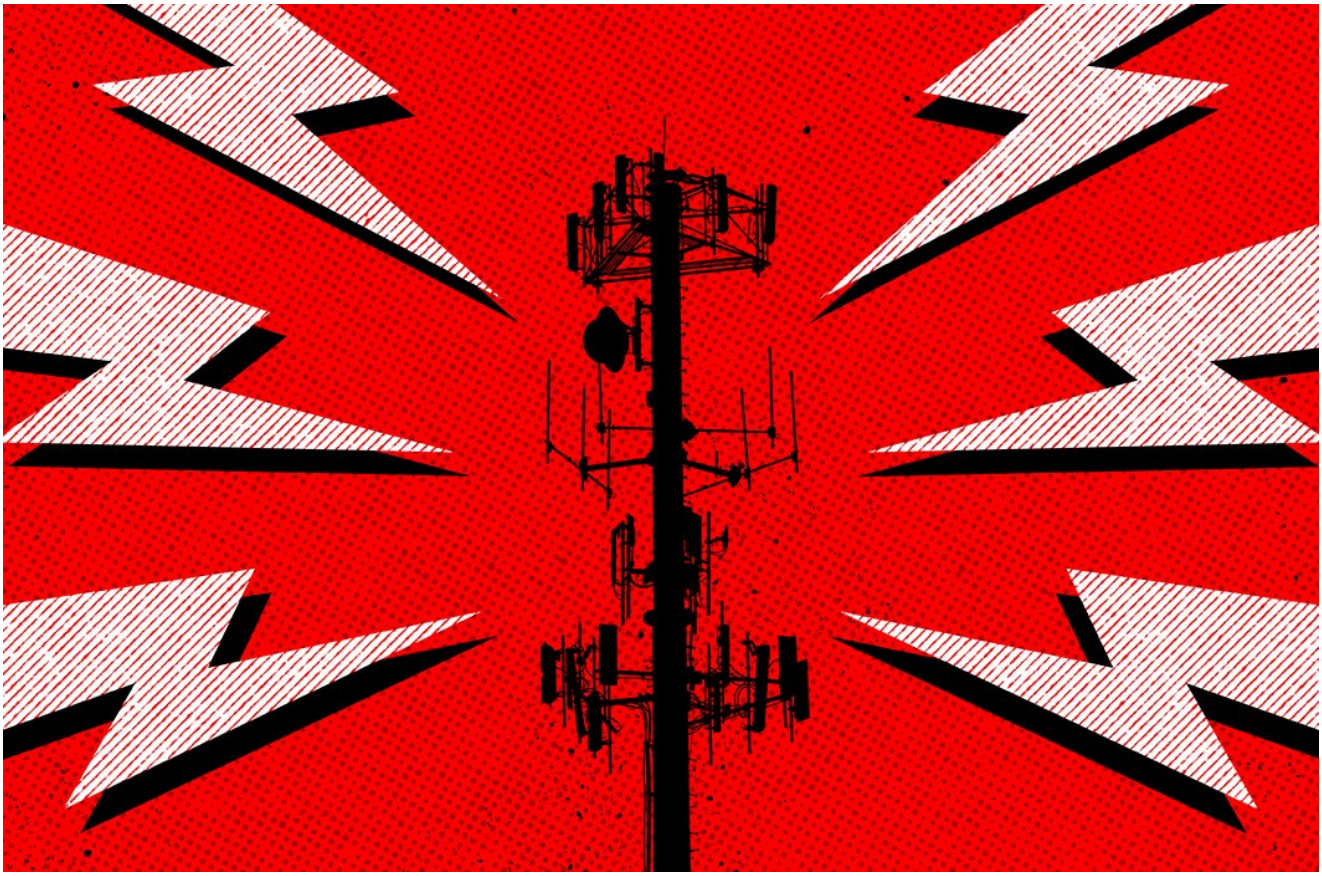
How to Hunt for DecisiveArchitect and Its JustForFun Implant

crowdstrike.com/blog/how-to-hunt-for-decisivearchitect-and-justforfun-implant/

Jamie Harries May 25, 2022



The security landscape is constantly developing to provide easier ways to establish endpoint visibility across networks through the use of endpoint detection and response (EDR) utilities. However, certain challenges still remain, particularly as a result of many organizations' need for systems running legacy or proprietary operating systems, such as Solaris. If such systems are not adequately protected using other security controls or unless they can only be accessed by systems with appropriate endpoint-based detection/prevention capabilities, this can cause a gap in visibility for an organization that an adversary could abuse.

On multiple occasions dating back to 2019, the CrowdStrike Services Incident Response team, CrowdStrike Intelligence team and Falcon OverWatch™ team have encountered an adversary targeting global entities, in particular telecommunications companies, to obtain targeted personal user information — for example, call detail records (CDRs) or information relating to specific phone numbers.

Similar to the activity cluster reported as LightBasin, this adversary primarily focuses on Linux and Solaris systems using a custom-built implant tracked by CrowdStrike Intelligence as JustForFun (also publicly known as BPFDoor). While this adversary does interact with Windows systems, mostly during the early stages of an intrusion, CrowdStrike has not yet identified any custom implants geared toward Windows systems. Instead, the adversary relies on publicly available tools, such as ldapdomaindump, or the post-exploitation framework Impacket, to target Windows systems from previously compromised Linux systems.

CrowdStrike Intelligence is currently tracking these intrusions under the DecisiveArchitect activity cluster (also publicly known as Red Menshen); however, this activity is not currently attributed by CrowdStrike to a specific country-nexus. While CrowdStrike has primarily observed the adversary targeting telecommunications companies, other isolated incidents targeting organizations such as logistics entities have also been observed.

DecisiveArchitect exhibits a high degree of operational security as part of their tactics to make it more difficult for defenders to identify and investigate their activity through the use of various defense evasion techniques. While other publicly available research highlights how the implant operates, this blog focuses on methods to hunt for this implant, or implants that may operate in a similar manner, while also highlighting techniques of interest across Solaris systems.

Spoofed Command Lines

DecisiveArchitect utilizes a custom implant tracked by CrowdStrike as JustForFun, which is typically persisted using SysVinit scripts. When executed, the implant overwrites the process command line within the process environment by randomly selecting a new command line from one of ten hard-coded options, listed in Figure 1.

```
/sbin/udevd -d
/sbin/mingetty /dev/tty6
/usr/sbin/console-kit-daemon --no-daemon
hald-addon-acpi: listening on acpi kernel interface /proc/acpi/event
dbus-daemon --system
hald-runner
pickup -l -t fifo -u
avahi-daemon: chroot helper
/sbin/auditd -n
/usr/lib/systemd/systemd-journald
```

Figure 1. Hard-coded options for command-line spoofing in JustForFun

When DecisiveArchitect interacts with the implants to establish an interactive shell on a system, the `bash` process spawned by the implant process displays the following command line instead. This makes it appear as if the Postfix queue manager is executing as a way to hide itself from analysts and system administrators:

```
qmgr -l -t fifo -u
```

On Solaris systems, though the executable itself exhibits no mechanism for similar command-line spoofing, DecisiveArchitect achieves similar functionality through the use of `LD_PRELOAD`, such as the following example identified within a SysVinit script:

```
LD_PRELOAD=/lib/librbtinfo.so.1 /usr/lib/vtdaemon -c 16
```

When executed, the process only shows the command line `/usr/lib/vtdaemon -c 16`, where the actual JustForFun implant is the file `/lib/librbtinfo.so.1`.

As recently as April 2022, CrowdStrike observed further variations with regard to DecisiveArchitect's tactics, techniques and procedures (TTPs), with the actor using the `LD_PRELOAD` environment variable across Linux systems as well, loading the JustForFun implant, `/lib64/libcaac.so.1`, within the legitimate process `/sbin/agetty`. This highlights a deviation from the standard list of spoofed command lines in Figure 1, likely as part of a further effort to remain undetected and emphasizing the importance of behavioral-based hunting and detection methods.

The spoofed command line appears in commands such as `ps` that may be used to investigate suspicious activity on the host. The spoofed command line makes it less likely that the process will be treated as suspicious.

Solaris Privilege Escalation Vulnerability Exploitation

DecisiveArchitect targets Solaris systems via publicly available exploit code for CVE-2019-3010, a vulnerability in `xscreensaver`. Binaries used to exploit this vulnerability have usually been observed within a few minutes of the JustForFun implant deployment. CVE-2019-3010 is a logic bug that utilizes the `LD_PRELOAD` technique to facilitate local privilege escalation to the root user on Solaris 11 systems. Proof-of-concept (POC) code is publicly available and was not modified by DecisiveArchitect.[1] Table 1 lists two files observed across Solaris systems related to this privilege escalation activity.

File Path	Purpose
/tmp/getuid.so	CVE-2019-3010 exploit binary
/usr/lib/secure/getuid.so	CVE-2019-3010 exploit binary or log file

Table 1. Solaris exploitation file details

Persistence

The way in which DecisiveArchitect achieves persistence across Linux systems involves the usage of SysVinit scripts (i.e., rc.d/init.d scripts). Instead of simply creating a new script that references the JustForFun implants, DecisiveArchitect uses a more operational security-conscious approach by modifying existing SysVinit scripts to reference a small script file, which then finally references the JustForFun implant. The following highlights an example, including the lines added to the legitimate SysVinit script `/etc/rc.d/init.d/pcscd` and the script `/etc/sysconfig/pcscd` referencing the JustForFun implant, `/etc/sysconfig/pcscd.conf`:

`/etc/rc.d/init.d/pcscd:`

- **Line 41:** `if [ -f /etc/sysconfig/pcscd ] ; then`
- **Line 42:** `/etc/sysconfig/pcscd`

`/etc/sysconfig/pcscd:`

```
# Source config
if [ -f /etc/sysconfig/pcscd.conf ] ; then
    /etc/sysconfig/pcscd.conf
fi
```

With this method of nested persistence, if an analyst simply reviews a set of SysVinit scripts by themselves, identifying the malicious line associated with the JustForFun implant would likely prove difficult without subsequently reviewing all of the files referenced within the scripts. Additionally, as part of DecisiveArchitect's continued commitment to operational security, the adversary modifies different legitimate SysVinit scripts across systems, and uses different file names/paths for the implant and associated persistence-related scripts, making it difficult to search across other systems for indicators identified through analysis of a single system.

Detection and Hunting

One of the best ways to detect or hunt for this activity is to utilize EDR technology across supported Linux systems, with machine learning capabilities to detect and prevent the malicious implants, and with hunting capabilities to identify anomalous usage of common Linux system administration utilities or processes running with spoofed command lines (such as the `bash` process running with the command line showing the Postfix queue manager command line instead). Figures 2 and 3 highlight the CrowdStrike Falcon® platform's machine learning capabilities and Falcon OverWatch detections associated with the JustForFun implant and JustForFun command-line tool.

Please note: The examples in the following scenario have CrowdStrike Falcon configured with DETECTIONS ONLY and PREVENTIONS off. A properly configured Falcon instance, as noted previously, would prevent the activity presented here.

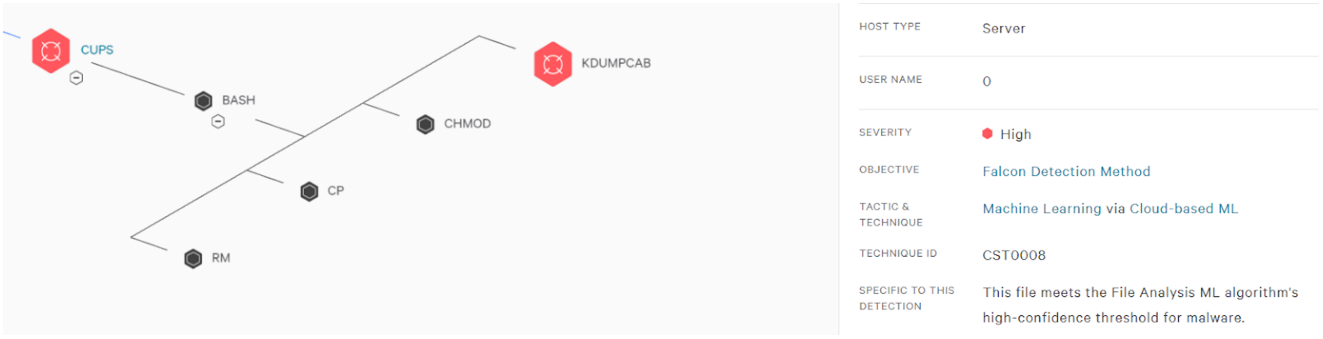


Figure 2. JustForFun implant detection (Click to enlarge)

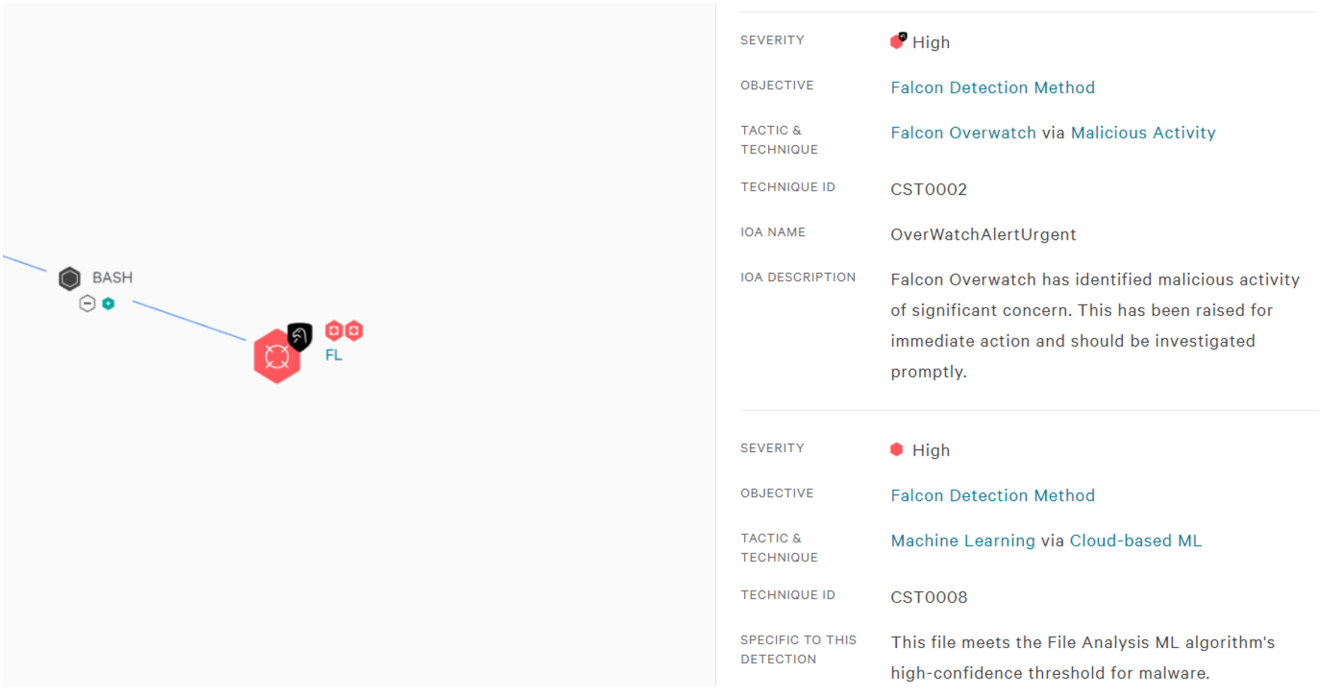


Figure 3. JustForFun command-line tool detections (Click to enlarge)

But, even if an organization has a significant number of legacy or proprietary systems, already has the adversary burrowed into their network, or simply does not have EDR software deployed across Linux systems, all is not lost.

Hunting for Traffic Signaling Implants

Given the fact the JustForFun implant opens a raw socket in order to wait for the magic packet, the built-in Linux utility `lsof` can be used to identify running processes with a raw socket open:

```
lsof -RPnl | grep SOCK_RAW | grep IP
```

Even though this command alone cannot solely determine whether the implant is present — as there are legitimate reasons for processes to have raw sockets open — analysts can highlight processes of interest for further investigation. Of particular importance is that DecisiveArchitect's use of spoofed command lines means that the `lsof` command will report the spoofed command line, as opposed to the actual malicious file, which may make it

difficult to determine whether the process is malicious or not through this alone. However, by running the `lsof` command against the process ID without any of the `grep` filtering, an analyst can list any open files associated with that process, which should reveal the binary. An example of a true positive can be seen below, with the start of the line (i.e., the command line) displaying the start of one of the spoofed command lines listed in Figure 1. It should be noted that DecisiveArchitect can quite easily change these spoofed command lines, so analysts should be conscious of other processes beyond those listed:

```
dbus-daem 1215 0 root 3u pack 11912 0t0 IP type=SOCK_RAW
```

While the `lsof` command is not part of a default Solaris installation, similar commands exist for obtaining additional details from processes:

```
for _PIDno in /proc/*; do line=$(pfiles "${_PIDno}"); echo $_PIDno $line | grep bpf;
done
for _PIDno in /proc/*; do line=$(pmap "${_PIDno}"); echo $_PIDno $line | grep
libpcap; done
for _PIDno in /proc/*; do line=$(pldd "${_PIDno}"); echo $_PIDno $line | grep
libpcap; done
```

The three commands above loop through every process, with the first command looking for a string indicative of a process running with a packet filter, and the other two searching for processes with the `libpcap` library loaded. As with the `lsof` command, this alone cannot solely determine whether the implant is present, so an analyst would need to further investigate the specific process to confirm the presence of the implant. DecisiveArchitect's capability to spoof command lines across Solaris systems also needs to be taken into account when investigating these processes.

When investigating any of these entries, one of the key questions to ask is whether the process in question has any reason to have a raw socket open, to be using a packet filter or to be utilizing the `libpcap` library. One of the most common false positives relates to systems running processes such as `tcpdump` or other packet capture utilities.

While these hunting techniques provide a relatively simple method for identifying DecisiveArchitect activity based on activity observed across multiple intrusions, CrowdStrike expects that DecisiveArchitect will continue development of their implant across both Linux and Solaris platforms, while also improving their techniques regarding operational security of their intrusions to further hinder the ability of a defender to identify or investigate their activity, which might include identifying ways to combat these hunting techniques.

Conclusion

DecisiveArchitect's operations present a clear and present threat to telecommunications companies, as well as other organizations such as logistics entities. This blog highlights important details about DecisiveArchitect's implant, their abilities to operate on Solaris

systems, and ways to hunt down the adversary's implants to help organizations identify whether they have fallen victim to this campaign.

Endnotes

1. https[:]//github[.]com/0xdea/exploits/blob/master/solaris/raptor_xscreensaver

Indicators of Compromise (IOCs)

Indicator	Platform	Purpose

`/run/lock/kdumpflush`
`/run/lock/kdumpcab`
`/var/lock/kdumpcab`
`/var/lock/kdumpcache`
`/dev/shm/kdmtmpflush`
`/dev/shm/kdevtmpfls`
`/dev/shm/ff`
	Linux	JustForFun implant pathnames (temporary – running process)

`/etc/avahi/avahi.conf`
`/etc/cups/cups`
`/etc/cups/cups.conf`
`/etc/gofer/gofor.conf`
`/etc/qss/qss.conf`
`/etc/jvm/jvm.conf`
`/etc/ntp/ntpd`
`/etc/opt/opt.conf`
`/etc/pm/pm.conf`
`/etc/pulp/agent.conf`
`/etc/ssl/ssl.conf`
`/etc/sysconfig/kdumplog`
`/etc/sysconfig/nfs.conf`
`/etc/sysconfig/pcscd.conf`
`/etc/xdg/xdg.conf`
`/usr/java/jdk1.8.0_181-amd64/.java/init.d/jexecd`
`/usr/local/mysql/bin/myisambug`
`/lib64/libcaac.so.1`
	Linux	JustForFun implant pathnames (persistent – on disk)

`/lib/librbtinfo.so.1`
`/usr/lib/autofs/mountd`
`/opt/VRTSvcs/bin/IP/online_Agent`
	Solaris	JustForFun implant pathnames

`/run/lock/lvv`
`/run/lock/lvm/lv`
`/var/run/lvm/vm`
`/dev/shm/sem`
	Linux	JustForFun CLI utility pathnames

`/tmp/getuid.so` `/usr/lib/secure/getuid.so`	Solaris	CVE-2019-3010 exploitation-related files (not unique to DecisiveArchitect)

`/usr/local/bin/GetADUsers.py`
`/usr/local/bin/GetNPUsers.py`
`/usr/local/bin/GetUserSPNs.py`
`/usr/local/bin/atexec.py`
`/usr/local/bin/dcomexec.py`
`/usr/local/bin/dpapi.py`
`/usr/local/bin/esentutl.py`
`/usr/local/bin/getArch.py`
`/usr/local/bin/getPac.py`
`/usr/local/bin/getST.py`
`/usr/local/bin/getTGT.py`
`/usr/local/bin/goldenPac.py`
`/usr/local/bin/ifmap.py`
`/usr/local/bin/karmaSMB.py`
`/usr/local/bin/lookupsid.py`
`/usr/local/bin/mimikatz.py`
`/usr/local/bin/mqtt_check.py`
`/usr/local/bin/mssqlclient.py`
`/usr/local/bin/mssqlinstance.py`
`/usr/local/bin/netview.py`
`/usr/local/bin/nmapAnswerMachine.py`
`/usr/local/bin/ntfs-read.py`
`/usr/local/bin/ntlmrelayx.py`
`/usr/local/bin/opdump.py`
`/usr/local/bin/ping.py`
`/usr/local/bin/ping6.py`
`/usr/local/bin/psexec.py`
`/usr/local/bin/raiseChild.py`
`/usr/local/bin/rdp_check.py`
`/usr/local/bin/reg.py`
`/usr/local/bin/registry-read.py`
`/usr/local/bin/rpcdump.py`
`/usr/local/bin/sambaPipe.py`
`/usr/local/bin/samrdump.py`
`/usr/local/bin/secretsdump.py`
`/usr/local/bin/services.py`
`/usr/local/bin/smbclient.py`
`/usr/local/bin/smbexec.py`
`/usr/local/bin/smbrelayx.py`
`/usr/local/bin/smbserver.py`
`/usr/local/bin/sniff.py`
`/usr/local/bin/sniffer.py`
`/usr/local/bin/split.py`
`/usr/local/bin/ticketer.py`
`/usr/local/bin/wmiexec.py`
`/usr/local/bin/wmipersist.py`
`/usr/local/bin/wmiquery.py`
	Linux	Impacket post-exploitation framework scripts (not unique to DecisiveArchitect)

`/usr/local/bin/ldapdomaindump` `/usr/local/bin/ldd2bloodhound`	Linux	ldapdomaindump binaries used for Active Directory reconnaissance (not unique to DecisiveArchitect)
`c:\users\use.bat` `c:\users\one.ps1`	Windows	Unknown scripts

Additional Resources

- *Read about another threat that targets the telecommunications sector in this blog: LightBasin: A Roaming Threat to Telecommunications Companies.*
- *Download the CrowdStrike 2022 Global Threat Report for insights into adversaries tracked by CrowdStrike Intelligence in 2020.*
- *Get a full-featured free trial of CrowdStrike Falcon Prevent™ and learn how true next-gen AV performs against today's most sophisticated threats.*