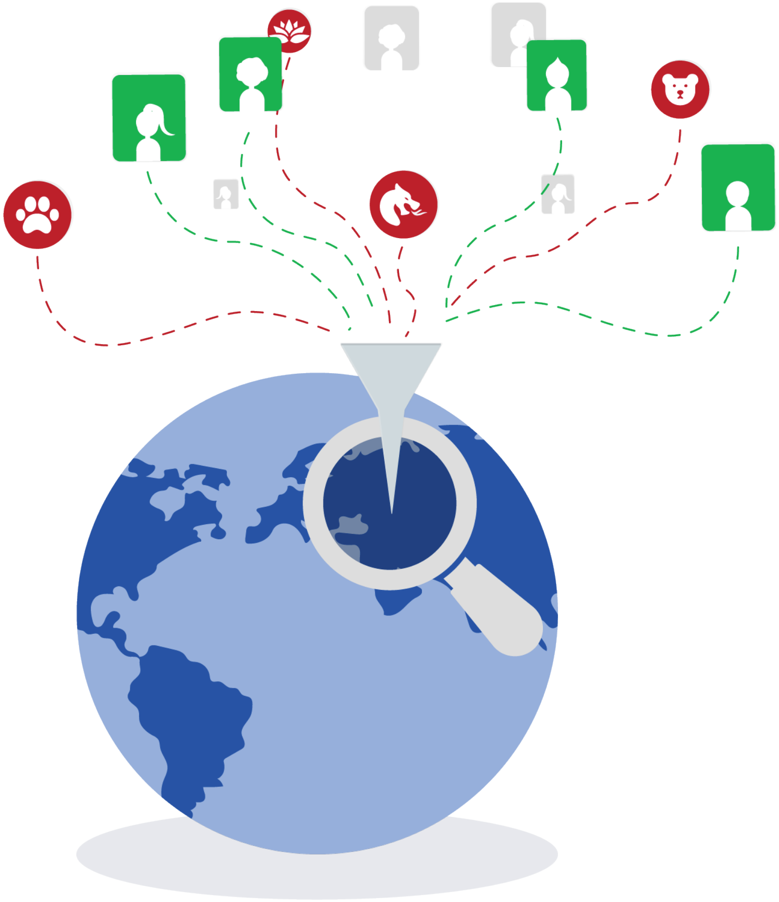
Recorded Future

CYBER THREAT ANALYSIS

A Multi-Method Approach to Identifying Rogue Cobalt Strike Servers

By Insikt Group®



CTA-2019-0618

Recorded Future assessed changes to Cobalt Strike servers in the wild in the aftermath of the public identification of several Cobalt Strike server detection methods. In our analysis, we conducted an evaluation of different methodologies and did a combined analysis based on them to determine if users changed their configurations to avoid detection. Sources include the Recorded Future® Platform, BinaryEdge, Censys, Rapid7 Lab's OpenData, Shodan, GreyNoise, ReversingLabs, VirusTotal, Farsight DNS, and other open sources. This report will be of greatest interest to organizations seeking to improve the speed of their response times, as well as analysts who deal with Cobalt Strike incidents on a regular basis..

Executive Summary

Cobalt Strike is an exploitation platform developed for the use of security professionals in emulating targeted attacks and post-exploitation actions by advanced adversaries. The tool, developed and licensed by Strategic Cyber LLC, a company based in Washington, D.C., is monitored for illicit usage by the firm and is subject to export controls. Despite this, the Cobalt Strike framework has become a popular option among the various software of this type, which includes other paid suites like Metasploit Pro, Core Impact, and others. Although not alone among such platforms in being used by unlicensed users and criminal actors, Cobalt Strike has been used by a variety of threat groups, including APT32, who have used the tool for initial exploitation, and the namesake Cobalt Group, which has heavily relied on the framework.

Considering the significant use of the Cobalt Strike platform by security testers — and, more importantly, malicious attackers — the necessity of recognizing Cobalt Strike server connections to corporate network assets is evident.

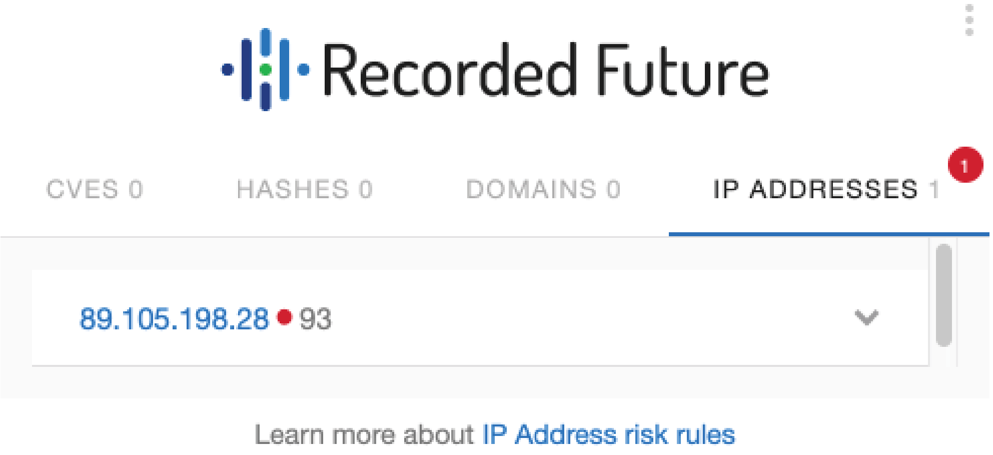
Despite the detection methodology being public, Recorded Future has observed that Cobalt Strike servers have been left largely unpatched, allowing fingerprinting and subsequent detection. This methodology, coupled with other detections, allowed Recorded Future to sample Cobalt Strike servers found in the wild, and compare fingerprinting methods to help defenders best track and monitor this framework. The tracking of Cobalt Strike servers can aid blue teams in detecting red team activity and containing activity from adversaries who have not modified their Cobalt Strike Team Server.

Key Judgments

- Cobalt Strike servers remain fairly exposed and relatively easy to detect, despite patching to make specific fingerprinting methods more difficult. Many Cobalt Strike servers operating before the patch was released have not updated their systems, while newer deployments have used the upgraded software.
- Recently deployed Cobalt Strike servers are more likely to deploy an updated Cobalt Strike version (beyond 3.12) while continuing to use the default TLS certificate, which remains a reliable detection mechanism.
- Recorded Future's sampling of current Cobalt Strike servers, contrasted with historic threat activity, found that criminal and state-aligned actors alike have used default, unpatched Cobalt Strike configurations, perhaps in an effort to blend in with other Cobalt Strike servers, or possibly simply because the default settings work well without alteration and the operator does not feel the need to alter anything.
- The detection of Cobalt Strike servers can aid defenders in creating alerts in their enterprise networks, providing a proactive measure to get ahead of their red team, criminal operations, or state-sponsored adversaries.

Background

A primary issue for incident response and security operations analysts today is determining which security events or alerts are a priority to review. Fortunately, applying accurate threat intelligence to a SIEM workflow, such as Splunk, can be valuable for identifying credible threats, and can even reveal crucial additional context to enable security teams to take more proactive measures. For example, an alert comes across your SIEM — it's an IP address, 89.105.198[.]28, that has been contacted by one of your endpoints. Now what?



89.105.198.28 ● 93

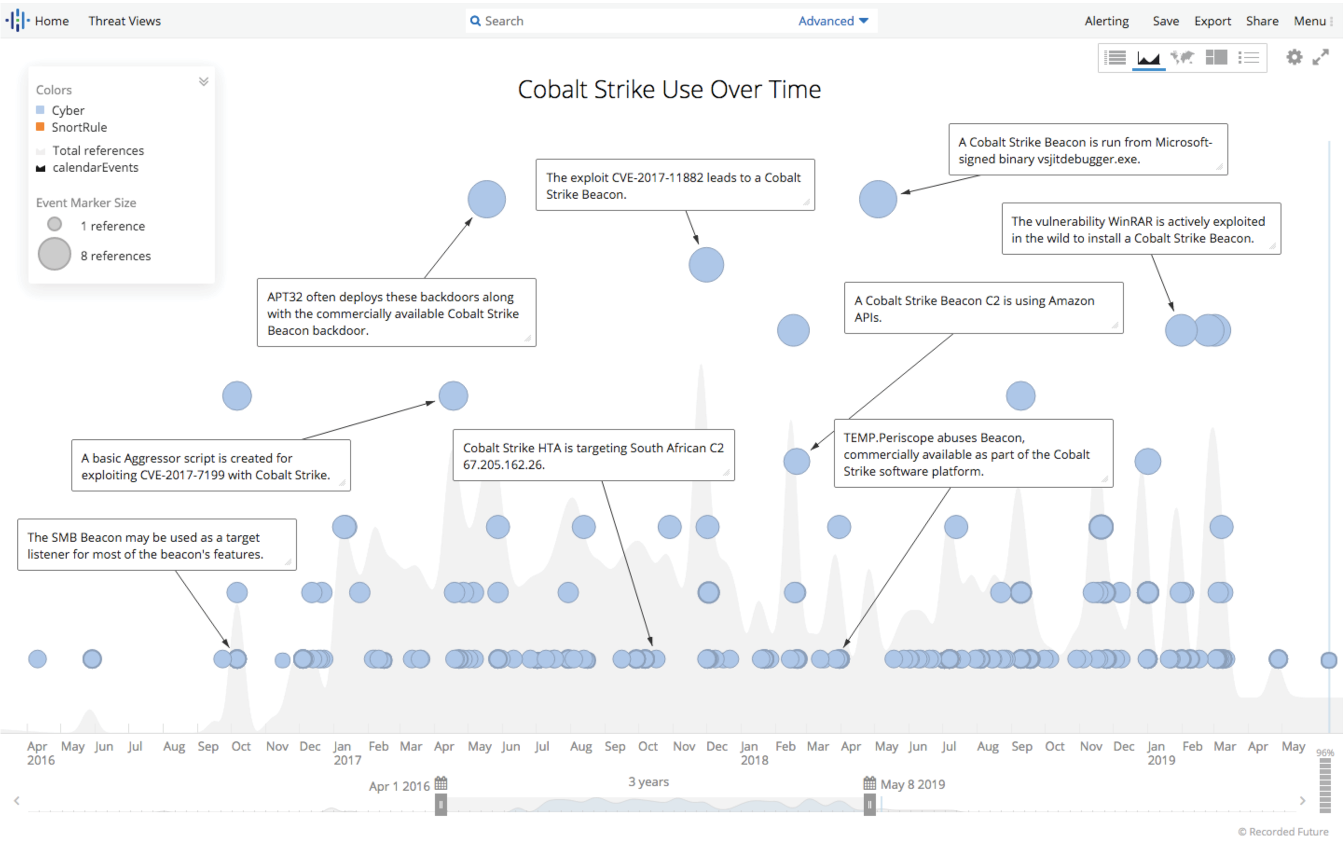
Learn more about IP Address risk rules

Recorded Future browser extension.

Upon opening the Recorded Future browser extension on the Splunk alerts page, the IP 89.105.198[.]28 jumps to the top with a risk score of 93 (this finding was made on May 6, 2019, and the risk score will decay on May 17, 2019 if no further malicious activity is observed). This investigation reveals that the IP address was previously reported by Sophos as part of the MegaCortex ransomware campaign, using a Cobalt Strike reverse shell.

Cobalt Strike is an adversary simulation platform developed for penetration testers by Raphael Mudge, founder of Strategic Cyber LLC. Designed for interoperability with other platforms such as Metasploit, NMAP, and Powershell Empire, it can be run using Armitage, a graphic user interface (GUI) developed by Mudge, initially for Metasploit. Armitage and Cobalt Strike are designed around a team server that allows for the sharing of information and the ability to direct and execute well-coordinated actions.

Known for its advanced functionality, Cobalt Strike has been adopted by numerous security professionals and also used illicitly by criminal and nation-state entities. As MITRE has stated, “Cobalt Strike’s interactive post-exploit capabilities cover the full range of ATT&CK tactics, all executed within a single, integrated system.” The framework has been a mainstay in the threat landscape in the last three years, frequently used by criminal groups, state-sponsored actors, and, of course, penetration testing teams.



A Cobalt Strike Beacon is run from Microsoft-signed binary vsjitdebugger.exe.

The exploit CVE-2017-11882 leads to a Cobalt Strike Beacon.

The vulnerability WinRAR is actively exploited in the wild to install a Cobalt Strike Beacon.

APT32 often deploys these backdoors along with the commercially available Cobalt Strike Beacon backdoor.

A Cobalt Strike Beacon C2 is using Amazon APIs.

A basic Aggressor script is created for exploiting CVE-2017-7199 with Cobalt Strike.

Cobalt Strike HTA is targeting South African C2 67.205.162.26.

TEMP.Periscope abuses Beacon, commercially available as part of the Cobalt Strike software platform.

The SMB Beacon may be used as a target listener for most of the beacon's features.

Cobalt Strike use over time. (Source: Recorded Future)

Cobalt Strike is professionally maintained and available under license currently for a $3,500 USD fee with annual renewals. In addition to export controls set by the United States, Strategic Cyber LLC attempts to strictly control its licensing and use to legitimate security professionals and keep the software out of the hands of malicious actors, making it difficult for both criminals and entities outside the United States to acquire it.

Strategic Cyber LLC regularly updates and patches licensed versions of the software. Recent changes to Cobalt Strike's server configurations attempt to help the framework evade detection. Pirated versions of the software, however, will not receive official updates and patches.

Although the software licensing has been strictly controlled, there are confirmed instances of pirated versions of Cobalt Strike in the wild, often cracked trial versions, and a variety of actors in the criminal underground have been observed attempting to acquire or trade them. The cracked versions, however, may come with their own added "features" such as backdoors, or be lacking in some way. One member on Raid Forums posted a link to a cracked copy of Cobalt Strike 3.13 (the latest version) on April 5, 2019, but other members pointed out that it was missing some features, and parts of the software which should have been removed, such as EICAR, remained. Legitimate versions of Cobalt Strike are therefore valuable; for example, one Maza forum member was observed last year offering $25,000 USD for a purchaser within the U.S. to obtain a licensed copy of Cobalt Strike and illegally transfer it to this forum member.

Returning to our investigation of 89.105.198[.]28, this IP address was used as a command-and-control server for a Cobalt Strike reverse shell on a victim domain controller during a MegaCortex incident. The ransomware was then distributed across the environment via PSExec. The MegaCortex ransomware campaign was active at the time of this analysis. Further investigation of the IP address reveals that it makes use of the Cobalt Strike server default security certificate to encrypt traffic.

This case involving 89.105.198[.]28 prompted Recorded Future to investigate this specific Cobalt Strike activity. This further encouraged larger-scale Cobalt Strike research, in the wake of security firm Fox-IT's findings around the anomalous space included in Cobalt Strike HTTP responses and other public detections, including common use of the standard, pre-configured, self-signed SSL/TLS certificate on Cobalt Strike servers. Servers that deploy this certificate can be detected via Shodan or Censys by the SHA256 hash or the serial number of the certificate.

```
Subject DN: C=, ST=, L=, O=, OU=, CN=
Issuer DN: C=, ST=, L=, O=, OU=, CN=
Serial: 146473198
SHA-256 Fingerprint:
87f2085c32b6a2cc709b365f55873e207a9caa10bffecf2fd16d3cf9d94d390c
Validity: 2015-05-20 18:26:24 to 2025-05-17 18:26:24 (3650 days, 0:00:00)
```

Default Cobalt Strike SSL/TLS certificate.

Public Methodologies for Identifying Cobalt Strike Team Servers

On February 19, 2019, Strategic Cyber LLC (the producer of Cobalt Strike) released the results of a "Cobalt Strike Team Server Population Study." The study was undertaken in part to discover the license status of discovered Cobalt Strike software, as well as identify and analyze any significant alterations made to versions of the software currently in use.

This study identified multiple methods that could be used to identify Cobalt Strike servers in the wild:

- Cobalt Strike servers are shipped with a default security certificate which can be used to fingerprint them unless the administrator changes it.

  SHA256:
  87f2085c32b6a2cc709b365f55873e207a9caa10bffecf2fd1
  6d3cf9d94d390c
  Serial Number: 146473198

- When enabled, the Cobalt Strike DNS server responds to any DNS request received with a bogon (fake) IP: 0.0.0.0 (this is not unique to Cobalt Strike servers).
- The default controller port for Cobalt Strike Team Server is 50050/TCP, a port unlikely to be found open on other servers.
- The "404 Not Found" HTTP response for Cobalt Strike is unique to NanoHTTPD web servers and can be detected.

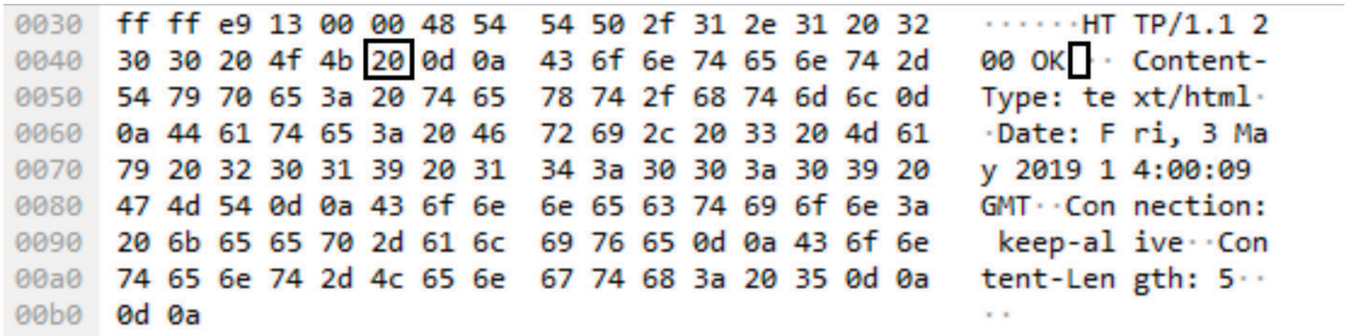
Taken as a whole, the surest method in the list above is fingerprinting Cobalt Strike servers using the default security certificate. The remaining detection methods are less certain and all will be of higher confidence when corroborated with other methodologies. For example, any server using port 50050 that also provides an HTTP response unique to NanoHTTPD web servers is more likely a Cobalt Strike server than one found to only exhibit an HTTP response signature.

NanoHTTPD is an open source web server framework. NanoHTTPD servers and Cobalt Strike servers running version 3.12 and earlier could be identified via a null space in the HTTP response where “HTTP/1.1” is followed by a blank space (0x20) not found in other web server responses. Any HTTP response from a pre-3.13 Cobalt Strike server will contain this null space, and a scanner that can retrieve HTTP server responses may be used to search for them. A simple manual method of identifying the aforementioned null space may be done with a packet capture of a browser HTTP connection to a Cobalt Strike server, in which the extra space can be easily seen.

As Cobalt Strike instances running cracked versions are not updated or patched, this method provides the added potential of discovering Cobalt Strike servers operated by criminals.

Not specifically mentioned in the Strategic Cyber LLC blog post is another method of identifying Cobalt Strike servers. On January 2, 2019, Cobalt Strike version 3.13 was released. The Cobalt Strike release notes state that one of the changes from previous versions was the removal of an “extraneous space from HTTP status responses.” An extra null byte in the HTTP server response of NanoHTTPD servers (an open source, Java-based web server) affected the Cobalt Strike Team Server, which was first released in 2012 and is based upon NanoHTTPD.

The research on Cobalt Strike servers published by security firm Fox-IT on February 26, 2019, provided not only details on how to identify the servers prior to version 3.13 (which respond with the additional null space in the HTTP response), but also a list of over seven thousand IPs hosting Cobalt Strike servers observed from 2015 to 2019 using this detection method found in publicly available data from Rapid7.



```
0030  ff ff e9 13 00 00 48 54  54 50 2f 31 2e 31 20 32   ······HT TP/1.1 2
0040  30 30 20 4f 4b 20 0d 0a  43 6f 6e 74 65 6e 74 2d   00 OK ·· Content-
0050  54 79 70 65 3a 20 74 65  78 74 2f 68 74 6d 6c 0d   Type: te xt/html·
0060  0a 44 61 74 65 3a 20 46  72 69 2c 20 33 20 4d 61   ·Date: F ri, 3 Ma
0070  79 20 32 30 31 39 20 31  34 3a 30 30 3a 30 39 20   y 2019 1 4:00:09 
0080  47 4d 54 0d 0a 43 6f 6e  6e 65 63 74 69 6f 6e 3a   GMT··Con nection:
0090  20 6b 65 65 70 2d 61 6c  69 76 65 0d 0a 43 6f 6e    keep-al ive··Con
00a0  74 65 6e 74 2d 4c 65 6e  67 74 68 3a 20 35 0d 0a   tent-Len gth: 5··
00b0  0d 0a                                              ··
```

Packet capture showing extra null space in the HTTP header from a Cobalt Strike server.

Similarly, on February 27, 2019, the Chinese Knownsec security research team published a blog detailing their use of the NanoHTTPD 404 Not Found response anomaly reported by Strategic Cyber LLC, as well as the null space anomaly, to identify Cobalt Strike servers. They found fewer numbers of servers in the data within their associated ZoomEye search engine platform, but still found over three thousand. Knownsec reported that the open source NanoHPPTD code that Cobalt Strike is built on responds in the following manner, precisely:

```
HTTP/1.1 404 Not Found
Content-Type: text/plain
Date: Day, DD Mmm YYYY HH:MM:SS GMT
Content-Length: 0
```

Knownsec based their detection logic on this finding. However, Knownsec also subsequently observed that the order within the HTTP response may in fact differ, after finding “content-type” presented after “date” in the response from some Cobalt Strike systems.

A reliable method for discovering Cobalt Servers is available to those with access to detailed network traffic data. The open source JA3 project, developed by three Salesforce researchers, allows for the detection of suspicious HTTPS traffic by fingerprinting the TLS negotiation between servers and clients. The TLS/SSL version, accepted cipher suites, and elliptic curve details (such as elliptic curve point formats) can be fingerprinted much like a browser can be fingerprinted by its version, add-ons, and other features specific to that one browser.

JA3 signatures are for the client side and JA3S signatures are for servers. In the case of Cobalt Strike, fingerprints have been created for TLS negotiation by the client beacon (which uses the Windows socket to initiate communication) and Cobalt Strike servers running on the Kali Linux operating system. These fingerprints would need to be used together to reliably discover a Cobalt Strike server. Although this detection method can be partly mitigated by the Cobalt Strike operator by using a “redirector,” many Cobalt Strike servers do not use such a proxy.

JA3 and JA3S signatures can be used with tools such as Zeek/Bro and Suricata. Data from these network detection tools can subsequently be fed into a SIEM such as Splunk. JA3 and JA3S signatures are available at Salesforce's Github account and from other sources.

As with detections of other tools such as Metasploit, Powershell, or PsExec that may be used by a security team or administrators, network defenders should exercise due diligence if they find evidence indicating connections from within their network to a Cobalt Strike server, as the detection itself will not identify the intentions of the user. Identifying a Cobalt Strike server as that of an authorized red team or a true adversary may be impossible based on detected traffic alone.

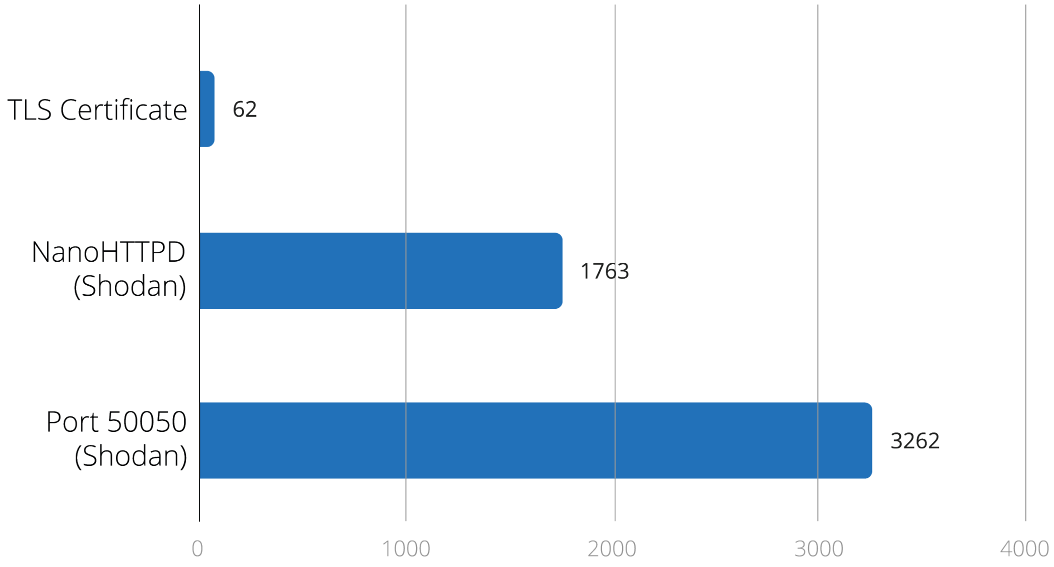
Changes Since Fox-IT and Knownsec Reports Publicizing Anomalous HTTP Responses

We expected the number of Cobalt Strike servers identified by these methods to decrease after the publication of information by Strategic Cyber LLC, Fox-IT, and Knownsec in late February 2019 concerning the detection of Cobalt Strike servers. Additionally, Cobalt Strike operators were encouraged by Strategic Cyber LLC in their February study to make use of an Apache or Nginx web server as a "redirector" to proxy their traffic; this precludes simple detections of Cobalt Strike servers by removing the anomalous HTTP responses, default security certificates, and other such identifiers from the equation. Updating legitimate, licensed servers to version 3.13 would decrease the number found using the extraneous null space method, but Cobalt Strike operators being aware of the well-publicized detection methods would also be expected to decrease the number of detectable servers.

By duplicating Fox-IT's methodology of detecting the anomalous null space in HTTP responses, Insikt Group confirmed a noticeable decrease in identified servers. 388 Cobalt Strike servers were observed for the first time in February 2019 using Rapid7's data. The number of first-seen Cobalt Strike servers using this method was only 90 in April 2019. However, this is only part of the story; older Cobalt Strike servers visible using this method have decreased in number but far less significantly. 441 of the servers observed in Rapid7's data were still observed to be up in April 2019, which is more than the 387 last observed in January 2019.

By analyzing the Knownsec research to identify Cobalt Strike with a different HTTP detection methodology, Insikt Group replicated their research in the same ZoomEye search engine data. Insikt Group identified 1,580 servers that were up in 2018, and only 1,053 through May 2019.

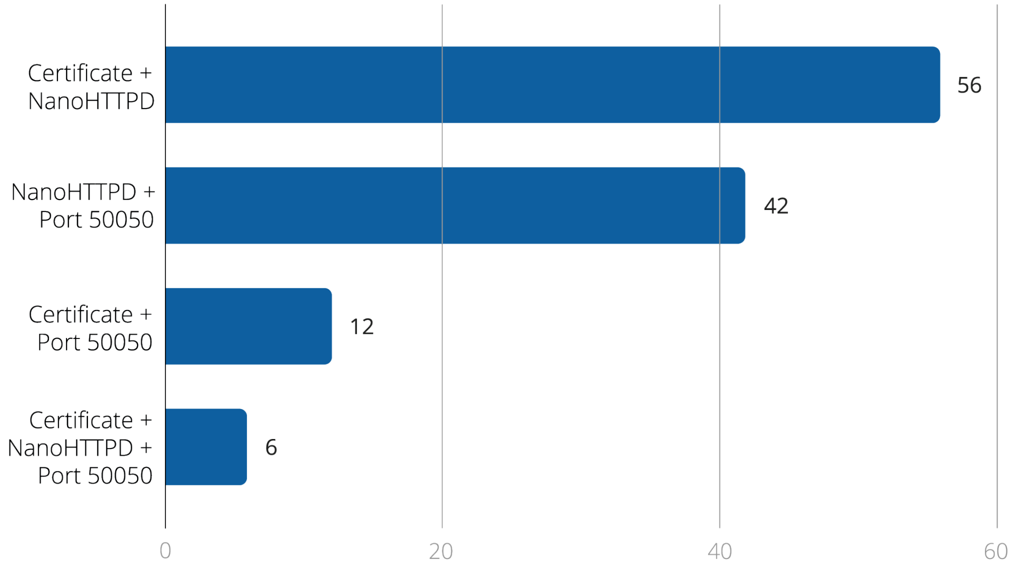
Individual Detections



Recent metrics of individual Cobalt Strike detection parameters. (January 2019 to May 2019)

As previously noted, both of these HTTP detection methods are based upon anomalies within NanoHTTPD, not Cobalt Strike systems in particular. Not all of those detected using these methods had corroborative data, such as open port 50050. Other variables are also involved in the change to the number of servers. Cobalt Strike servers may change IPs and do not always remain up for long periods of time. Although there has been a reduction in newly sighted Cobalt Strike servers since January 2019, the data indicates that there are still a large number of servers in operation that are detected by the HTTP null-space anomaly method.

Cobalt Strike Detection Overlap



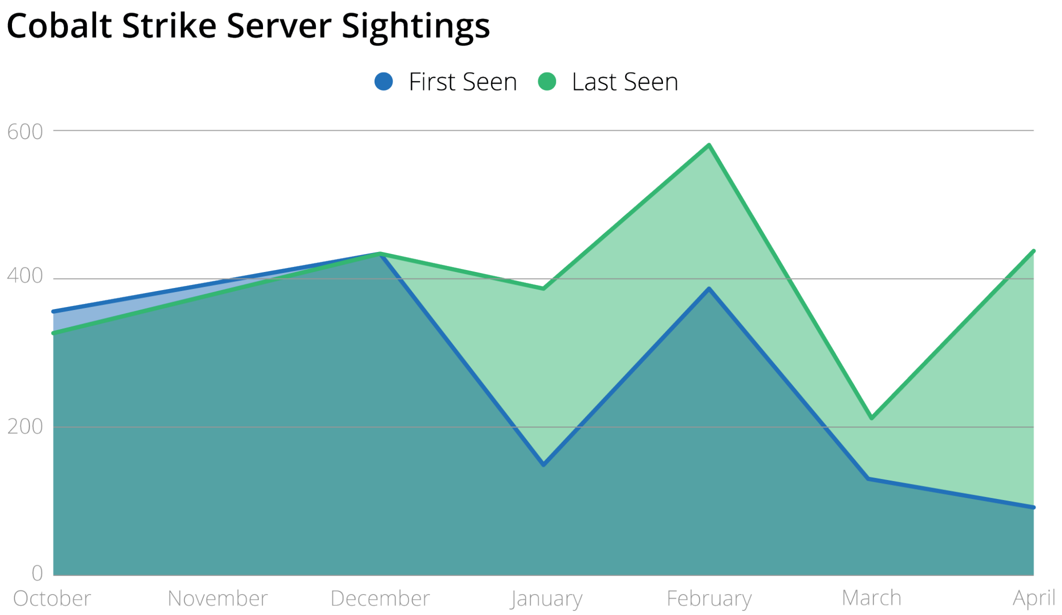
Cobalt Strike identification using combined detection methods. (January 2019 to May 2019)

The combination of the three detections made for high confidence assessments for the servers to be hosting Cobalt Strike — in fact, all six of the servers identified in this manner were previously reported to host Cobalt Strike and communicate with various Cobalt Strike beacons. The use of the default Cobalt Strike proves to be the best detection methodology; however, monitoring the combined usage of NanoHTTPD and open port 50050 can narrow the field of IPs to monitor greatly.

Threat Analysis

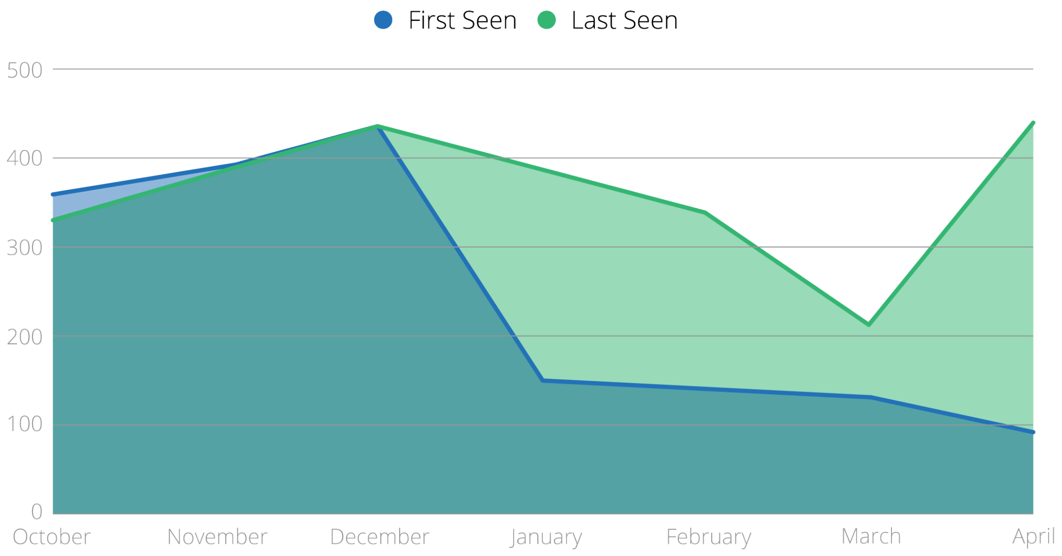
By using Fox-IT's methodology and looking for use of the standard-issue Cobalt Strike TLS certificate on accessible IPs, Recorded Future attempted to profile Cobalt Strike usage in the wake of Strategic Cyber LLC patching a major detection mechanism. It should be noted that the forthcoming methodology and study tracks visible Cobalt Strike servers, and cannot account for Cobalt Strike servers that evade detection even by simple changes.

In this research, Recorded Future anticipated Fox-IT’s findings to shift the adoption of Cobalt Strike to more recent versions, which has occurred, to some extent. Despite Strategic Cyber LLC providing a patch to address this detection, and the publication of IP addresses with the additional space in the HTTP response, Cobalt Strike deployments from before the update do not appear to have been updated. The month following the update to the framework saw the largest increase in newly observed Cobalt Strike servers based on Fox-IT’s detection methodology, as it was applied to Rapid7’s data sets. These servers spent an average of 70 days online.



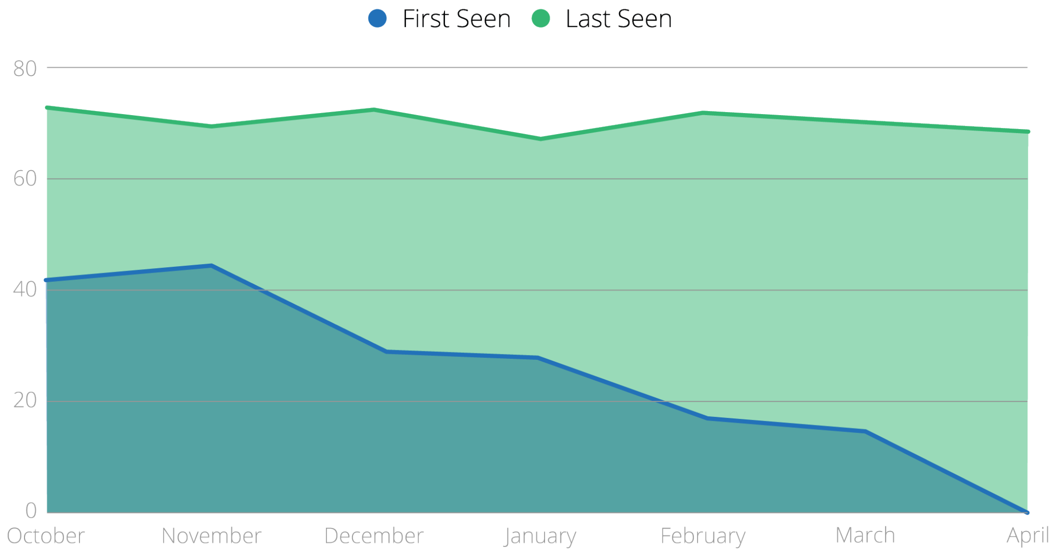
However, this detection proved unreliable, as the method found 248 devices on consecutive CIDR ranges on AS 132839, using NanoHTTPD on port 1443, solely active on February 1, 2019. After removing this anomaly, the data indicates a stark drop in the detection of new Cobalt Strike hosts using NanoHTTPD. This may be due to fewer Cobalt Strike new deployments overall, but may also reflect the updated software being used.

Updated Cobalt Strike Server Sightings



The last-seen data from April 2019 largely indicate that previously deployed Cobalt Strike instances have not been removed or updated. Additionally, the amount of time the servers have stayed online, based on that same data set, shows no noticeable shift in servers that have continued to be detected, hovering around the data set's average of 70 days. However, there was a decline in new Cobalt Strike servers found with the null space in the HTTP header.

Time Online in Days



The continued identification of Cobalt Strike servers using an outdated version of the framework (via the null space in the HTTP header) and the default configurations may indicate that a large population of Cobalt Strike servers are cracked or stolen versions. It may also be an instance of operators not reading security publications, but the answer may be more simple than that — most targets are not likely searching for Cobalt Strike servers, and the payloads remain effective, so why change their behavior?

Sampling of Cobalt Strike Servers

Recorded Future took a sampling of the IP addresses from which we had seen activity in April 2019 to look at both noted activity and detection overlaps. These servers fit into a number of categories: confirmed Cobalt Strike activity, Cobalt Strike servers associated with other malware, Cobalt Strike servers with links to known threat groups, and unreported Cobalt Strike servers that have yet to be named in threat lists or reporting.

The research methods used were unable to help determine if the systems analyzed were licensed or not, and similarly, could not identify if the servers were conducting authorized security testing or illicit attacks.

A number of IP addresses found overlap in signals related to Cobalt Strike. All three made use of the default certificate, had the Cobalt Strike controller port 50050 open, and were previously identified for hosting Cobalt Strike beacons or Meterpreter reverse proxies. It should again be stressed that higher-fidelity detections are made when using corroborating detection methods.

- The IP address **89.105.202.58** made use of the standard Cobalt Strike certificate. Previous URLscan.io results show an HTTP 404 Not Found response, with no content and plain text Content-Type, and Shodan scanning indicates that port 50050 was open, which can host the Cobalt Strike controller. Twitter user @Scumbots has previously identified the server as hosting a Meterpreter reverse proxy, which was contacted via Powershell script that has been hosted on PasteBin since February 2019.

- **199.189.108.71** also made use of the default Cobalt Strike certificate, had port 50050 open, and had previously been identified by Twitter user @Scumbots for hosting a Meterpreter reverse proxy, which also made use of base64-encoded Powershell to obfuscate execution.
- The IP **31.220.43.11** was identified using the baseline Cobalt Strike certificate, corroborated by port 50050 being open on the server. A Meterepreter sample has been observed sending HTTP traffic to the IP in a command-and-control capacity. According to Shodan data, The IP has a number of ports open and is vulnerable to a number of exploits, which may indicate that the host is compromised to serve other malware. The IP hosts a single domain at the time of analysis: cob.ozersk[.] today.

A number of IPs used the standard Cobalt Strike certificate, and had been previously associated with FIN6 activity, for both the delivery of ransomware and the initial attack vector to distribute point-of-sale malware. At the time of this analysis, both of these Cobalt Strike Team Servers were active, despite the campaign being publicly burned. This speaks to FIN6's lack of need for clean up after its operations, as well as the speed with which the operation was abandoned.

Interestingly, while one of the servers was detectable by all three methods, one of the servers had been patched for the NanoHTTPD extra space, implying that either the standard web server was reconfigured, or the actors had an updated version of Cobalt Strike. The diversity of the Cobalt Strike servers deployed in the same incident show that FIN6 uses the standard Cobalt Strike framework with little modification.

- The server at **185.80.233.166** uses the default Cobalt Strike security certificate. This system also has the default Cobalt Strike Team Server port 50050/TCP open. The system had an MX record of mail.sexlove24[.]com, and Talos telemetry data indicates that no mail has been observed to or from this system in the month of April 2019. This IP was identified by Morphisec in February 2019 as part of a coordinated attack on point-of-sale systems using FrameworkPOS. The activity made use of TTPs used by the FIN6 group, specifically the use of WMI/PowerShell for lateral movement and privilege escalation.

- The IP **176.126.85.207** was detected both by Fox-IT's anomalous space and the use of the default Cobalt Strike certificate, with data corroborated by having port 50050 open. The IP has been observed delivering a Metasploit Meterpreter reverse HTTP payload in conjunction with LockerGoga and Ryuk delivery from FIN6.

Two IPs used the standard Cobalt Strike certificate, and made use of Cobalt Strike reflective loaders. Reflective DLL (dynamic load library) loading is a method of injecting a DLL into the memory of a process while bypassing the Windows DLL loader and avoiding storing the DLL on a disk. A DLL injected in this manner may be difficult to detect, as it is only resident in memory. Reflective DLL loading, famously used by APT40 (also known as TEMP.Periscope) and in the Wilted Tulip campaign, is not exclusive to Cobalt Strike and is conducted through various means by a number of actors. The use of a reflective loader is not evidence that these groups were active on these servers. Neither IP address hosted domains at the time of this analysis.

- The IP **89.105.198.18** made use of the default Cobalt Strike certificate as well. The IP previously was identified as a command-and-control server, receiving Meterpreter data over HTTP, according to @Scumbots. Previous scan data from Shodan corroborated the Cobalt Strike server existing on the IP address by having the Cobalt Strike controller port 50050 open.
  Recorded Future's collections identified two files which contacted the IP address 89.105.198.18, first observed in March 2019. The payloads had inconsistent detections on VirusTotal, likely due to at least the first file being UPX-packed. An inspection of the memory dumps from executing the files found that both were Cobalt Strike reflective loaders.

```
3a143d038aae9e4253ed6656beaaae298795a3df20e874544c0122435ef79bc0
9668c17504a0d9471668dac64b3c5c2abfb3b186c25dc28d91afbe95ed341002:00:00)
```

Another IP address on the same CIDR range also made use of the default Cobalt Strike certificate: **89.105.198.21**. The IP address did not host domains at the time of this analysis, but previous scan data corroborated the presence of a Cobalt Strike server.

- The IP **106.12.204.25** was detected both by Fox-IT's anomalous space and the use of the default Cobalt Strike certificate. The IP also had port 50050 open, and had a plaintext 404 Not Found response, as mentioned above. The IP has been reported as delivering with a Cobalt Strike beacon, which was also detected by a VirusTotal user as a Cobalt Strike reflective loader related to APT40. Recorded Future has not observed the IP operating in connection with APT40.

Another general category of IPs that was identified as hosting Cobalt Strike had uncorrelated threat activity involving other malware or suspicious activity, but largely produced inconclusive results.

- The IP address **91.152.8.14** made use of the standard Cobalt Strike certificate in mid-April 2019. A generic trojan was found to communicate with the IP address via HTTP methods over port 433. The IP hosted no domains at the time of analysis, but shared a certificate with forum.happyhippos[.]org. The certificate issuer claimed to be from Espoo, Uusimaa, Finland, the same relative geolocation of the IP address. Another IP address on the same CIDR range was detected via the anomalous HTTP header space, on **91.152.8.173**. While this IP made use of a different certificate, previous Shodan scans over port 443 show a 404 Not Found response with no content and plain text Content-Type, which is a low-confidence signal of a Cobalt Strike server. Without further data, Recorded Future could not come to a conclusion about these IPs.
- The IP **99.81.122.12** was identified in late April 2019, from the anomalous spacing, use of the Cobalt Strike certificate, and having the controller port 50050 open. The server is now inactive, but previously served as a Cobalt Strike beacon, accessed via HTTP. The server did not host domains at the time of analysis.

- The IP **72.14.184.90** also made use of the generic Cobalt Strike certificate. The IP address is contacted by a malicious file, reaching out over HTTP to the URL hxxps://72.14.184[.]90/search/news/. The file is detected as a Cobalt Strike beacon. The IP address was also implicated for being involved in a spearphishing campaign in late January 2019. Shodan scan data indicates the server has a number of vulnerabilities, which points to the server potentially being compromised to host the Cobalt Strike server, rather than the server being rented for a pen-testing engagement.

```
06f8004835c5851529403f73ad23168b1127315d02c68e0153e362a73f915c72
```

Finally, a number of IPs had limited reports of threat activity, but bore indications of potential malicious activity coming in the near term:

- The IP 172.96.250.199 used the baseline Cobalt Strike certificate, but has not had any threat activity associated with it, according to Recorded Future telemetry. The IP address has since swapped certificates to use a pair of LetsEncrypt certificates, including one linked to the domain haqiu[.]cf. The domain was neither active nor hosted on this IP at the time of this analysis. The IP has been associated with hosting a suspicious domain, ssss.ppwu[.]xyz, which has made use of dual LetsEncrypt certificates. These certificates are good for only 90 days and have since been rotated onto Cloudflare servers. This may indicate a forthcoming red-team engagement, or future threat activity using Cobalt Strike.
- The IP 139.162.18.83 was identified from the use of the default Cobalt Strike SSL certificate, but no threat activity or other oddities, have been observed on the IP address. However, on the same CIDR range, 139.162.18.179 was identified from the anomalous space included in the HTTP response, and it was found to be using the default Cobalt Strike SSL certificate. There is no threat activity currently associated with the server, but a number of suspicious domains are hosted on the IP.

- The IP 124.156.106.98 also made use of the default Cobalt Strike certificate, and had port 50050 open which can be used for the Cobalt Strike controller panel. The IP has been observed as the command and control for a Cobalt Strike beacon, observed in March 2019. However, an odd domain was registered and hosted on the IP as of May 2, 2019, kongbu.koubaogangjiao[.]xyz, while the Cobalt Strike signals were still live. The domain may be used going forward for penetration testing or malicious infections.

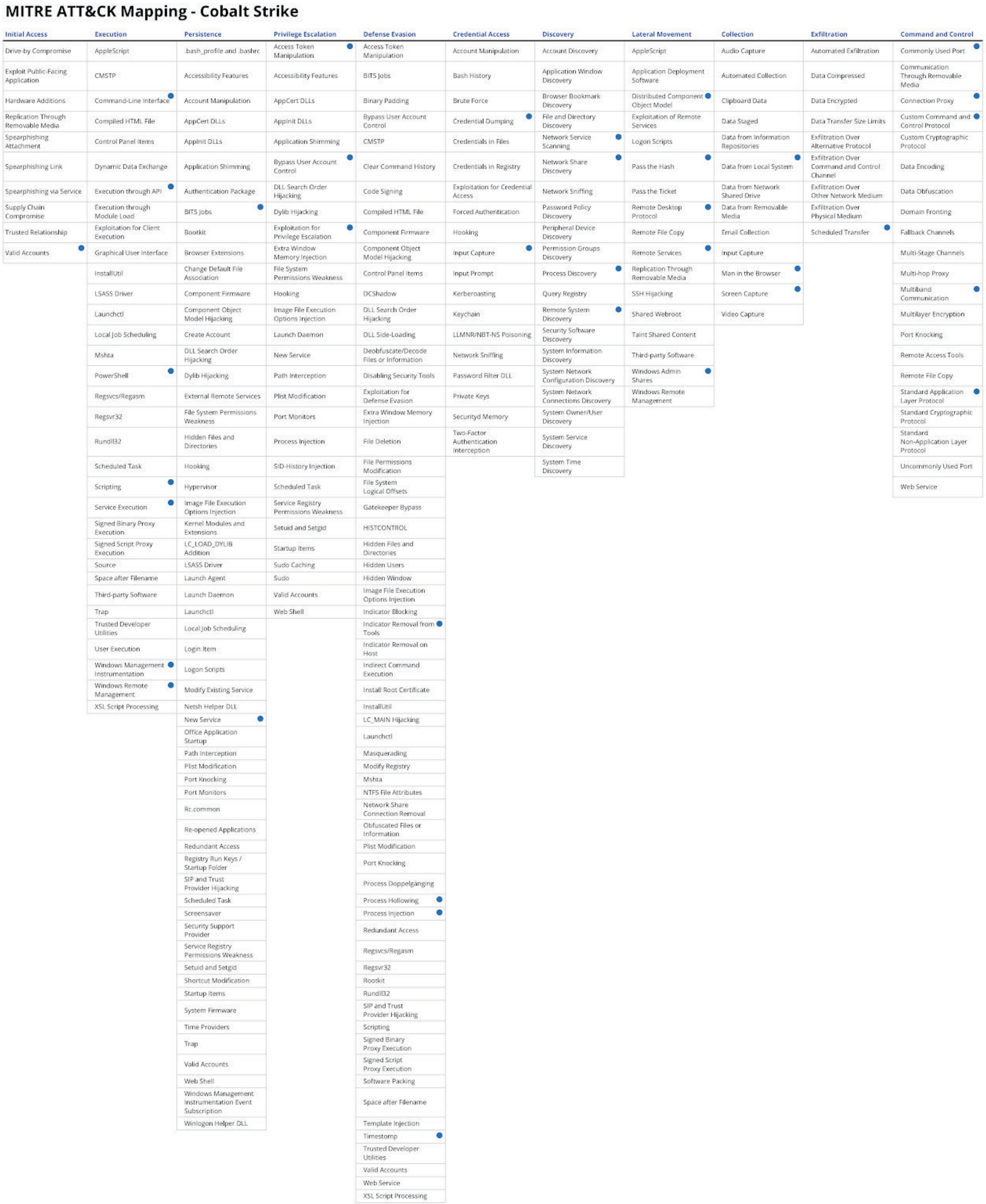
Outlook

Recorded Future finds it important to cluster together signals from known threats to help baseline threat activity and to make it easier to identify more unique threats. The continued sightings of standard Cobalt Strike certificates, along with the anomalous space in HTTP responses from versions earlier than 3.13, indicates that the collaborative use of multiple signatures will prove to be the best method for identifying active Cobalt Strike servers.

While espionage-oriented actors often have large amounts of development time and resources at their disposal, they also have a vested interest to blend in with the crowd. Obstacles other than intentional tradecraft may prevent the patching of Cobalt Strike servers, including lack of knowledge of the update due to a language barrier, operational comfort with currently installed versions, or other modifications that prevent the installation of the update. The use of cracked versions of Cobalt Strike or deployment of standard Cobalt Strike instances causes a blending together of threats, making attribution difficult. Additionally, by running cracked versions of the framework, actors can blend in with older versions of Cobalt Strike.

Detection of these servers on a rolling basis can provide rules for SOC and IR teams to develop alerting or blocking capabilities, and can prompt investigations into hosts communicating with these servers.

Appendix A — MITRE ATT&CK Mapping



MITRE ATT&CK Mapping - Cobalt Strike

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control
Drive-by Compromise	AppleScript	.bash_profile and .bashrc	Access Token Manipulation ●	Access Token Manipulation	Account Manipulation	Account Discovery	AppleScript	Audio Capture	Automated Exfiltration	Commonly Used Port ●
Exploit Public-Facing Application	CMSTP	Accessibility Features	Accessibility Features	BITS Jobs	Bash History	Application Window Discovery	Application Deployment Software	Automated Collection	Data Compressed	Communication Through Removable Media
Hardware Additions	Command-Line Interface ●	Account Manipulation	AppCert DLLs	Binary Padding	Brute Force	Browser Bookmark Discovery	Distributed Component Object Model ●	Clipboard Data	Data Encrypted	Connection Proxy ●
Replication Through Removable Media	Compiled HTML File	AppCert DLLs	AppInit DLLs	Bypass User Account Control	Credential Dumping ●	File and Directory Discovery	Exploitation of Remote Services	Data Staged	Data Transfer Size Limits	Custom Command and Control Protocol ●
Spearphishing Attachment	Control Panel Items	AppInit DLLs	Application Shimming	CMSTP	Credentials in Files	Network Service Scanning ●	Logon Scripts	Data from Information Repositories	Exfiltration Over Alternative Protocol	Custom Cryptographic Protocol
Spearphishing Link	Dynamic Data Exchange	Application Shimming	Bypass User Account Control ●	Clear Command History	Credentials in Registry	Network Share Discovery ●	Pass the Hash ●	Data from Local System ●	Exfiltration Over Command and Control Channel	Data Encoding
Spearphishing via Service	Execution through API ●	Authentication Package	DLL Search Order Hijacking	Code Signing	Exploitation for Credential Access	Network Sniffing	Pass the Ticket	Data from Network Shared Drive	Exfiltration Over Other Network Medium	Data Obfuscation
Supply Chain Compromise	Execution through Module Load	BITS Jobs ●	Dylib Hijacking	Compiled HTML File	Forced Authentication	Password Policy Discovery	Remote Desktop Protocol ●	Data from Removable Media	Exfiltration Over Physical Medium	Domain Fronting
Trusted Relationship	Exploitation for Client Execution	Bootkit	Exploitation for Privilege Escalation ●	Component Firmware	Hooking	Peripheral Device Discovery	Remote File Copy	Email Collection	Scheduled Transfer ●	Fallback Channels
Valid Accounts ●	Graphical User Interface	Browser Extensions	Extra Window Memory Injection	Component Object Model Hijacking	Input Capture ●	Permission Groups Discovery	Remote Services ●	Input Capture		Multi-Stage Channels
	InstallUtil	Change Default File Association	File System Permissions Weakness	Control Panel Items	Input Prompt	Process Discovery ●	Replication Through Removable Media	Man in the Browser ●		Multi-hop Proxy
	LSASS Driver	Component Firmware	Hooking	DCShadow	Kerberoasting	Query Registry	SSH Hijacking	Screen Capture ●		Multiband Communication ●
	Launchctl	Component Object Model Hijacking	Image File Execution Options Injection	DLL Search Order Hijacking	Keychain	Remote System Discovery ●	Shared Webroot	Video Capture		Multilayer Encryption
	Local Job Scheduling	Create Account	Launch Daemon	DLL Side-Loading	LLMNR/NBT-NS Poisoning	Security Software Discovery	Taint Shared Content			Port Knocking
	Mshta	DLL Search Order Hijacking	New Service	Deobfuscate/Decode Files or Information	Network Sniffing	System Information Discovery	Third-party Software			Remote Access Tools
	PowerShell ●	Dylib Hijacking	Path Interception	Disabling Security Tools	Password Filter DLL	System Network Configuration Discovery	Windows Admin Shares ●			Remote File Copy
	Regsvcs/Regasm	External Remote Services	Plist Modification	Exploitation for Defense Evasion	Private Keys	System Network Connections Discovery	Windows Remote Management			Standard Application Layer Protocol ●
	Regsvr32	File System Permissions Weakness	Port Monitors	Extra Window Memory Injection	Securityd Memory	System Owner/User Discovery				Standard Cryptographic Protocol
	Rundll32	Hidden Files and Directories	Process Injection	File Deletion	Two-Factor Authentication Interception	System Service Discovery				Standard Non-Application Layer Protocol
	Scheduled Task	Hooking	SID-History Injection	File Permissions Modification		System Time Discovery				Uncommonly Used Port
	Scripting ●	Hypervisor	Scheduled Task	File System Logical Offsets						Web Service
	Service Execution ●	Image File Execution Options Injection	Service Registry Permissions Weakness	Gatekeeper Bypass						
	Signed Binary Proxy Execution	Kernel Modules and Extensions	Setuid and Setgid	HISTCONTROL						
	Signed Script Proxy Execution	LC_LOAD_DYLIB Addition	Startup Items	Hidden Files and Directories						
	Source	LSASS Driver	Sudo Caching	Hidden Users						
	Space after Filename	Launch Agent	Sudo	Hidden Window						
	Third-party Software	Launch Daemon	Valid Accounts	Image File Execution Options Injection						
	Trap	Launchctl	Web Shell	Indicator Blocking						
	Trusted Developer Utilities	Local Job Scheduling		Indicator Removal from Tools ●						
	User Execution	Login Item		Indicator Removal on Host						
	Windows Management Instrumentation ●	Logon Scripts		Indirect Command Execution						
	Windows Remote Management ●	Modify Existing Service		Install Root Certificate						
	XSL Script Processing	Netsh Helper DLL		InstallUtil						
		New Service ●		LC_MAIN Hijacking						
		Office Application Startup		Launchctl						
		Path Interception		Masquerading						
		Plist Modification		Modify Registry						
		Port Knocking		Mshta						
		Port Monitors		NTFS File Attributes						
		Rc.common		Network Share Connection Removal						
		Re-opened Applications		Obfuscated Files or Information						
		Redundant Access		Plist Modification						
		Registry Run Keys / Startup Folder		Port Knocking						
		SIP and Trust Provider Hijacking		Process Doppelgänging						
		Scheduled Task		Process Hollowing ●						
		Screensaver		Process Injection ●						
		Security Support Provider		Redundant Access						
		Service Registry Permissions Weakness		Regsvcs/Regasm						
		Setuid and Setgid		Regsvr32						
		Shortcut Modification		Rootkit						
		Startup Items		Rundll32						
		System Firmware		SIP and Trust Provider Hijacking						
		Time Providers		Scripting						
		Trap		Signed Binary Proxy Execution						
		Valid Accounts		Signed Script Proxy Execution						
		Web Shell		Software Packing						
		Windows Management Instrumentation Event Subscription		Space after Filename						
		Winlogon Helper DLL		Template Injection						
				Timestomp ●						
				Trusted Developer Utilities						
				Valid Accounts						
				Web Service						
				XSL Script Processing						

About Recorded Future

Recorded Future arms security teams with the only complete threat intelligence solution powered by patented machine learning to lower risk. Our technology automatically collects and analyzes information from an unrivaled breadth of sources and provides invaluable context in real time and packaged for human analysis or integration with security technologies.