Malware Family: NTrip

**Summary**:

NTrip is a live of the land malware that uses various advanced techniques to install and execute CobaltStrike Egg and CobaltStrike Beacon. The techniques used by the malware allows the malware to stay in memory, and stay hidden inside a Microsoft’s signed executable. These techniques typically seem in advance APT groups such as CobaltGroup.

**Detection And Mitigation**:

For threat hunting, we recommends searching for the following:

Network:

* https:// raw.githubusercontent[.]com/pick90/hit/master/1.txt?[random]
* https:// n-trip.com/index.png?[random]
* https:// ns1.n-trip.com,/c/msdownload/update/others/2016/12/29136388\_
* https:// ns2.n-trip.com,/c/msdownload/update/others/2016/12/29136388\_
* https:// ns3.n-trip.com,/c/msdownload/update/others/2016/12/29136388\_
* https://download.008ex.com/auth.log?[RANDOM]

FileSystem:

* %USER%\Document\null.hta
* c:\ProgramData\\A164C0BF-67AE-3C7E-BC05-BFE24A8CDB62.dat
* %ALLUSERSPROFILE%\27F31D55-D6C6-3676-9D42-C40F3A918636.dat
* %ALLUSERSPROFILE%\auth.log

Registry:

* HKCU:\Software\{6D8BB3D3-9D87-4a91-AB56-4F30CFFEF007}\Info

Process:

* rundll32 url.dll,OpenURL null.hta
* InstallUtil /u A164C0BF-67AE-3C7E-BC05-BFE24A8CDB62.dat
* mshta.exe %USER%\Documents\null.hta

We recommend implementing a monitor of the processes to detect potential application whitelist bypass.

**Overview**:

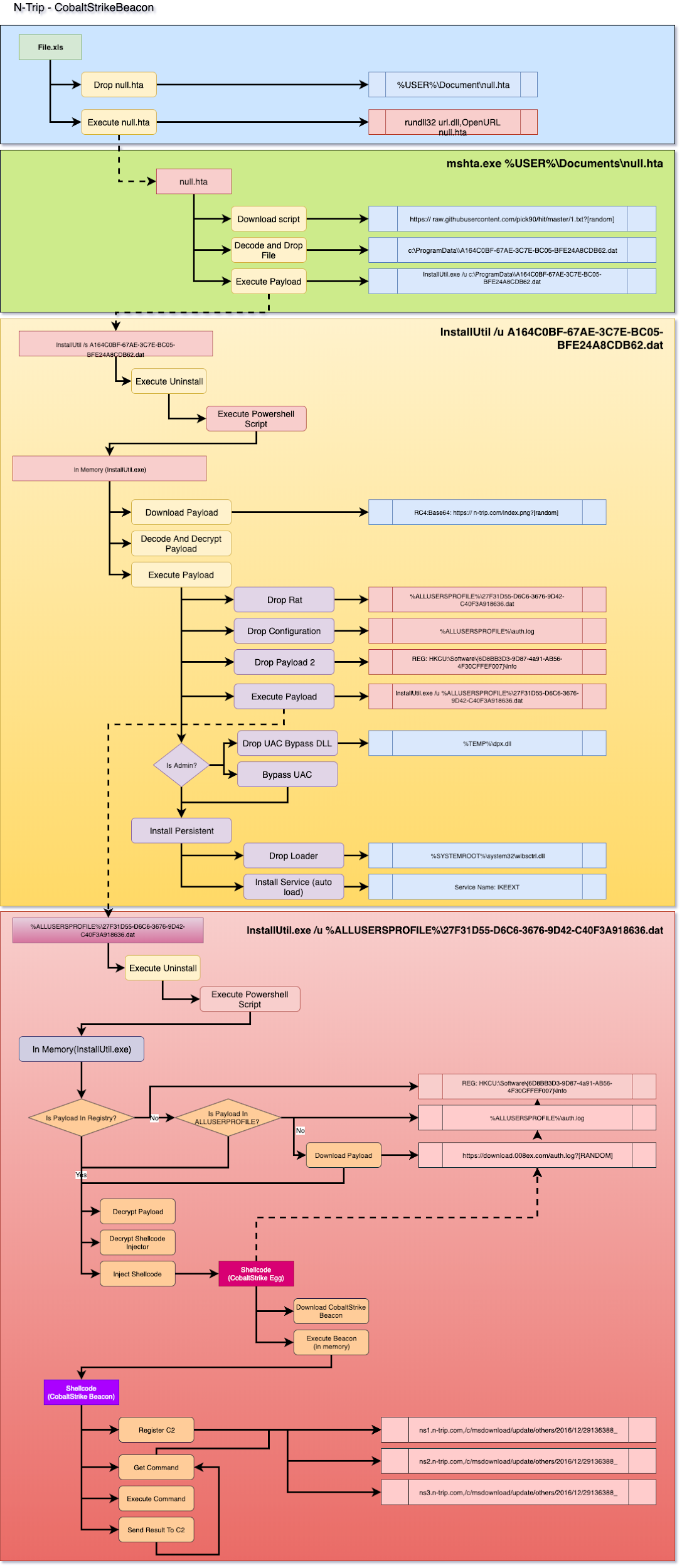
N-Trip is a pseudonym used to classify this live of the land (LOL) malware family. The LOL malware is malwares stay in memory and does not touch the file system. N-Trip malware is a malware used various techniques to stay in memory, stay hidden from anti-virus software and security tools. Below is a list of features of the malware.

Features of N-Trip Malware:

* Leverage whitelisted and signed software(InstallUtil.exe) to execute and host the malware (bypass application whitelisting).
* Leverage Base64 and RC4 to hide itself from static analysis.
* Leverage github (whitelisted) to download 2nd stage payload (bypass network security).
* Leverage https and payload is encrypted using RC4 (bypass heuristic detection).
* Final payload loader is Powershell script encoded and encrypted.

The malware uniquely leverage InstallUtil.exe from .Net Framework to execute the first stage loader. In addition, the first stage executable doesn’t execute normally, the executable must be executed by InstallUtil.exe with “/u” option (uninstall) to trigger the malware. Once the executable is triggered, the executable decode and execute the embedded Powershell script. The Powershell script is a loader which is responsible to decrypt and execute the final Powershell loader. The final loader decrypt the CobaltStrike Egg (Shellcode) and inject the shellcode into the host process.

Figure 1 shows the overview of N-Trip.



**File Properties Overview:**

The N-Trip malware with the following properties.

**MD5**: *2eb28e11e4105b0dbb916237d3f56a2f*

**SHA1**: *f58f52772eb1893033d673d02c70685d9a9ab6fd*

**SHA256**: *e373283c9706803fe0b0f09231ab53111e7e40bce2c300b90c80117b60bad2a5*

**File Type**: *Composite Document File V2 Document*

**File Size**: *294 KB*

***File Name****: mxc客服人员色诱客户侵吞平台财产.xls*

**N-Trip Analysis:**

**First Stage:** Dropper (mxc客服人员色诱客户侵吞平台财产.xls)

The first stage of the NTrip malware is the MS Excel document, which has a MD5 of *2eb28e11e4105b0dbb916237d3f56a2f.* Once the document is opened, the document drop a file named null.hta in %USER%\Documents directory. The null.hta has the following properties.

**File Nane**: null.hta

**MD5**: 2c4866942dd70b11284a44e1fa6e4f52

**SHA1**: 653ce355f1afb94b9e686f572641d81341b637c2

**SHA256**: 1f973b5faa3b1758a02952d2f65628fa08f8eaf09f2c68ed5e1d94dc36674ec3

**File Type**: HTML executable file

Once the file is dropped into the file system, the document opens the HTA file by using the following command.

* *rundll32.exe url.dll,OpenURL null.hta*

The dropper leverage rundll32.exe and url.dll to bypass application whitelisting.

**Second Stage:** Installation (null.hta)

Null.hta is a HTML executable file which VBscript once executes. Figure 2 shows the content of the script and the 3rd stage payload.

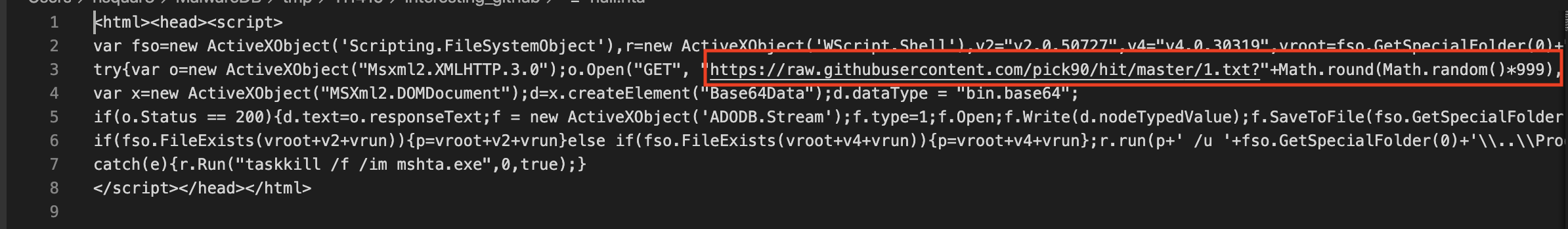


Figure 2: Null.hta VBscript

The script download an encoded Powershell from github.com. Actors use github.com to bypass network detection as it’s a common whitelisted website. The 3rd stage payload is an executable with the following properties.

Filename: 1.bin (A164C0BF-67AE-3C7E-BC05-BFE24A8CDB62.dat)

MD5: 454324267f8871dfbe5999a7266bd6c6

SHA1: d38f2987ae2d0f33f38c906e8d6a65b5756648de

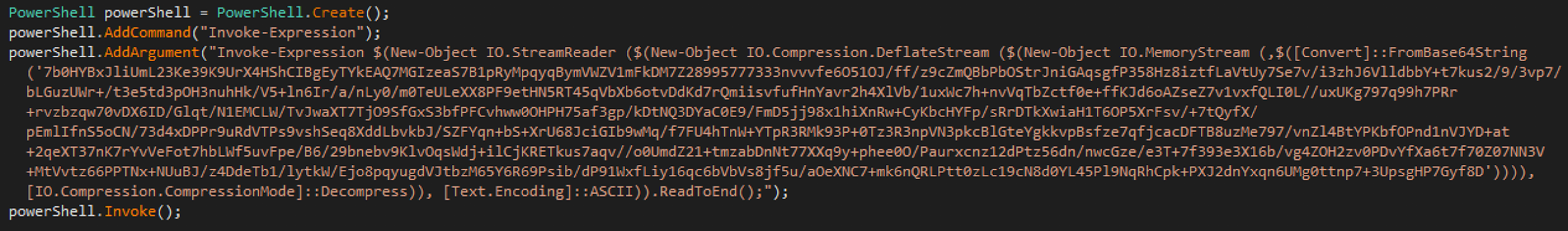
SHA256: 481736a8471479d8cdbb583faa2f2e841febede2ea575640117caad63f09a55f

The VBscript would then execute the 3rd stage with the following command.

* InstallUtil.exe /u A164C0BF-67AE-3C7E-BC05-BFE24A8CDB62.dat

**Third Stage:** Initialization and Persistent Setup

The executable is a .Net Setup file that doesn’t take any action when executed by itself. When the executable gets executed by InstallUtil.exe with the “/u” parameter(Uninstall), the executable executes an embedded Powershell script. Figure 3 shows the Uninstall methods.



The decoded Powershell script download a payload from n-trip.com, decode, and decrypt the payload using RC4 encryption. The script download from the URL below.

* https:// n-trip.com/index.png?[random]

The content is encrypted and encoded in Base64, the downloaded file has the following properties.

File Name: Index.png

MD5: 37722aaa4df57fd3f1e09d49011d3566

SHA1: b943f9162b9fb093838896badcc3bf668cefb6fc

SHA256: 937145aba6c2c2b6b99fdc691f8be0c954365a01efaca69f8728c247257b3fbc

The following Python script decrypt the payload.

*from Crypto.Cipher import ARC4*

*import sys*

*import base64*

*data64 = open(sys.argv[1], "rb").read()*

*data = base64.b64decode(data64)*

*cipher = ARC4.new(data[0:16])*

*decrypted = cipher.decrypt(data[16:])*

*open(sys.argv[1]+".decrypted", "wb").write(decrypted)*

The decrypted content is a Powershell responsible to install the final payload and install persistent. The decrypted content has the following properties.

MD5: dd61def2fbae3c38027084ce9a5a5d0a

SHA1: d6daa23f10a3de626dbb79e60794a76fa350f820

SHA256: cd530b21ef8b2f5cc21590d4db58e2b785dd9291ff6d5a68079d579ba8b5845b

Figure 4 shows the Powershell of the decrypted file.

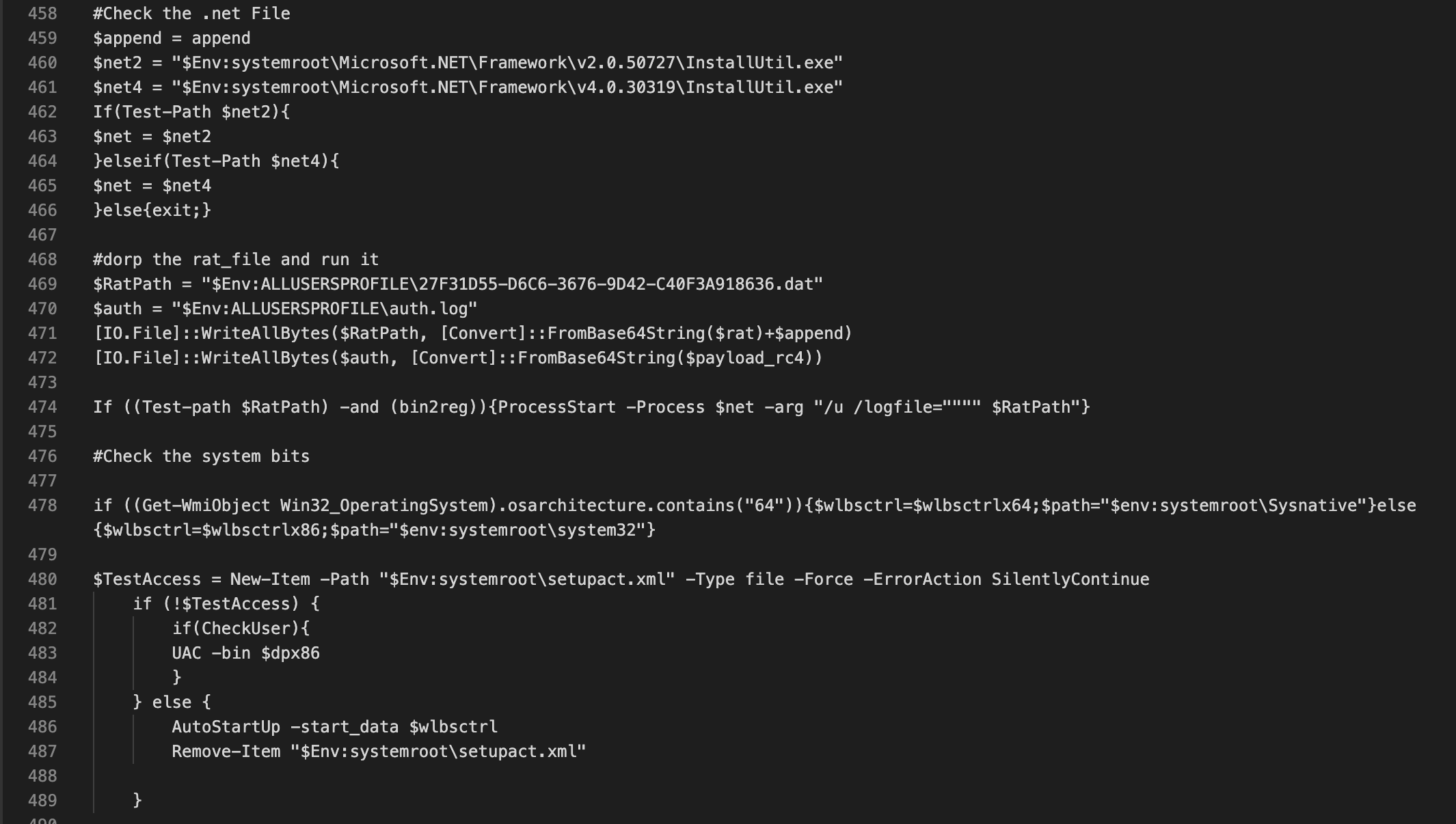


Figure 4: Installer Overview

Once executed, the installer drops the final payload loader and the encrypted payload into the file system. Below is the path of the dropped file.

* %ALLUSERSPROFILE%\27F31D55-D6C6-3676-9D42-C40F3A918636.dat
* %ALLUSERSPROFILE%\auth.log

The loader has the following properties.

MD5: 91479962820d112add59b5d839a1f00d

SHA1: c5e61d719dbd9722ccba668c790ebf59ca5e4c09

SHA256: 835d28e90c390fbb871021272bb8e49a4f21f35061f7c18f22fc5532bbe7c7d7

File type: PE32 executable (DLL) (console) Intel 80386 Mono/.Net assembly, for MS Windows

The encrypted payload has the following properties.

MD5: d3bce95e355f28b833e123eda249bc56

SHA1: 8924562aeab5cc1fd830a58f8b6cac3ce8a2561e

SHA256: 2f403849f00c93e9831d3fd471c2473d2682a975af307d7723fb5c3a37c21d26

File Type: x86 Shellcode (CobaltStrike Egg)

In addition, if the malware unable to drop the final payload into the file system, the malware would encode the loader and payload and install in the registry in the following path.

* HKCU:\Software\{6D8BB3D3-9D87-4a91-AB56-4F30CFFEF007}\Info

Finally the script would then execute the loader with the following command.

* InstallUtil.exe /u %ALLUSERSPROFILE%\27F31D55-D6C6-3676-9D42-C40F3A918636.dat

**Fourth Stage**: Loader & CobaltStrike Egg

The loader is a .Net Setup executable, once triggered correctly, the loader would decode and execute the embedded Powerscript. The script loads the file “Auth.log” and decrypts the payload. It would then inject the shellcode into the current process. If the script is unable to locate the file in the file system, it would download from the URL below.

* https://download.008ex.com/auth.log?[random]

The shellcode is a CobaltStrike Egg, which would connect to the C2 at **ns1.n-trip.com** and download the CobaltStrike Beacon. The CobaltStrike Egg uses the following user-agent.

* User-Agent: Windows-Update-Agent/10.0.10011.16384 Client-Protocol/1.40

**Fifth Stage:** CobaltStrike Beacon

CobaltStrike Beacon is a full backdoor allowing the actor to control the infected machine. The CobaltStrike Beacon has the following properties.

MD5: 0dfab0343b7310ed0bcabe0d09cd8600

SHA1: c095ed9210d7e8c0405d7249b2fb76e0c581f4ab

SHA256: e0b47f5ee850f5a00e85e4d7190432523f3aa3b892b65ac552d49a6d60da4c19

The following is the configuration of the Beacon.

**SLEEPTIME**: 60000

**C2\_REQUEST\_HEADER**: Accept: \*/\*, Host: download.windowsupdate.com

**C2\_VERB\_POST**: GET

**PIPENAME**: \\\\%s\\pipe\\msagent\_%x

**SUBMIT\_URI**: /c/msdownload/update/others/2016/12/3215234\_

**DOMAINS**: ns1.n-trip.com, ns2.n-trip.com, ns3.n-triplcom

**USERAGENT**: Windows-Update-Agent/10.0.10011.16384 Client-Protocol/1.40

**SPAWNTO\_X86**: %windir%\\syswow64\\rundll32.exe

**SPAWNTO\_X64**: %windir%\\sysnative\\rundll32.exe