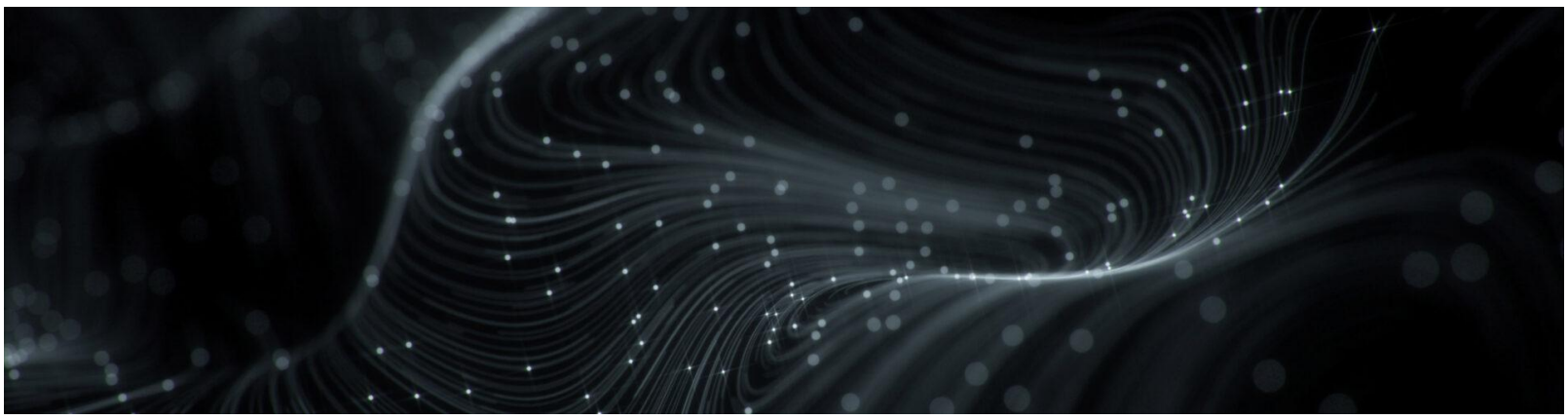
More on Avast Inside Out

- Menu



Avast Finds Compromised Philippine Navy Certificate Used in Remote Access Tool

by Threat Intelligence TeamMarch 28, 20224 min read

Avast Threat Intelligence Team has found a remote access tool (RAT) actively being used in the wild in the Philippines that uses what appears to be a compromised digital certificate belonging to the Philippine Navy. This certificate is now expired but we see evidence it was in use with this malware in June 2020.

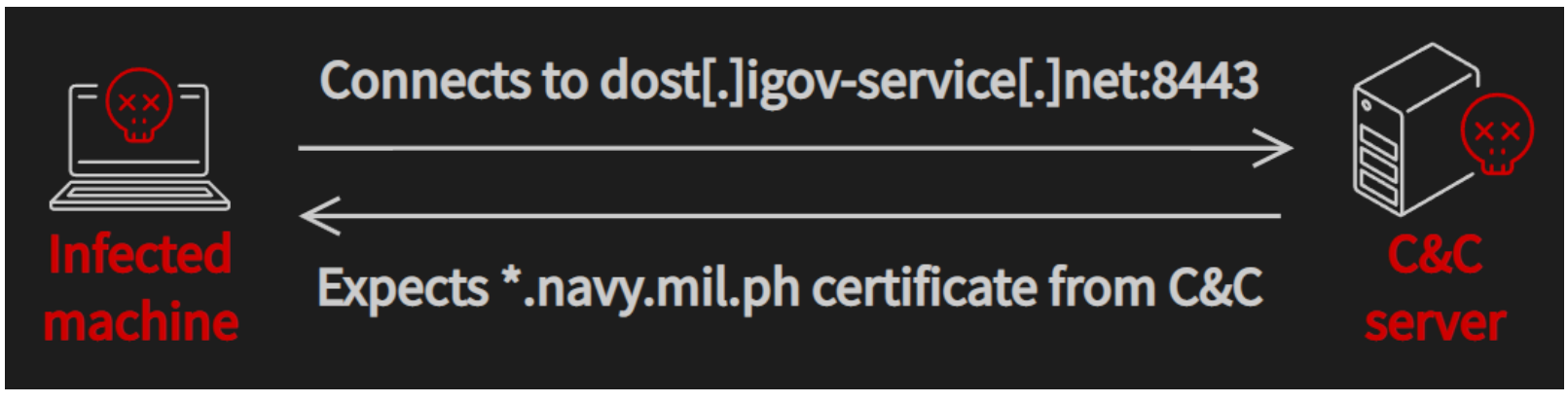
Based on our research, we believe with a high level of confidence that the threat actor had access to the private key belonging to the certificate.

We got in touch with CERT-PH, the National Computer Emergency Response Team for the Philippines to help us contact the navy. We have shared with them our findings. The navy security team later let us know that the incident has been resolved and no further assistance was necessary from our side.

Because this is being used in active attacks now, we are releasing our findings immediately so organizations can take steps to better protect themselves. We have found that this sample is now available on VirusTotal.

Compromised Expired Philippine Navy Digital Certificate

In our analysis we found the sample connects to `dost[.]igov-service[.]net:8443` using TLS in a statically linked OpenSSL library.



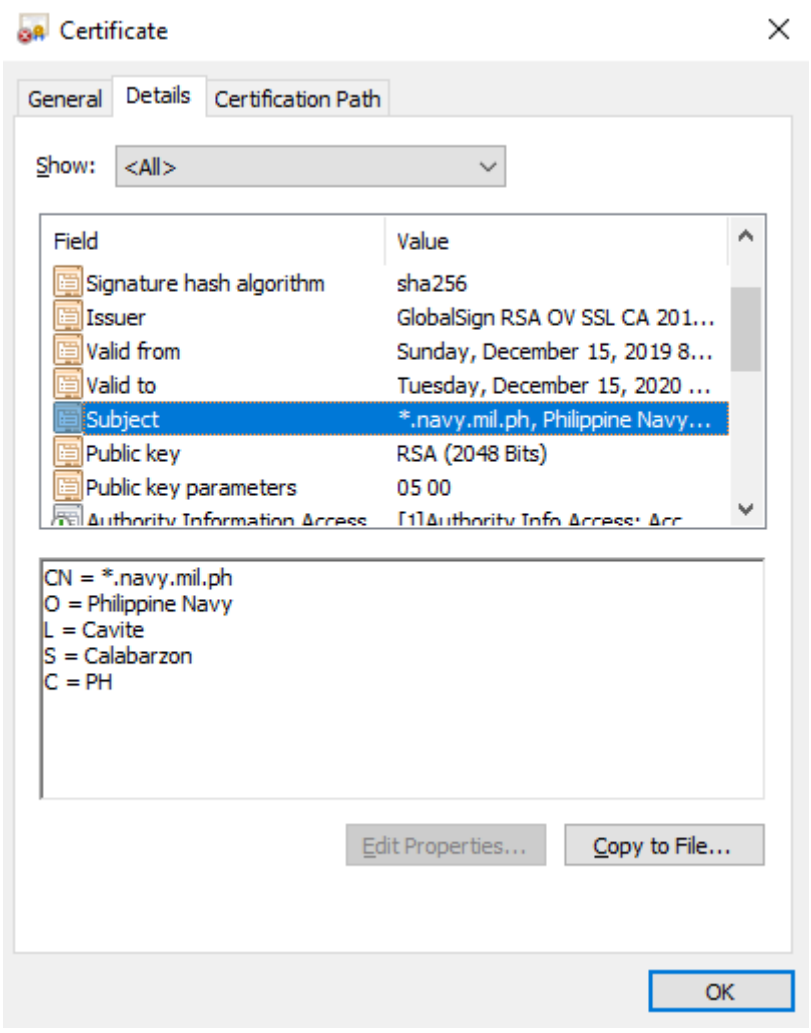
A WHOIS lookup on the C&C domain gave us the following:

Whois Lookup

```
Create date: 2019-12-09
Domain name: igov-service.net
Domain registrar id: 146
Domain registrar url: http://registrar.godaddy.com
Expiry date: 2021-12-09
Name server 1: ns01.domaincontrol.com
Name server 2: ns02.domaincontrol.com
Query time: 2019-12-10 17:42:29
Registrant address: 3267309318f7846c
Registrant city: 3267309318f7846c
Registrant company: 3267309318f7846c
Registrant country: United States
Registrant email: 3267309318f7846cs@
Registrant fax: 3267309318f7846c
Registrant name: 3267309318f7846c
Registrant phone: 3267309318f7846c
Registrant state: 40840a16fcb405fc
Registrant zip: 3267309318f7846c
Update date: 2019-12-09
```

The digital certificate was pinned so that the malware requires the certificate to communicate.

When we checked the digital certificate used for the TLS channel we found the following information:



Some important things to note:

- The certificate is a valid certificate with a subject of `*.navy.mil.ph`, the Philippine Navy.
- The certificate has recently expired: it was valid for one year, from Sunday December 15, 2019 until Tuesday December 15, 2020.
- Our research shows that Censys saw this certificate employed by the actual navy.mil.ph website

Based on our research, we believe with a high level of confidence that the threat actor had access to the private key belonging to the certificate.

While the digital certificate is now expired we see evidence it was in use with this malware in June 2020.

The malicious PE file was found with filename: `C:\Windows\System32\wlbsctrl.dll` and its hash is: `85FA43C3F84B31FBE34BF078AF5A614612D32282D7B14523610A13944AADAACB`.

In analyzing that malicious PE file itself, we found that the compilation timestamp is wrong or was edited. Specifically, the `TimeDateStamp` of the PE file was modified and set to the year 2004 in both the PE header and Debug Directory as shown below:

```
ta:0000000180266700                                     ; sub_18000FF60+81tr ...
ta:0000000180266710 ; Debug Directory entries
ta:0000000180266710                 dd 0                ; Characteristics
ta:0000000180266714                 dd 40E13AA1h        ; TimeDateStamp: Tue Jun 29 09:47:13 2004
ta:0000000180266718                 dw 0                ; MajorVersion
ta:000000018026671A                 dw 0                ; MinorVersion
ta:000000018026671C                 dd 0Dh              ; Type: IMAGE_DEBUG_TYPE_POGO
ta:0000000180266720                 dd 3BCh             ; SizeOfData
ta:0000000180266724                 dd rva aGctl        ; AddressOfRawData
ta:0000000180266728                 dd 269E54h          ; PointerToRawData
ta:000000018026672C                 dd 0                ; Characteristics
ta:0000000180266730                 dd 40E13AA1h        ; TimeDateStamp: Tue Jun 29 09:47:13 2004
ta:0000000180266734                 dw 0                ; MajorVersion
ta:0000000180266736                 dw 0                ; MinorVersion
ta:0000000180266738                 dd 0Eh              ; Type: IMAGE_DEBUG_TYPE_ILTCG
ta:000000018026673C                 dd 0                ; SizeOfData
ta:0000000180266740                 dd 0                ; AddressOfRawData
ta:0000000180266744                 dd 0                ; PointerToRawData
ta:0000000180266748                 align 10h
```

```
:0000000180001000 ; Format      : Portable executable for AMD64 (PE)
:0000000180001000 ; Imagebase   : 180000000
:0000000180001000 ; Timestamp   : 40E13AA1 (Tue Jun 29 09:47:13 2004)
:0000000180001000 ; Section 1. (virtual address 00001000)
:0000000180001000 ; Virtual size                  : 001CA5D1 (1877457.)
:0000000180001000 ; Section size in file          : 001CA600 (1877504.)
:0000000180001000 ; Offset to raw data for section: 00000400
:0000000180001000 ; Flags 60000020: Text Executable Readable
:0000000180001000 ; Alignment     : default
:0000000180001000 ; OS type          :  MS Windows
:0000000180001000 ; Application type:  DLL
:0000000180001000
```

However, we found that the author used OpenSSL 1.1.1g and compiled it on April 21, 2020 as shown below:

```
rdata:00000001801E6772                 db    0
rdata:00000001801E6773                 db    0
rdata:00000001801E6774                 db 0A2h ; ¢
rdata:00000001801E6775                 db    2
rdata:00000001801E6776                 db    0
rdata:00000001801E6777                 db    0
rdata:00000001801E6778 aOpenssl111g21A db 'OpenSSL 1.1.1g  21 Apr 2020',0
rdata:00000001801E6794                 db    0
rdata:00000001801E6795                 db    0
rdata:00000001801E6796                 db    0
rdata:00000001801E6797                 db    0
```

The username of the author was probably `udste`. This can be seen in the debug information left inside the used OpenSSL library.

```
1E6798                 db    0
1E6799                 db    0
1E679A                 db    0
1E679B                 db    0
1E679C                 db    0
1E679D                 db    0
1E679E                 db    0
1E679F                 db    0
1E67A0 ; const char aCUsersUdsteDes[]
1E67A0 aCUsersUdsteDes db 'C:\Users\udste\Desktop\openssl\openssl-1.1.1g\ssl\packet_local.h',0
1E67A0                                         ; DATA XREF: sub_180092300+209↑o
1E67A0                                         ; sub_180092300+240↑o ...
1E67E1                 align 8
1E67E8 ; const char file[]
1E67E8 file            db 'ssl\ssl_lib.c',0    ; DATA XREF: sub_18008F340+D↑o
1E67E8                                         ; ssl_accept+D↑o ...
1E67F6                 align 20h
```

We found that the malware supported the following commands:

- run shellcode
- read file
- write file
- cancel data transfer
- list drives
- rename a file
- delete a file
- list directory content

```
; enum COMMANDS, mappedto_2257, width 2 bytes
RUN_SHELL_COMMAND  = 1
READ_FILE          = 5
WRITE_FILE         = 6
CANCEL_TRANSFER    = 7
GetDrives          = 8
DoPathRename       = 9
DoPathDelete       = 0Ah
GetDirectory       = 0Ch
QUIT               = 1Eh
```

Some additional items of note regarding the malicious PE file:

- All configuration strings in the malware are encrypted using `AES-CBC` with the exception of the mutex it uses.That mutex is used as-is without decryption: `t7As7y9I6EGwJOQkJz1oRvPUFx1CJTsjzgDlm0CxIa4=`.
- When this string is decrypted using the hard-coded key it decrypts to `QSR_MUTEX_zGKwWAejTD9sDitYcK`. We suspect that this is a failed attempt to disguise this malware as the infamous Quasar RAT malware. But this cannot be the case because this sample is written in C++ and the Quasar RAT is written in C#.

Avast customers are protected against this malware.

Indicators of Compromise (IoC)

- Repository: https://github.com/avast/ioc/tree/master/Philippine-Navy-Certificate

SHA256	File name
85FA43C3F84B31FBE34BF078AF5A614612D32282D7B14523610A13944AADAACB	C:\Windows\System32\wlbsctrl.dll

Mutex

t7As7y9I6EGwJOQkJz1oRvPUFx1CJTsjzgDlm0CxIa4=

C&C server

dost[.]igov-service[.]net:8443

Tagged asanalysis, backdoor, malware, rat Share:TwitterFacebook