在Nexus平台中配置密码MAC密钥交换算法

目录

简介

先决条件

- 要求
- 使用的组件

背景信息

- 查看可用的密码、MAC和Kex算法
  - 选项1.从PC使用CMD行
  - 选项2.使用Feature Bash-Shell访问“dcos_sshd_config”文件
  - 选项3.使用Dplug文件访问“dcos_sshd_config”文件
- 解决方案
- 步骤1.导出“dcos_sshd_config”文件
- 步骤2.导入“dcos_sshd_config”文件
- 步骤3.使用副本替换原始“dcos_sshd_config”文件
  - 手动流程（在重新启动时不持久）— 所有平台
  - 自动化流程 — N7K
  - 自动化流程 — N9K、N3K
  - 自动化流程 — N5K、N6K

平台注意事项

- N5K/N6K
- N7K
- N9K
- N7K、N9K（10.4和10.5之前的代码）、N3K

简介

本文档介绍在Nexus平台中添加（或）删除Cipher、MAC和Kex算法的步骤。

先决条件

要求

Cisco建议您了解Linux和Bash的基础知识。

使用的组件

本文档中的信息基于下列硬件和软件版本：

- 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)

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

一些安全扫描可以发现Nexus设备使用的加密方法较弱。如果发生这种情况，则需要更改交换机上的dcos_sshd_config文件，以删除不安全的算法。

查看可用的密码、MAC和Kex算法

要确认平台使用哪些密码、MAC和Kex算法，请从外部设备检查这一点。您可以使用以下选项：

选项1.从PC使用CMD行

在可以访问Nexus设备的PC上打开CMD行并运行命令ssh -vvv

<#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 algor

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
```

选项2.使用Feature Bash-Shell访问“dcos_sshd_config”文件

这适用于：

- N3K运行7. X， 9. X， 10. X
- 所有N9K代码
- 运行8.2及更高版本的N7K

1.在bash模式下启用bash-shell功能：

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

2.查看dcos_sshd_config文件中的内容：

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



注意：您可以使用egrep查看特定行：cat /isan/etc/dcos_sshd_config | grep MAC

选项3.使用Dplug文件访问“dcos_sshd_config”文件

这适用于：

- 运行6. X的N3K无法访问bash-shell

- 所有N5K和N6K代码
- 运行6. X和7. X代码的N7Ks

步骤:

1.打开TAC案例以获取与交换机上运行的NXOS版本相匹配的插件文件。

2.将dplug文件上传到bootflash并创建其副本。

<#root>

```
switch# copy bootflash:

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

 bootflash:

dp
```

注意：原始dplug文件的副本(“dp”)在bootflash中创建，因此，只有在dplug加载后删除副本，原始dplug文件仍在bootflash中以供后续运行。

3.通过运行load命令加载插件的副本。

<#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.检查dcos_sshd_config文件。

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

解决方案

步骤1.导出“dcos_sshd_config”文件

1.将dcos_sshd_config文件的副本发送到bootflash:

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

2.确认副本位于bootflash:

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

3.导出到服务器：

```
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.完成对文件的所有必要更改，然后导入回bootflash。

步骤2.导入“dcos_sshd_config”文件

1.将修改的dcos_sshd_config文件上传到引导闪存。

```
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#
```

步骤3.使用副本替换原始“dcos_sshd_config”文件

手动流程（在重新启动时不持久）— 所有平台

使用位于bootflash中的已修改的dcos_sshd_config文件替换/isan/etc/下的现有dcos_sshd_config文件。此过程在重新启动后不会持续

1. 将修改的ssh config文件上传到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，使用引导闪存中的文件覆盖现有的dcos_sshd_config文件：

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

3.确认更改成功：

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

自动化流程 — N7K

通过使用EEM脚本，在重新加载后刷新日志“VDC_MGR-2-VDC_ONLINE”时触发此功能。如果EEM被触发，将运行py脚本，并使用bootflash中经过修改的dcos_sshd_config文件替换/isan/etc/下的现有dcos_sshd_config文件。这仅适用于支持“feature bash-shell”的NX-OS版本。

1. 将修改的ssh配置文件上传到bootflash:

<#root>

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

dcos_sshd_config_modified_7k

```
switch#
```

2.创建将更改应用于dcos_sshd_config文件的py脚本。确保使用“py”扩展名保存文件。

<#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.上传Python脚本到bootflash。

<#root>

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

ssh_workaround_7k.py

注意：Python脚本在所有平台上几乎都是相同的，除了N7K，N7K包含一些额外的行来克服Cisco Bug ID CSCva14865。

4.确保脚本和bootflash（步骤1.）中的dcos_sshd_config文件名相同：

<#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#
```

5.运行一次脚本，因此dcos_sshd_config文件已更改。

<#root>

```
switch#

source ssh_workaround_7k.py


switch#
```

6.配置EEM脚本，一旦每次重新启动交换机时运行py脚本，该脚本就会重新打开。

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语法因不同的NXOS版本而异（某些版本需要“action cli”和其他“action cli command”），请确保EEM命令准确。

自动化流程 — N9K、N3K

1. 将修改的SSH配置文件上传到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.创建将更改应用于dcos_sshd_config文件的py脚本。确保使用“py”扩展名保存文件。

<#root>

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

/bootflash/dcos_sshd_config_modified

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

3.上传python脚本到bootflash。

<#root>

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

ssh_workaround_9k.py

```
switch#
```

4.确保脚本和bootflash（步骤1.）中的dcos_sshd_config文件名相同：

<#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#
```

5.运行一次脚本，因此dcos_sshd_config文件已更改。

<#root>

```
switch#

python bootflash:ssh_workaround_9k.py
```

6.配置EEM脚本，每次运行py脚本后，交换机都会重新启动并重新启动。

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语法因不同的NXOS版本而异（某些版本需要“action cli”和其他“action cli command”），请确保EEM命令准确。

自动化流程 — N5K、N6K

已通过Cisco Bug ID CSCvr2348创建修改的dplug文件，以删除以下Kex算法：

- diffie-hellman-group-exchange-sha256
- diffie-hellman-group-exchange-sha1
- diffie-hellman-group1-sha1

通过Cisco Bug ID CSCvr23488提供的dpug文件与用于访问Linux Shell的dpug文件不同。打开TAC案例，从Cisco Bug ID CSCvr2348获取修改后的dplug。

1. 验证默认dcos_sshd_config设置：

<#root>

```
C:\Users\user>ssh -vvv admin@


      ---- 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.创建修改后的dplug文件的副本。

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

注意：在bootflash中创建原始dplug文件的副本(“dp”)，只有在dplug加载后删除副本，并且原始dplug文件仍保留在bootflash中以供后续运行。

3.手动应用来自Cisco Bug ID CSCvr23488的插件文件：

```
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.验证新的dcos_sshd_config设置：

<#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脚本确保在重新启动后此更改保持不变：

```
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"
```

注意：应用修改后的dplug后，必须在此平台上重置SSH功能。bootflash中必须存在dplug文件，并且EEM必须配置正确的dplug文件名。插头文件名可能因交换机的版本而异，请确保根据需要修改脚本。操作1在bootflash中将原始dplug文件复制到另一个名为“dp”的文件中，因此加载后不会删除原始dplug文件。

平台注意事项

N5K/N6K

- 不能通过修改dcos_sshd_config文件来更改这些平台上的MAC（消息身份验证代码）。唯一支持的MAC是hmac-sha1。

N7K

- 要更改MAC，需要8.4代码。有关详细信息，请参阅Cisco Bug ID CSCwc26065。
- 默认情况下，“Sudo su”在8.X上不可用。参考思科漏洞ID:CSCva14865。 如果执行，则会出现以下错误：

<#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$
```

要解决此问题，请运行：

<#root>

```
    bash-4.3$

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

运行之前的命令后，“sudo su”将工作：

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

注意：此更改在重新加载后无法生效。

每个VDC都有一个单独的dcos_sshd_config文件，如果必须在不同的VDC上修改SSH参数，请确保修改相应的dcos_sshd_config文件。

<#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平台上重新启动后，dcos_sshd_config文件上的更改都不会持续。如果更改必须是永久性的，则每次交换机启动时，都可以使用EEM修改文件。
- 对N9K更改的增强功能以10.4(2)开头，10.5(1)中也有此功能。 它未被添加到之前软件系列的更新版本中。
- 有关详细信息，请参阅Cisco Bug ID CSCwd82985。

运行10.5(1)的交换机上的CLI示例：

```
switch(config)# ssh ?
cipher-mode          Set Cipher-mode for ssh
ciphers              Ciphers to encrypt the connection <<<<<<<<<
idle-timeout         SSH Client session idle timeout value
kexalgos             Key exchange methods that are used to generate per-connection keys <<<<<<<<<
key                  Generate SSH Key
keytypes             Public key algorithms that the server can use to authenticate itself to the client
login-attempts       Set maximum login attempts from ssh
login-gracetime      Set login gracetime for ssh connection
macs                 Message authentication codes used to detect traffic modification <<<<<<<<<
port                 Set port number for ssh
rekey                Renegotiate ssh key

switch(config)# ssh ciphers ?
WORD                 Algorithm name to be configured (Max Size 128)
aes256-gcm           <Deprecated> enable aes256-gcm
all                  Control known weak SSH algorithms in current version of NX-OS in addition to the b

switch(config)# ssh macs ?
WORD                 Algorithm name to be configured (Max Size 128)
```

```
all                  Control known weak SSH algorithms in current version of NX-OS in addition to the b

switch(config)# ssh kexalgos ?
WORD                 Algorithm name to be configured (Max Size 128)
all                  Control known weak SSH algorithms in current version of NX-OS in addition to the b
```

N7K、N9K（10.4和10.5之前的代码）、N3K

对于运行早期代码的平台，您可以添加密码、MAC和KexAlgorithms。这些版本不提供单独选择特定算法的选项；仅支持“all”选项。

```
N9000(config)# ssh macs ?
all                  Enable algorithms supported in current version of SSH

N9000(config)# ssh kexalgos ?
all                  Enable algorithms supported in current version of SSH
ecdh-sha2-nistp384   Enable ecdh-sha2-nistp384

N9000(config)# ssh ciphers ?
aes256-gcm           Enable aes256-gcm
all                  Enable algorithms supported in current version of SSH
```

注意：这些命令在版本8.3(1)及更高版本的Nexus 7000上可用。对于Nexus 3000/9000平台，该命令在版本7.0(3)I7(8)及更高版本中可用。(所有9.3(x)版本也使用此命令。请参阅Cisco Nexus 9000系列NX-OS安全配置指南，版本9.3(x)。

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。