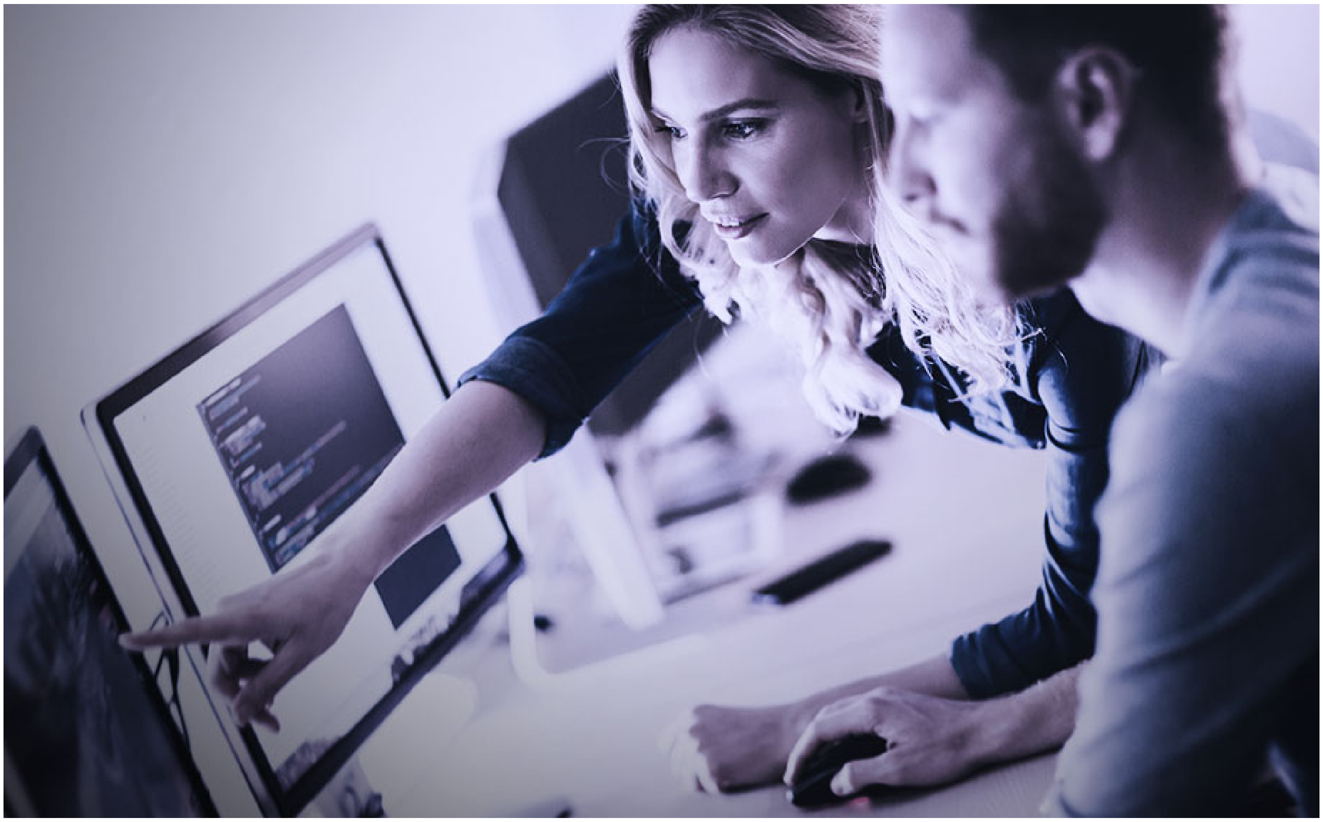
Cobalt Strike – Post-Exploitation Attackers Toolkit

deepinstinct.com/2021/03/18/cobalt-strike-post-exploitation-attackers-toolkit/

March 18, 2021



Learn more

March 18, 2021 | Ben Gross

Introduction to the Framework

Cobalt Strike (CS) is a paid penetration testing toolkit that allows an attacker to deploy a component named ***Beacon*** on a victim's machine. The simplicity, reliability, and versatility of CS make it very popular among threat actors—and there are plenty of cracked versions of CS available on the dark web[1]. Given this reality, it's been used frequently in recent cyber-attacks[2].

CS provides a wealth of functionality to the attacker, including command execution, key logging[3], file transfer, privilege escalation, port scanning, lateral movement, and more. The framework is split into two components: client and server. The server module, aka team server, is the controller of the Beacon payload. By using this module the attacker can track and execute commands on an infected host and utilize all of the framework capabilities.

Cobalt Strike Beacon

The Beacon, which is the main component being used to target accounts, allows its operators to execute commands, log keystrokes, drop files, and communicate with targeted systems. CS is primarily used as a post-exploitation tool; leveraged by attackers after they have a foothold in a system and want to remain hidden.

Deploying a Beacon and making sure its communication will stay hidden from cybersecurity products and teams is a critical task for adversaries. The Beacon has several communication methods[4] to make this happen, including HTTP, HTTPS, DNS, and SMB. By default, the Beacon will reach out to its C2 periodically, sending meta-data back and gathering any commands issued by the operator. The Beacon console allows the attacker to monitor which tasks were issued to a Beacon and track their status, check the output of commands, and find additional information on targets.

How Attackers Use Cobalt Strike

Even though CS is a paid penetration testing product, it is incredibly popular due to its wealth of capabilities and its ability to add new features and modify existing ones. This flexibility allows attackers to implement their own tools, use built-in tools, or integrate other penetration testing tools such as the Metasploit framework and Mimikatz. By design the main use of CS is to act as a post-exploitation tool that allows attackers to gather information, harvest credentials, and deploy other payloads on an infected host. That also means that is not designed to gain initial access to a system, even though it does have components that can help to gain access such as its VBA macros and Windows-executable generators.

CS provides the attacker a wide set of tools; we will cover some of the framework capabilities from an attack-chain point of view. The full list of capabilities is available in the MITRE matrix[5].

Initial Access

- **System Profiler**: A honeypot used as a reconnaissance tool to collect information about a target. It is designed to collect information on systems or users that visit CS-controlled servers and provide a list of applications and plug-ins discovered (it is not designed to infect a host).
- **MS-Macros Generator**: CS can generate VBA code to embed in Office documents.

- **Website Clone Tool**: This tool can create a local copy of a website with some code added to fix links and images so they work as they should. An attacker can lure a victim to enter the cloned website to collect information about the victim's network.
- **Windows Loaders and Payload Generator**: CS can generate a Windows executable, a script (e.g., PowerShell, HTA), or a raw blob of position-independent code that contains a Beacon. CS provides an internal kit for building shellcode and executables. The kit can be easily modified to suit attacker's needs.
- **Phishing**: The CS phishing module helps an attacker replace links and text to build a convincing phish in an email template, which it can send to multiple recipients and track who entered them. This module can be used along with the website clone tool to lure a victim into CS-owned websites.

Privilege Escalation and Lateral Movement

- **Mimikatz:** An open-source tool that allows users to view and save credentials, extract plaintext passwords, hash, PIN codes, and Kerberos tickets from the systems memory. Mimikatz is fully supported in CS. Attackers can run and execute Mimikatz commands directly from the CS command-line interface.
- **User-Account-Contraol (UAC) Bypass[6]**: CS can bypass UAC by utilizing a method called reflective DLL injection.
- **Antimalware Scan Interface (AMSI) Bypass:** CS can bypass AMSI by patching OS functions that limit AMSI's capabilities.

Commands Execution

- **Aggressor Script**: CS has its own scripting language which allows its users to modify and extend the Beacons functionality.
- **Running Commands:** CS uses a command-line interface to interact with infected systems. The commands may run via cmd.exe[7], powershell.exe[8], psinject[9], Powerpick[10], and more.
- **Native API**[11]: Beacon can run shell commands without cmd.exe or powershell.exe by directly calling the OS API functions or by using Powerpick, which is a program that allows the execution of Powershell without the use of Powershell.exe.

Command and Control Communication

- **Web Protocols:** CS uses its own command-and-control communication protocol that can be encapsulated by HTTP/HTTPS/DNS.
- **SMB (Server Message Block):** CS can conduct P2P communication over Windows-named pipes encapsulated in the SMB protocol.

Protection from Cobalt Strike

Deep Instinct prevents the CS framework and its components at all attack stages. The first possible attack vector is loaders. Whether they are Windows executables or Office documents, we prevent them and stop the attack chain at the earliest possible stage by using Deep-Learning based static analysis.

In the event that an attacker has already gained access into a victim's system and is trying to deploy a Beacon, our behavioral capabilities can spot in-memory actions such as DLL injection and shellcode execution and prevent these post-exploitation attempts from running. In addition, our PowerShell Deep Learning-based static analysis and behavioral analysis will prevent all malicious PowerShell activities.

Summary

Cobalt Strike is a paid penetration testing product that is in continual development and its team builds the framework with the most advanced and up-to-date security features and capabilities. Since CS is being used by both security teams and threat actors for the same purposes it poses a serious and ongoing threat for security products, organizations, and individuals.

Using our advanced Deep Learning-based static analysis and behavioral capabilities, customers of Deep Instinct can be rest assured that they have protection against Cobalt Strike and its capabilities as the attack is detected and prevented in a matter of milliseconds.

To see our capabilities for yourself, request a demo via our contact us form.

----

[1] https://www.bleepingcomputer.com/news/security/alleged-source-code-of-cobalt-strike-toolkit-shared-online/

[2] https://www.bleepingcomputer.com/news/security/solarwinds-hackers-used-7-zip-code-to-hide-raindrop-cobalt-strike-loader/

[3] https://attack.mitre.org/techniques/T1056/001

[4] https://attack.mitre.org/techniques/T1071/001/

[5] https://attack.mitre.org/software/S0154/

[6] https://attack.mitre.org/techniques/T1548/002/

[7] https://attack.mitre.org/techniques/T1059/003/

[8] https://attack.mitre.org/techniques/T1059/001

[9] https://github.com/EmpireProject/PSInject

[10] https://github.com/PowerShellEmpire/PowerTools/tree/master/PowerPick

[11] https://attack.mitre.org/techniques/T1106