Cobalt Strike

Despite a rise in alternatives, Cobalt Strike remains a popular command and control (C2) framework among adversaries, particularly ransomware operators.

PAIRS WITH THIS SONG

#8

OVERALL RANK

3.0%

CUSTOMERS AFFECTED

ANALYSIS

***Editor's note**: While the detection opportunities and analysis on this page are still relevant, it has not been updated since 2023.*

Analysis

Cobalt Strike continues to be a favorite post-exploitation tool for adversaries. At #8, it is the only post-exploitation framework to make the top 10. **Ransomware** operators in particular rely substantially on Cobalt Strike's core functionalities as they seek to deepen their foothold in their victims' environments. Its speed, flexibility, and advanced features are likely contributing factors as to why ransomware attacks have been ticking upward in recent years. Some of the most notorious ransomware operators— including groups like **Lockbit** and **Royal**—are known to rely heavily on Cobalt Strike in their attacks.

Striking developments

Cobalt Strike developers made **multiple changes** throughout 2022, including even more flexible C2 profiles, SOCKS5 proxy support, and injection options. These improvements allow adversaries to further customize their TTPs, making detection challenging. While those additions benefitted adversaries, the developers of Cobalt Strike also imposed major changes to discourage the cracking and abuse of Cobalt Strike packages. Notably, the developers changed how they distributed Cobalt Strike's team server component, resulting in better product security. That said, we often observe Cobalt Strike beacons from older versions of the software, indicating that some criminal adversaries take advantage of older cracked or pirated versions over the newer ones.

TAKE ACTION

The security community is embracing the fact that whatever functional label you place on Cobalt Strike, it's here to stay, it's implicated in all variety of intrusions, and it's our duty to defend against it. Luckily for defenders, the security community has produced a plethora of great technical analysis and detection opportunities around preventing and investigating Cobalt Strike. For defenders getting started with understanding how the tool works and operates, we highly recommend reading each of the following

- **Defining Cobalt Strike Components & BEACON**
- **New Snort, ClamAV coverage strikes back against Cobalt Strike**
- **Cobalt Strike, a Defender's Guide – Part 1**
- **Cobalt Strike, a Defender's Guide – Part 2**
- **Full-Spectrum Cobalt Strike Detection**

Hunting team servers

There are several strategies to hunt proactively for Cobalt Strike team servers in the wild, mostly based around network data and service fingerprinting. These strategies include using tools such as Shodan and Censys to find servers using default TLS certificate values, default team server ports (50050), and default JARM hashes associated with Cobalt Strike. While many adversaries change these default values, we still often find adversaries that don't change them, resulting in simpler identification. For more details on proactively identifying Cobalt Strike infrastructure, check out these resources:

- **Hunting Cobalt Strike C2 with Shodan by Michael Koczwara**
- **Cobalt Strike Analysis and Tutorial: Identifying Beacon Team Servers in the Wild**

Detection opportunities

Cobalt Strike beacon implant

named pipes, utilizing a host of default names commonly used by adversaries. Analysis should focus on any file modifications to a suspicious named pipe within this process.

```
file_modifications_include
('pipe\msagent_' ||
'pipe\interprocess_' ||
'pipe\lsarpc_' || 'pipe\samr_'
|| 'pipe\netlogon_' ||
'pipe\wkssvc_' || 'pipe\srvsvc_'
|| 'pipe\mojo_' || 'pipe\postex'
|| 'pipe\status_' || 'pipe\msse-
')
```

`rundll32.exe` to spawn SQL Server Client Configuration Utility

This analytic identifies instances of `rundll32.exe` spawning the SQL Server Client Configuration Utility (`cliconfg.exe`). We often see this pattern of process execution when Cobalt Strike leverages **DLL Search Order Hijacking** as a method of UAC bypass.

```
parent_process == rundll32.exe
&&
```

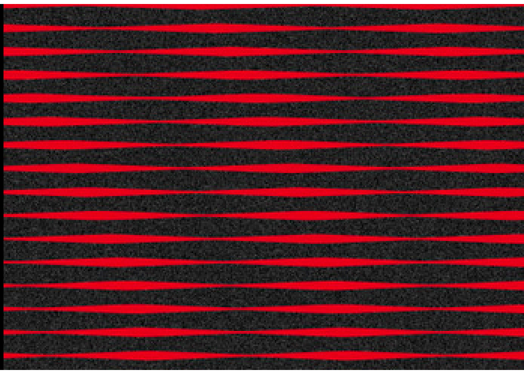
Command-line patterns for Cobalt Strike beacons via GetSystem

This analytic identifies commonly observed command-line patterns when Cobalt Strike beacons escalate privileges via the **GetSystem feature**. Adversaries use **GetSystem** to impersonate a token for the SYSTEM account. This level of access allows an adversary to perform privileged actions beyond that of an administrator.

```
process == cmd.exe
&&
command_includes ('/(?i)echo\s+
[0-9a-f]{11}\s+\>\;?
\s+\\\\\.\\pipe\\[0-9a-f]
{6}/.match')*
```

**NOTE: The above regular expression will match on the following example what of using GetSystem may look like via a Cobalt Strike beacon:* `C:\Windows\system32\cmd.exe /c echo 92d8cc45954 >; \\.\pipe\446b3c`

See Red



red canary

in action

— Schedule your demo now

Search

PRODUCTS

Managed Detection and Response (MDR)
Readiness Exercises
Linux EDR
Atomic Red Team™
Mac Monitor
What's New?
Plans

SOLUTIONS

Deliver Enterprise Security Across Your IT Environment
Get a 24×7 SOC Instantly
Protect Your Corporate Endpoints and Network
Protect Your Users' Email, Identities, and SaaS Apps
Protect Your Cloud
Protect Critical Production Linux and Kubernetes
Stop Business Email Compromise
Replace Your MSSP or MDR

red canary

Operationalize Your Microsoft Security Stack

Minimize Downtime with After-Hours Support

RESOURCES

View all Resources
Blog
Integrations
Guides & Overviews
Cybersecurity 101
Case Studies
Videos
Webinars
Events
Customer Help Center
Newsletter

PARTNERS

Overview
Incident Response
Insurance & Risk
Managed Service Providers
Solution Providers
Technology Partners
Apply to Become a Partner

COMPANY

About Us
The Red Canary Difference
News & Press
Careers – We’re Hiring!
Contact Us
Trust Center and Security

info@redcanary.com
+1 855-977-0686 Privacy Policy Trust Center and Security Cookie Settings