏洞ID CSCvr23488中的dplug檔案：

```
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. 驗證新dcos_sshd_config的設定：

<#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:

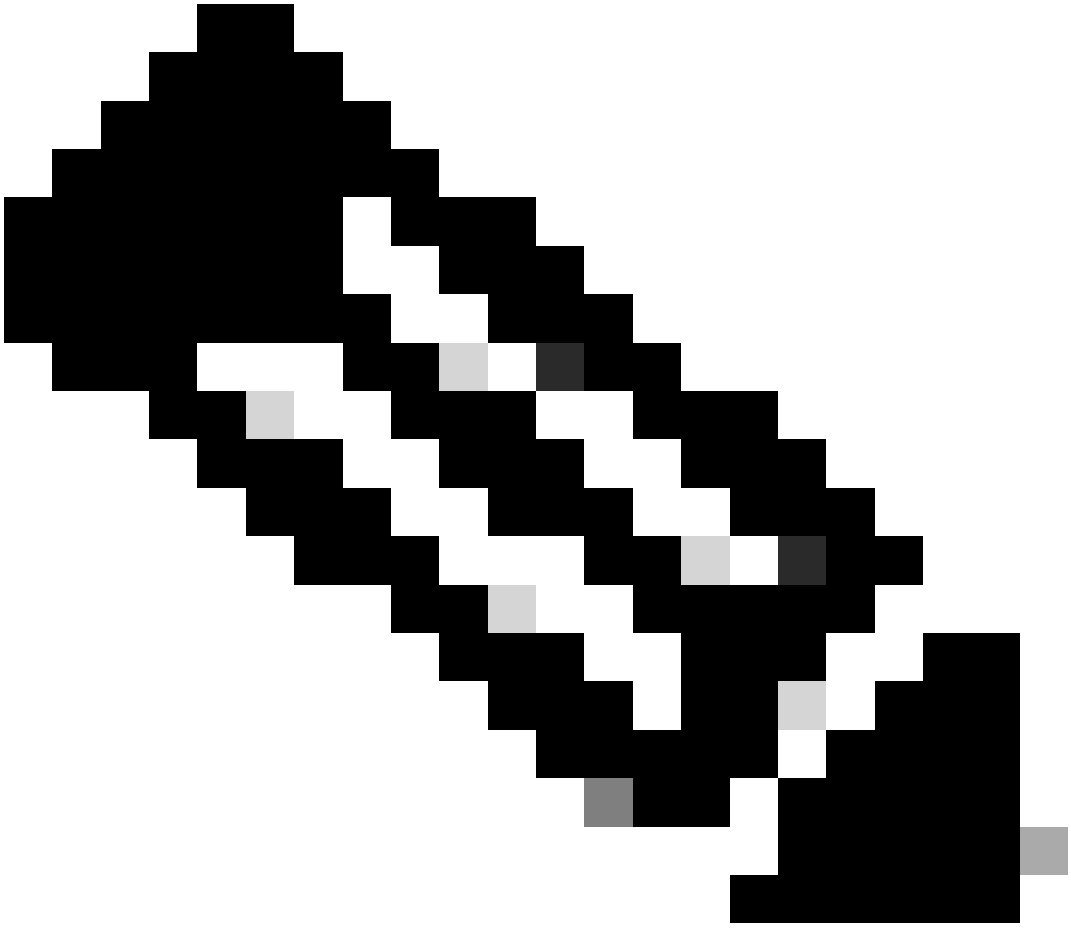
**MACs stoc: hmac-sha1**

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

**compression stoc: none,zlib@openssh.com**

5. 使用EEM程式檔，讓此變更在重新啟動後持續生效：

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"



**附註：**

- 應用修改後的dplug後，必須在此平台上重置SSH功能。
- 確保bootflash中存在dplug檔案，並且使用正確的dplug檔名配置EEM。Dplug檔案名稱會因交換器的版本而異，因此請務必視需要修改指令碼。
- 動作1會在bootflash中建立原始dplug檔案到另一個名為「dp」的檔案的副本，因此原始dplug檔案在載入後不會被刪除。

平台考量

N5K/N6K

- 在這些平台上，不能透過修改dcos_sshd_config檔案來更改MAC(消息驗證代碼)。唯一支援的MAC是hmac-sha1。

N7K

- 要更改MAC，需要使用8.4代碼。 有關詳細資訊，請參閱思科漏洞ID CSCwc26065。
- 預設情況下，「Sudo su」在8.X上不可用。參考思科漏洞ID：CSCva14865。如果執行，則會出現以下錯誤：

<#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$

要克服這種情況，請鍵入：

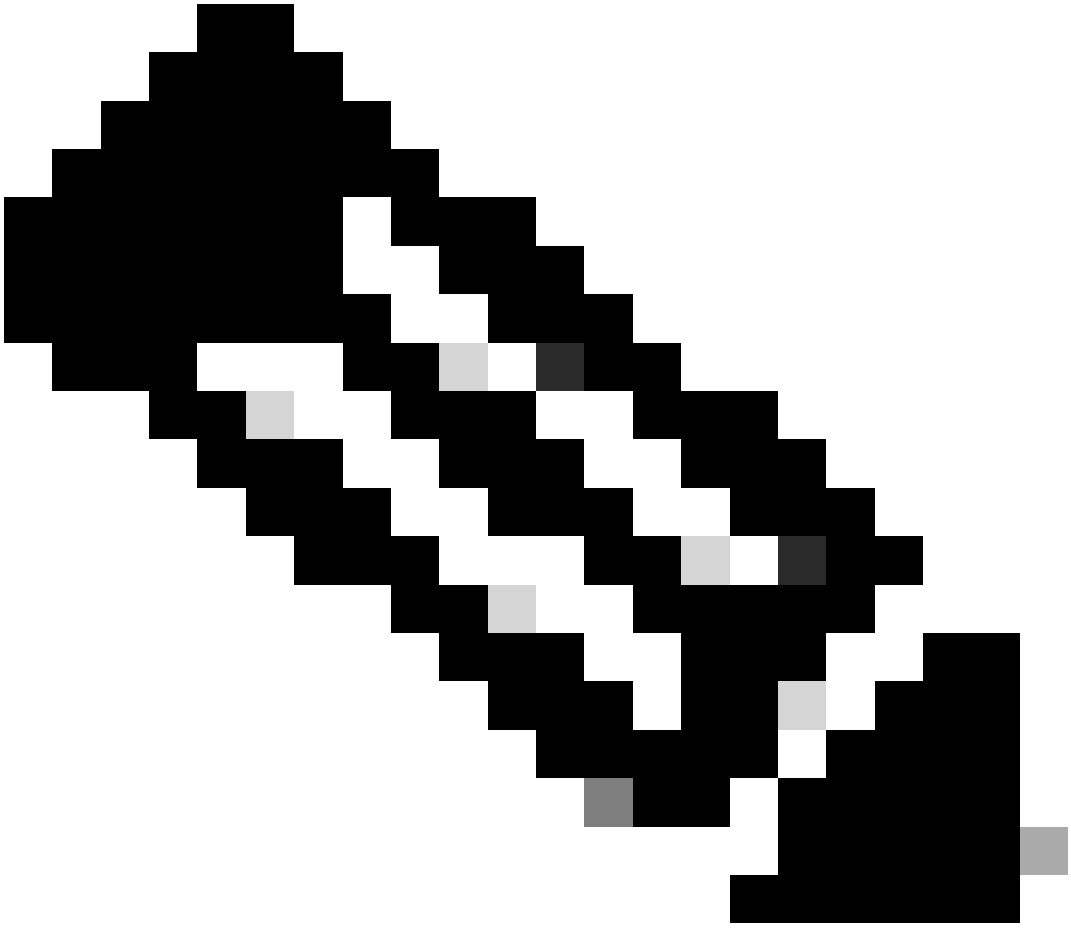
<#root>

bash-4.3$

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

在這個「sudo su」奏效之後：

bash-4.3$ sudo su
bash-4.3#



**附註**：此變更無法在重新載入後生效。

- 每個VDC有一個單獨的dcos_sshd_config檔案，如果需要修改不同VDC上的SSH引數，請確保修改相應的dcos_sshd_config檔案。

<#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

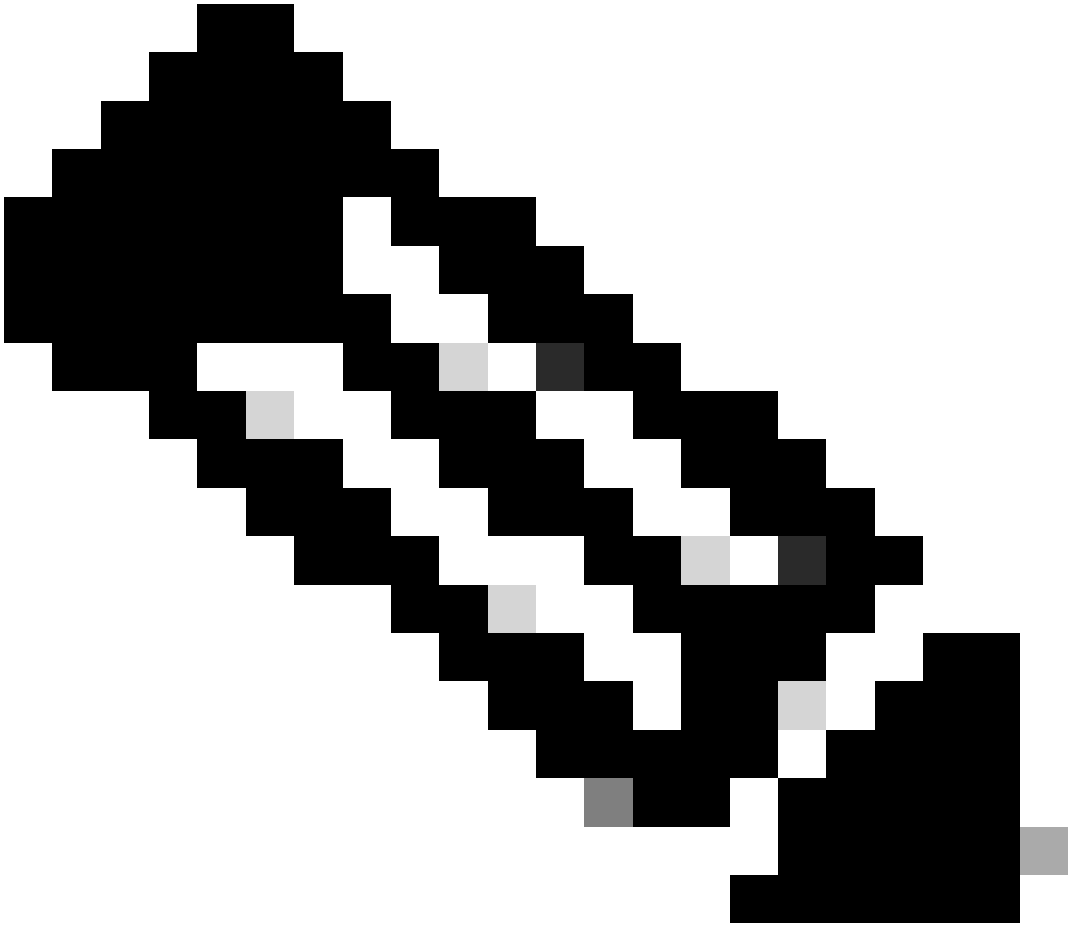
- 在任何Nexus平台上重新啟動後，dcos_sshd_config檔案的更改都不會持續。如果需要持續進行變更，則每次啟動交換機時，都可以使用EEM修改檔案。 在N9K上的增強功能會從10.4開始更改此設定。 有關詳細資訊，請參閱思科漏洞ID CSCwd82985。

**N7K、N9K、N3K**

如果需要，可以增加其他密碼、MAC和KexAlgorithm：

```
<#root>

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



**注意**：這些命令在版本8.3(1)及更高版本的Nexus 7000上可用。對於Nexus 3000/9000平台，7.0(3)I7(8)版及更高版本中提供了該命令。(所有9.3(x)發行版本也使用此命令。請參閱Cisco Nexus 9000系列NX-OS安全配置指南9.3(x)版)

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。