Cobalt Strike: Using Known Private Keys To Decrypt Traffic – Part 1

By Didier Stevens

Published: 2021-10-21 · Archived: 2026-04-05 15:04:07 UTC

We found 6 private keys for rogue Cobalt Strike software, enabling C2 network traffic decryption.

The communication between a Cobalt Strike beacon (client) and a Cobalt Strike team server (C2) is encrypted with AES (even when it takes place over HTTPS). The AES key is generated by the beacon, and communicated to the C2 using an encrypted metadata blob (a cookie, by default).

RSA encryption is used to encrypt this metadata: the beacon has the public key of the C2, and the C2 has the private key.

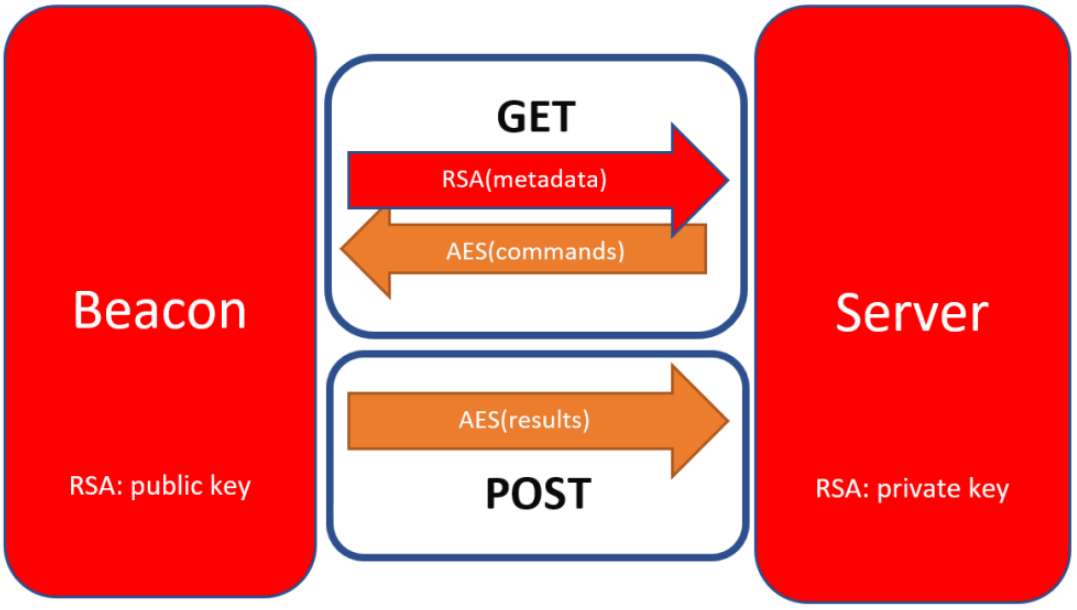

Figure 1: C2 traffic

Public and private keys are stored in file .cobaltstrike.beacon_keys. These keys are generated when the Cobalt Strike team server software is used for the first time.

During our fingerprinting of Internet facing Cobalt Strike servers, we found public keys that are used by many different servers. This implies that they use the same private key, thus that their .cobaltstrike.beacon_keys file is shared.

One possible explanation we verified: are there cracked versions of Cobalt Strike, used by malicious actors, that include a .cobaltstrike.beacon_keys? This file is not part of a legitimate Cobalt Strike package, as it is generated at

first time use.

Searching through VirusTotal, we found 10 cracked Cobalt Strike packages: ZIP files containing a file named .cobaltstrike.beacon_keys. Out of these 10 packages, we extracted 6 unique RSA key pairs.

2 of these pairs are prevalent on the Internet: 25% of the Cobalt Strike servers we fingerprinted (1500+) use one of these 2 key pairs.

This key information is now included in tool 1768.py, a tool developed by Didier Stevens to extract configurations of Cobalt Strike beacons.

Whenever a public key is extracted with known private key, the tool highlights this:

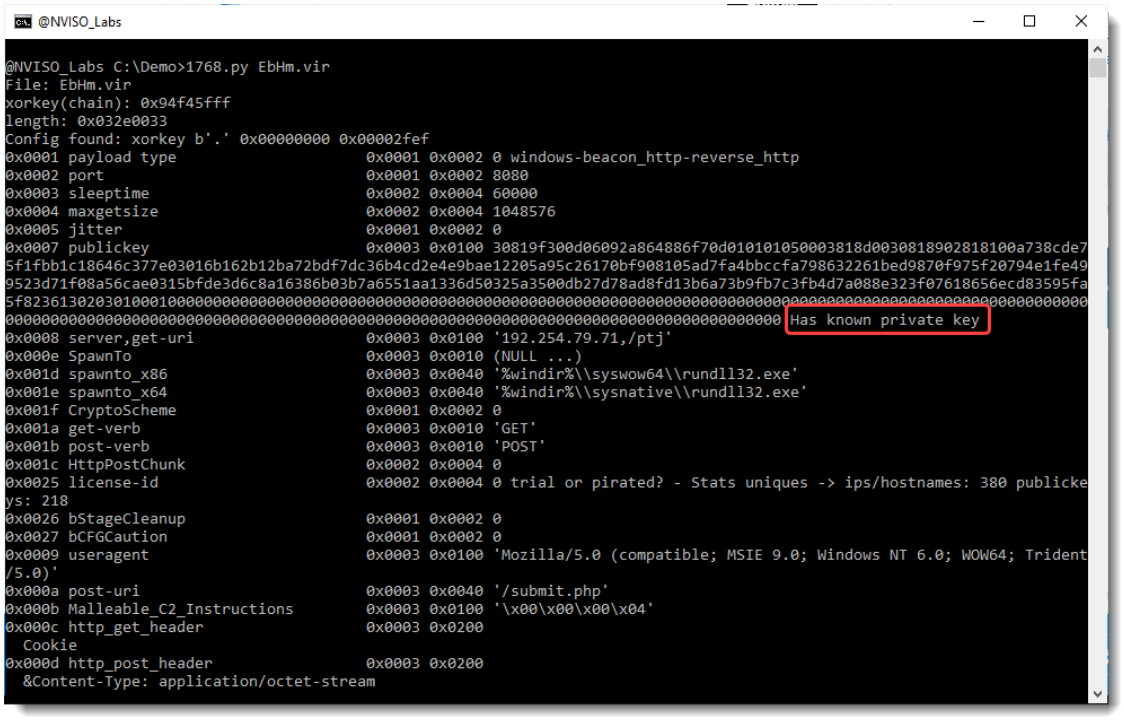

Figure 2: 1768.py extracting configuration from beacon

At minimum, this information is further confirmation that the sample came from a rogue Cobalt Strike server (and not a red team server).

Using option verbose, the private key is also displayed.

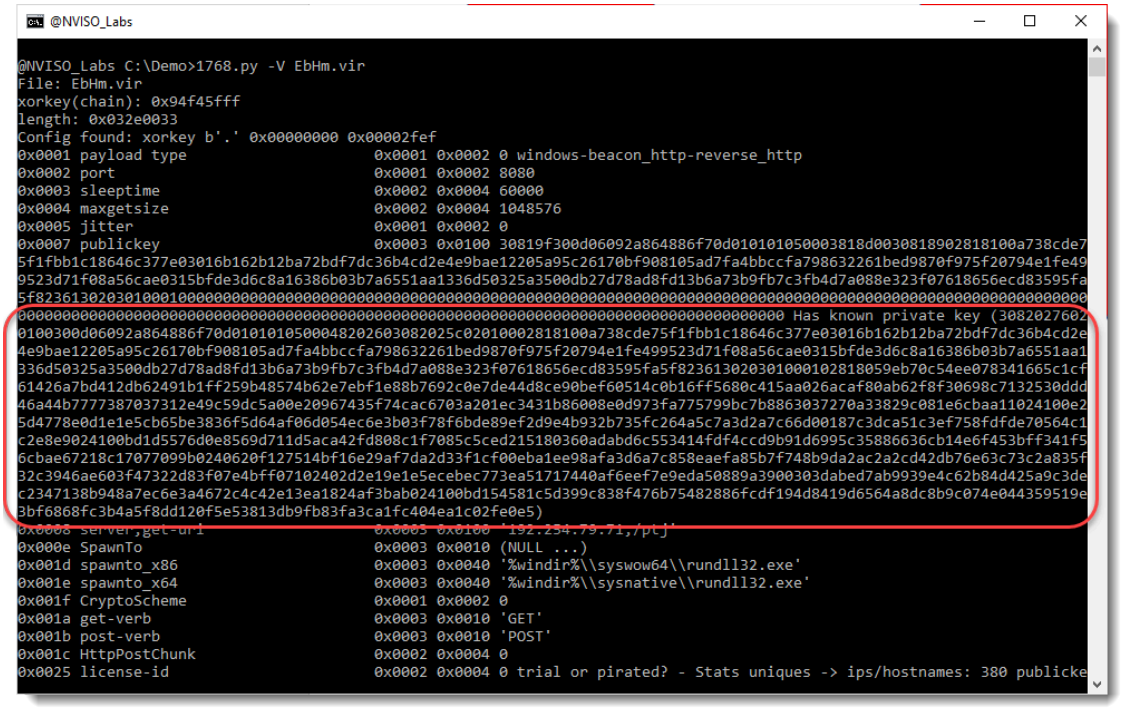

Figure 3: using option verbose to display the private key

This can then be used to decrypt the metadata, and the C2 traffic (more on this later).

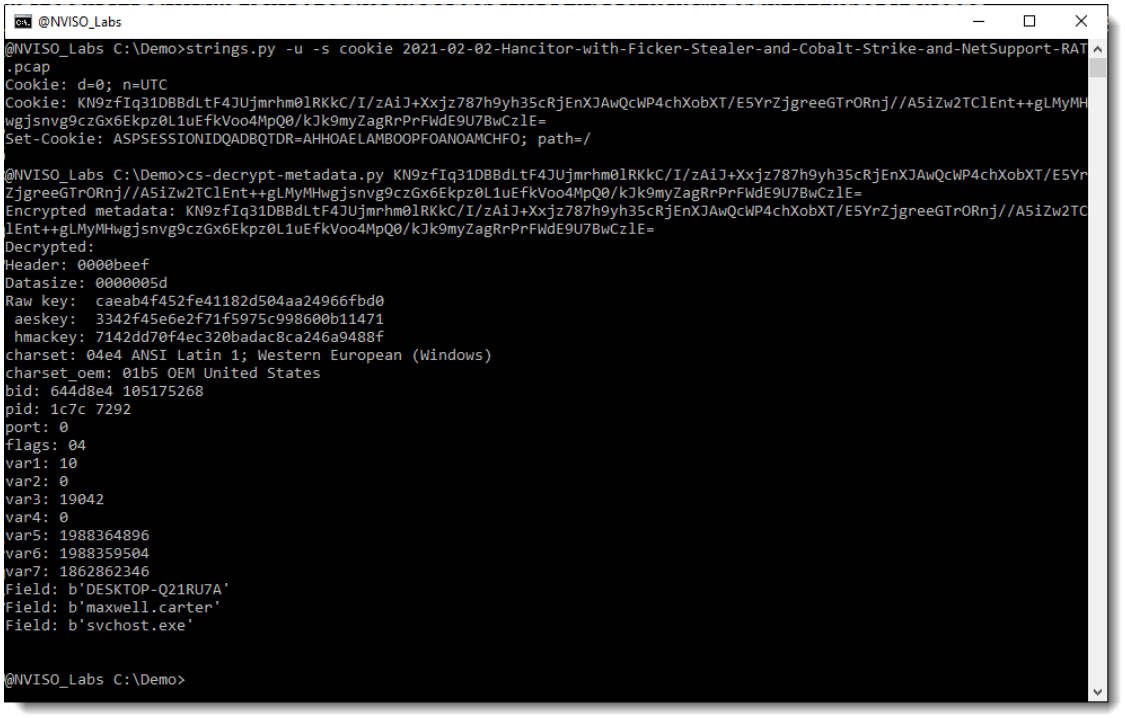

Figure 4: decrypting metadata

In upcoming blog posts, we will show in detail how to use these private keys to decrypt metadata and decrypt C2 traffic.

About the authors

Didier Stevens is a malware expert working for NVISO. Didier is a SANS Internet Storm Center senior handler

and Microsoft MVP, and has developed numerous popular tools to assist with malware analysis. You can find Didier on Twitter and LinkedIn.

You can follow NVISO Labs on Twitter to stay up to date on all our future research and publications.

Published October 21, 2021March 10, 2022

Post navigation
