# Deceiving Bloodhound - Remote Registry Session Spoofing

blog.spookysec.net/DnD-Deceiving-BH

24 Aug 2022

Hello Everyone!

Today we're back with another blog post in the Deception in Depth series. Recently, I've found a new way to spoof user sessions using Windows' Remote Registry feature. Before we begin talking about how to spoof a user session using this method, we first have to understand *how* SharpHound performs Session Enumeration.

## SharpHound Session Enumeration

There are primarily three different ways that SharpHound performs Session enumeration. Here's quick breakdown for all three:

1. NetSessionEnum - Enumeration is done via Win32 API calls. Moderately accurate. This information is considered privileged as of Windows 10 v1607+ and Server 2016+.
2. NetWkStaUserEnum - Enumeration is done via Win32 API calls. Most accurate. This information is privileged and always requires Admin to collect.
3. RegistryKey.OpenRemoteBaseKey .NETv6 Method - Ability to read the remote registry on the victim system(s). This feature is generally not enabled by default.

### A Quick Look Into the Past

So far we've managed to tackle the first method by using the CreateProcessWithLogonW and a specific flag within that Win32 API that does not attempt to validate credentials against Active Directory. There is one important thing to note; If you call this program as an Administrator, it will run with the privileges of the caller. This is immensley useful for us, knowing this, we can make a call to `net use \\localhost\c$` (or something like that) to create an artificial SMB session.

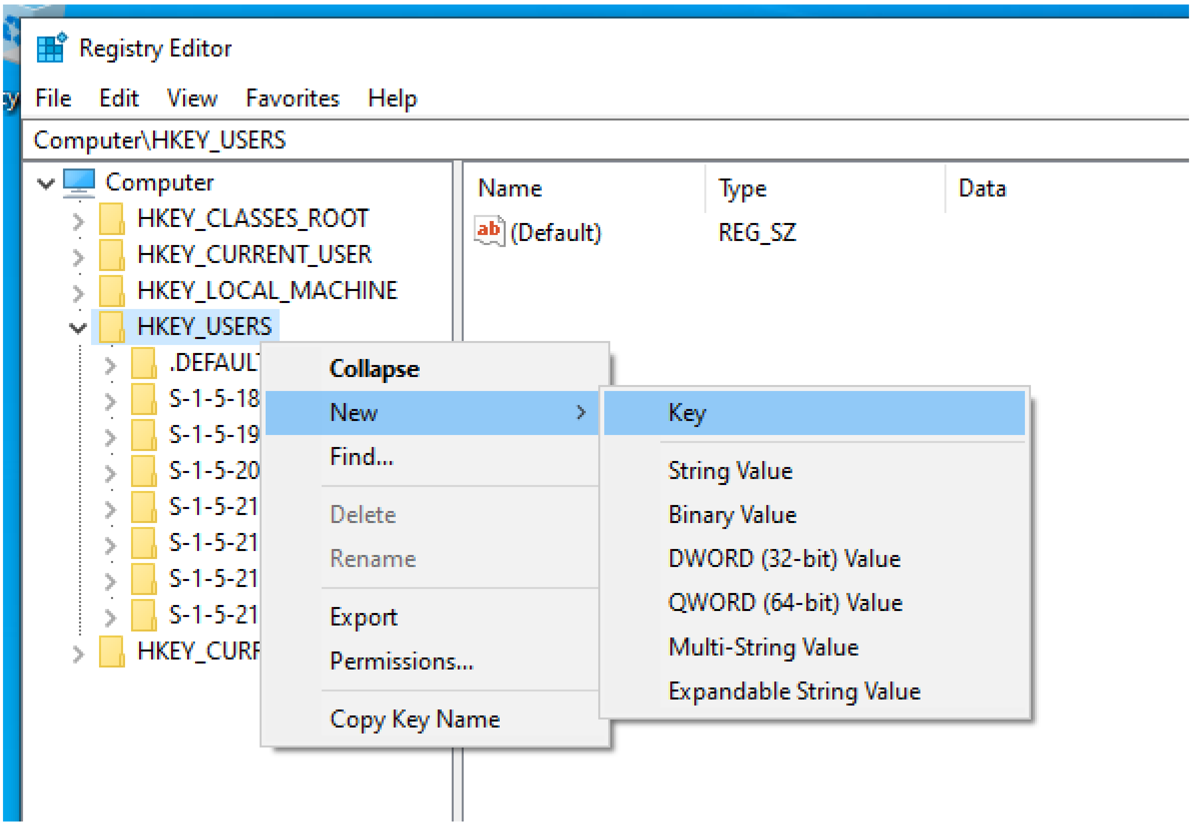
This is a relatively complicated approach, but is super effective. You can read my Blog post about it here.

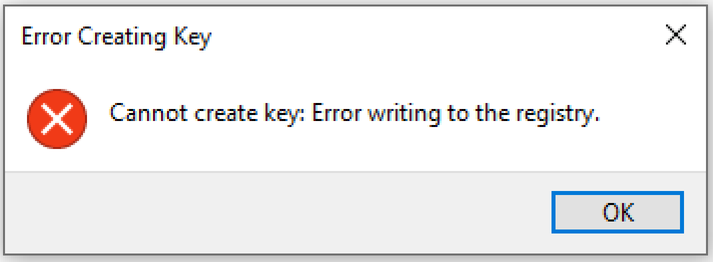
## Another Day Another Method

As I described earlier, this new method leverages Windows' Rmote Registry. Option 3 performs a call to the Remote Registry service running on another device to list all of the entries in HKEY_USERS. This is where user profile data is stored during a logon session, another fun fact is that this data is **only** loaded during a logon session, and at no other time. After the session is finished, or the computer is shut down, the profile is flushed from the registry. This introduces an opportunity for us; We can create a new scheduled task *or* service to create a registry key under HKEY_USERS.

## Trial and Error

Ok - Confession time, I made this sound a lot easier than it actually is… Going into the Registry, right clicking and adding a new key under `HKEY_USERS` doesn't actually work…



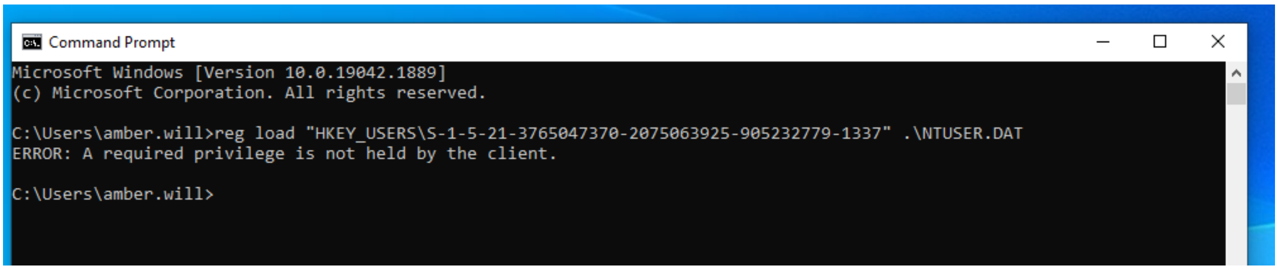


Error Creating Key

Cannot create key: Error writing to the registry.

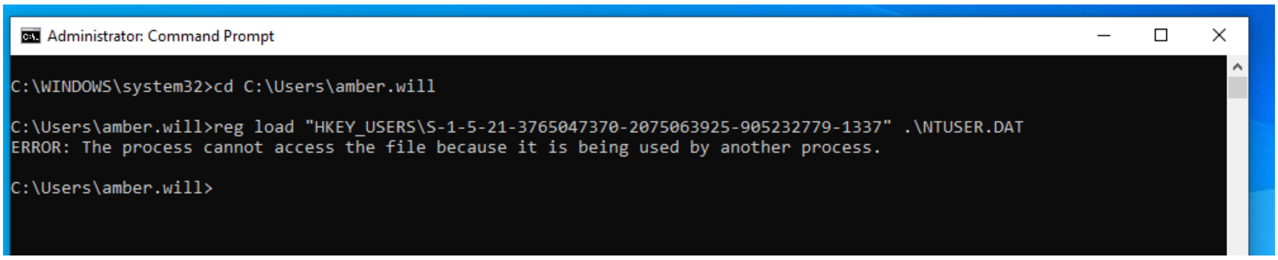
:(

Plan foiled… But wait; If Windows loads a profile when a user logs on, surely there must be a way! And, well, there is! Have you ever noticed that file in your folder "NTUSERS.DAT"? Well, it turns out that file can be loaded into the registry. It stores all of your preference settings and a ton more! It's not a really well documented file structure, so I'm not going to pretend like I know what I'm talking about, *but*, it turns out that you can load this into the registry with the `reg load` and `reg unload` command! So let's give it a shot.



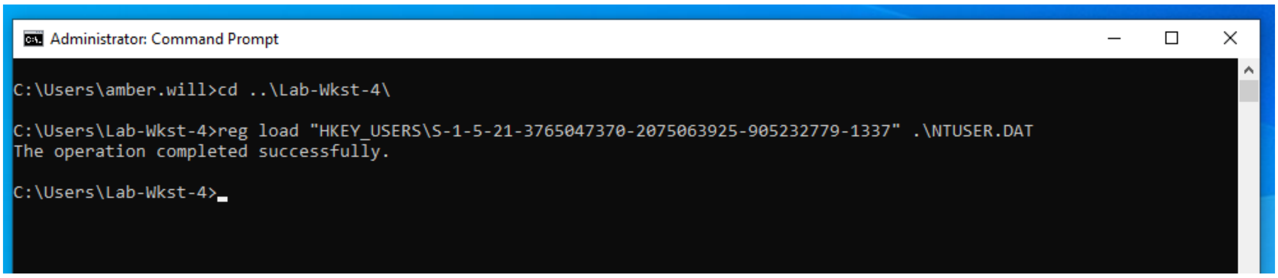
```
Microsoft Windows [Version 10.0.19042.1889]
(c) Microsoft Corporation. All rights reserved.

C:\Users\amber.will>reg load "HKEY_USERS\S-1-5-21-3765047370-2075063925-905232779-1337" .\NTUSER.DAT
ERROR: A required privilege is not held by the client.

C:\Users\amber.will>
```

So close! Lets try running this as an Administrator…



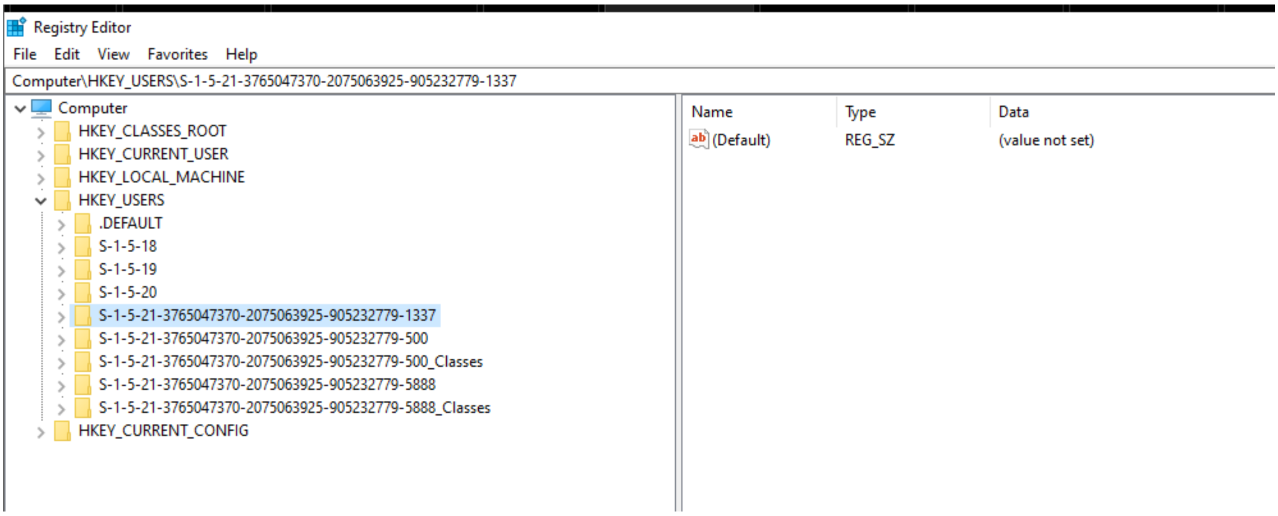
```
C:\WINDOWS\system32>cd C:\Users\amber.will

C:\Users\amber.will>reg load "HKEY_USERS\S-1-5-21-3765047370-2075063925-905232779-1337" .\NTUSER.DAT
ERROR: The process cannot access the file because it is being used by another process.

C:\Users\amber.will>
```

Well thats certainly a different error! Let’s try loading an *inactive* NTUSER.DAT file from another users account….



```
C:\Users\amber.will>cd ..\Lab-Wkst-4\

C:\Users\Lab-Wkst-4>reg load "HKEY_USERS\S-1-5-21-3765047370-2075063925-905232779-1337" .\NTUSER.DAT
The operation completed successfully.

C:\Users\Lab-Wkst-4>_
```

Well, it appears its loaded! Lets check regedit.



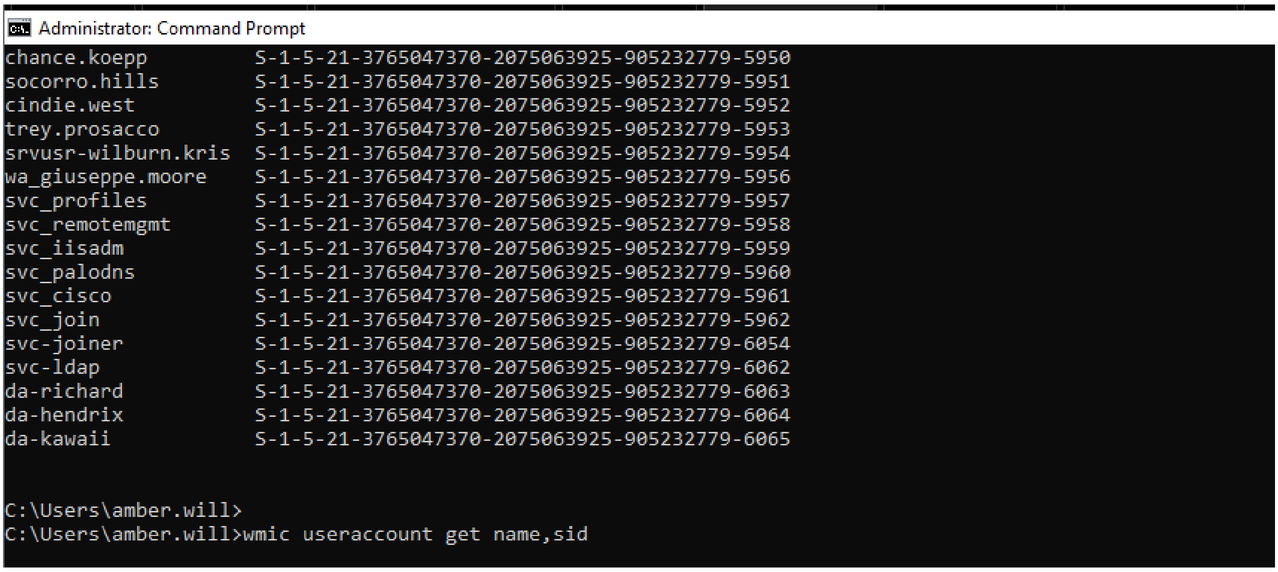
Computer\HKEY_USERS\S-1-5-21-3765047370-2075063925-905232779-1337

```
Computer
  HKEY_CLASSES_ROOT
  HKEY_CURRENT_USER
  HKEY_LOCAL_MACHINE
  HKEY_USERS
    .DEFAULT
    S-1-5-18
    S-1-5-19
    S-1-5-20
    S-1-5-21-3765047370-2075063925-905232779-1337
    S-1-5-21-3765047370-2075063925-905232779-500
    S-1-5-21-3765047370-2075063925-905232779-500_Classes
    S-1-5-21-3765047370-2075063925-905232779-5888
    S-1-5-21-3765047370-2075063925-905232779-5888_Classes
  HKEY_CURRENT_CONFIG
```

| Name | Type | Data |
|---|---|---|
| ab (Default) | REG_SZ | (value not set) |

## Great Success - Demo Time

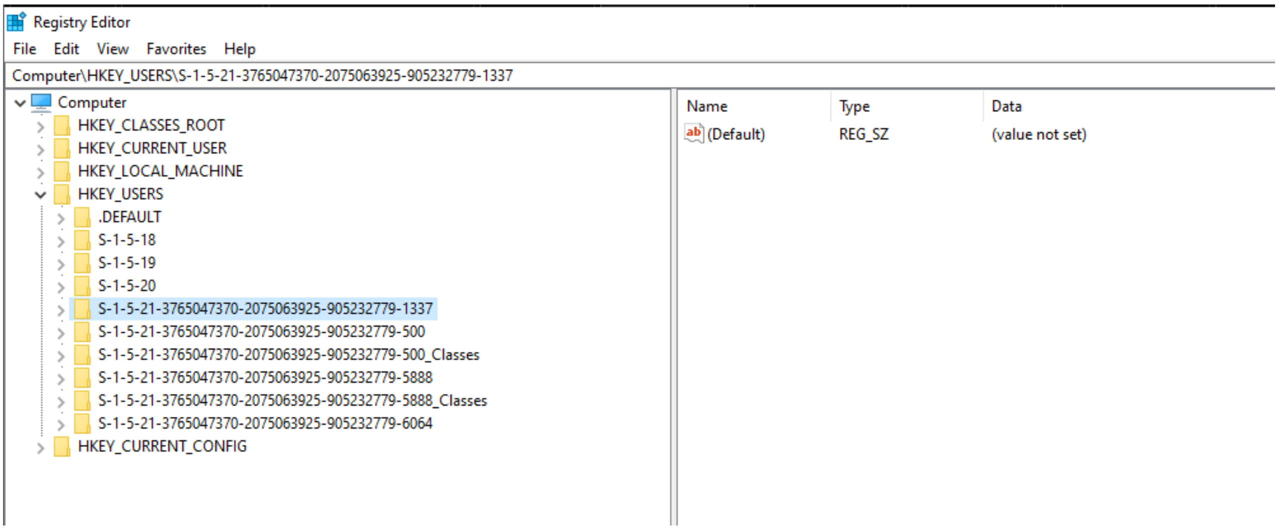
There it is! Our custom SID. Now that we have a working POC for loading SIDs, let’s select a *real* Domain Admin SID and start the Remote Registry service. To dump all the sids in the Domain, we can use `wmic useraccount get name,sid`.



```
chance.koepp         S-1-5-21-3765047370-2075063925-905232779-5950
socorro.hills        S-1-5-21-3765047370-2075063925-905232779-5951
cindie.west          S-1-5-21-3765047370-2075063925-905232779-5952
trey.prosacco        S-1-5-21-3765047370-2075063925-905232779-5953
srvusr-wilburn.kris  S-1-5-21-3765047370-2075063925-905232779-5954
wa_giuseppe.moore    S-1-5-21-3765047370-2075063925-905232779-5956
svc_profiles         S-1-5-21-3765047370-2075063925-905232779-5957
svc_remotemgmt       S-1-5-21-3765047370-2075063925-905232779-5958
svc_iisadm           S-1-5-21-3765047370-2075063925-905232779-5959
svc_palodns          S-1-5-21-3765047370-2075063925-905232779-5960
svc_cisco            S-1-5-21-3765047370-2075063925-905232779-5961
svc_join             S-1-5-21-3765047370-2075063925-905232779-5962
svc-joiner           S-1-5-21-3765047370-2075063925-905232779-6054
svc-ldap             S-1-5-21-3765047370-2075063925-905232779-6062
da-richard           S-1-5-21-3765047370-2075063925-905232779-6063
da-hendrix           S-1-5-21-3765047370-2075063925-905232779-6064
da-kawaii            S-1-5-21-3765047370-2075063925-905232779-6065


C:\Users\amber.will>
C:\Users\amber.will>wmic useraccount get name,sid
```

Afer executing it, we can see there are some new Domain Admins on the block! Let's simulate a session for *da-richard* on this workstation. To do so, we must copy their SID (S-1-5-21-3765047370-2075063925-905232779-6064) and load the NTUSERS.DAT registry hive. We can do so with the following command:

```
reg load HKU\S-1-5-21-3765047370-2075063925-905232779-6064 .\NTUSER.DAT
```

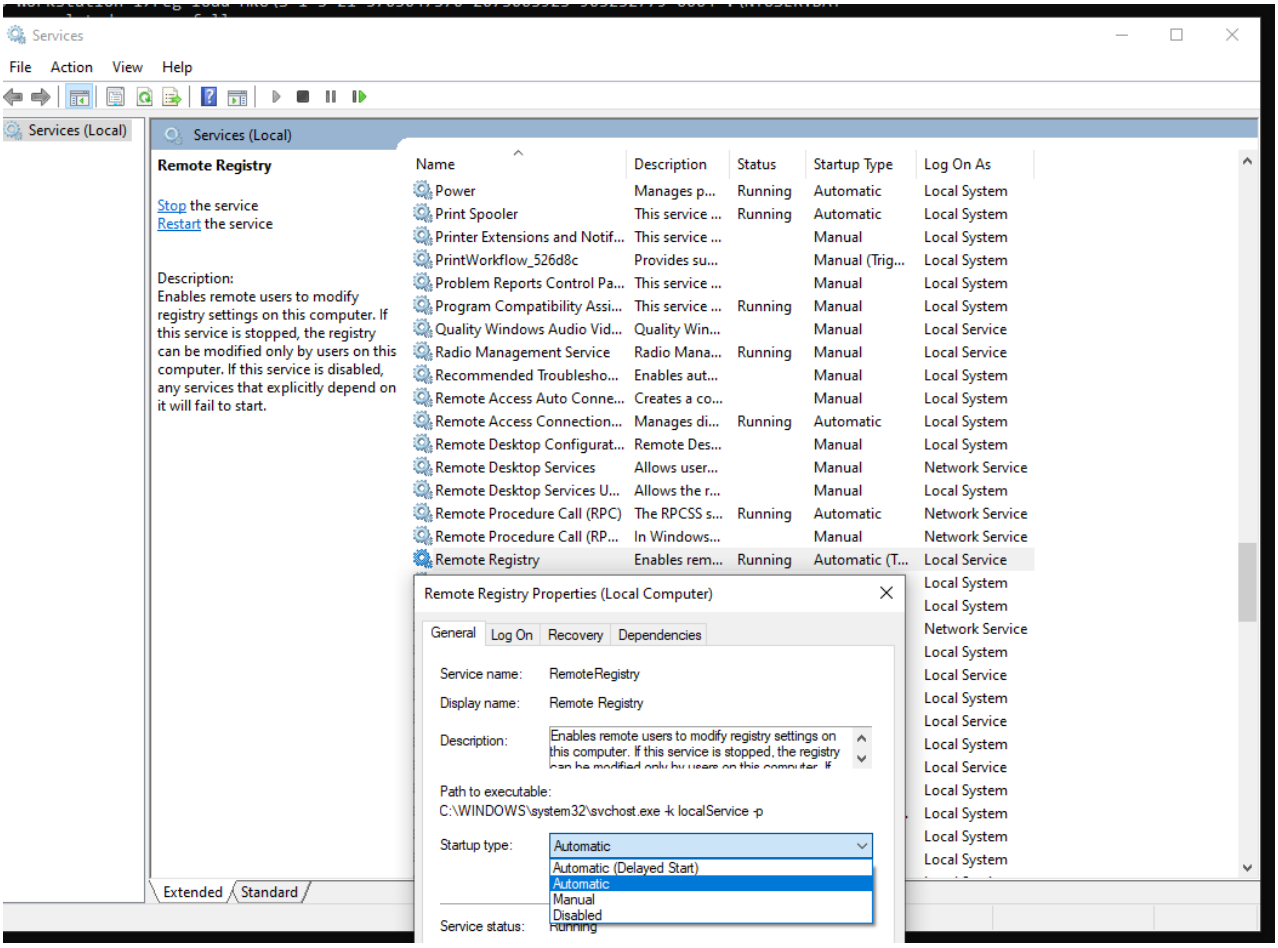


Computer\HKEY_USERS\S-1-5-21-3765047370-2075063925-905232779-1337

```
Computer
  HKEY_CLASSES_ROOT
  HKEY_CURRENT_USER
  HKEY_LOCAL_MACHINE
  HKEY_USERS
    .DEFAULT
    S-1-5-18
    S-1-5-19
    S-1-5-20
    S-1-5-21-3765047370-2075063925-905232779-1337
    S-1-5-21-3765047370-2075063925-905232779-500
    S-1-5-21-3765047370-2075063925-905232779-500_Classes
    S-1-5-21-3765047370-2075063925-905232779-5888
    S-1-5-21-3765047370-2075063925-905232779-5888_Classes
    S-1-5-21-3765047370-2075063925-905232779-6064
  HKEY_CURRENT_CONFIG
```

| Name | Type | Data |
|---|---|---|
| (Default) | REG_SZ | (value not set) |

And there it is, da-richard's SID loaded into the registry. Let's start the Remote Registry service through `services.msc`.



**Remote Registry**

Stop the service
Restart the service

Description:
Enables remote users to modify registry settings on this computer. If this service is stopped, the registry can be modified only by users on this computer. If this service is disabled, any services that explicitly depend on it will fail to start.

| Name | Description | Status | Startup Type | Log On As |
|---|---|---|---|---|
| Power | Manages p... | Running | Automatic | Local System |
| Print Spooler | This service ... | Running | Automatic | Local System |
| Printer Extensions and Notif... | This service ... | | Manual | Local System |
| PrintWorkflow_526d8c | Provides su... | | Manual (Trig... | Local System |
| Problem Reports Control Pa... | This service ... | | Manual | Local System |
| Program Compatibility Assi... | This service ... | Running | Manual | Local System |
| Quality Windows Audio Vid... | Quality Win... | | Manual | Local Service |
| Radio Management Service | Radio Mana... | Running | Manual | Local Service |
| Recommended Troublesho... | Enables aut... | | Manual | Local System |
| Remote Access Auto Conne... | Creates a co... | | Manual | Local System |
| Remote Access Connection... | Manages di... | Running | Automatic | Local System |
| Remote Desktop Configurat... | Remote Des... | | Manual | Local System |
| Remote Desktop Services | Allows user... | | Manual | Network Service |
| Remote Desktop Services U... | Allows the r... | | Manual | Local System |
| Remote Procedure Call (RPC) | The RPCSS s... | Running | Automatic | Network Service |
| Remote Procedure Call (RP... | In Windows... | | Manual | Network Service |
| Remote Registry | Enables rem... | Running | Automatic (T... | Local Service |
| | | | | Local System |
| | | | | Local System |
| | | | | Network Service |
| | | | | Local System |
| | | | | Local Service |
| | | | | Local System |
| | | | | Local Service |
| | | | | Local System |
| | | | | Local Service |
| | | | | Local System |
| | | | | Local System |
| | | | | Local System |
| | | | | Local System |

Remote Registry Properties (Local Computer)

Service name: RemoteRegistry

Display name: Remote Registry

Description: Enables remote users to modify registry settings on this computer. If this service is stopped, the registry can be modified only by users on this computer. If

Path to executable:
C:\WINDOWS\system32\svchost.exe -k localService -p

Startup type: Automatic

Service status: Running

Ensure that the service start type is set to Automatic and is started. Once this is complete, the Remote Registry service should be running. One last thing worth noting - The Remote Registry service will stop if there is no activity on the workstation for some time, there is a registry key that must be updated for it to run 24/7.

**Key Name:** HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\RemoteRegistry

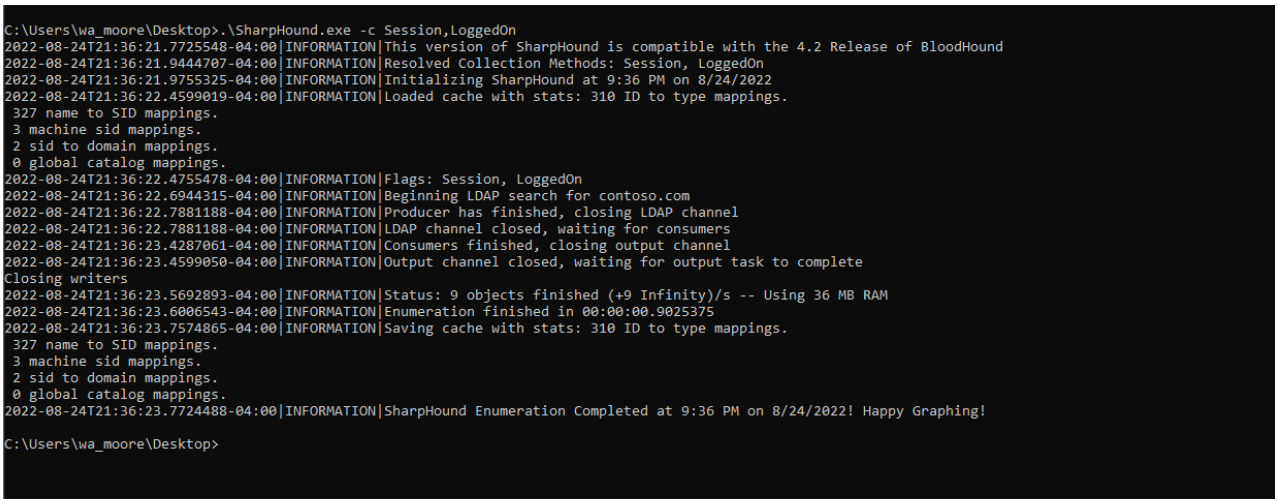
**Value:** DisableIdleStop

**REG_DWORD:** 1

One last thing **make sure you disable the Firewall**. By default, the Remote Registry service is filtered and you may not be able to see your session!

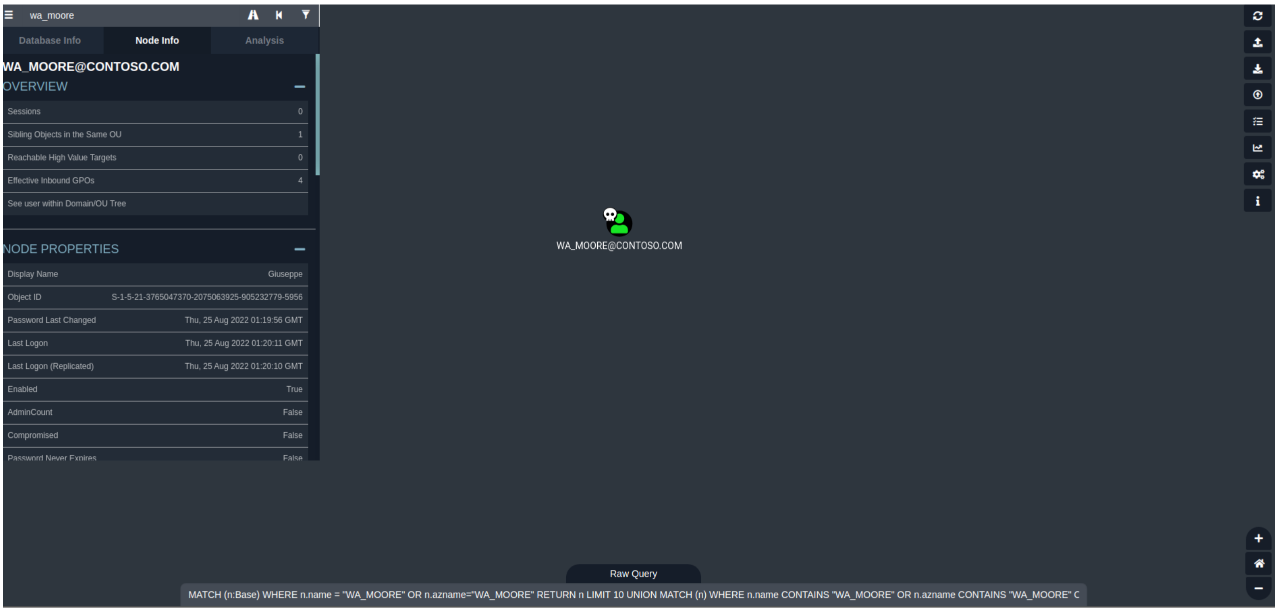
## Running SharpHound

Now that we have our deception setup, let's pretend that an attacker has compromised a Workstation Admin account and has the ability to collect session data, so in theory, if this deception object was deployed on a production system, an adversary would be none the wiser.



```
C:\Users\wa_moore\Desktop>.\SharpHound.exe -c Session,LoggedOn
2022-08-24T21:36:21.7725548-04:00|INFORMATION|This version of SharpHound is compatible with the 4.2 Release of BloodHound
2022-08-24T21:36:21.9444707-04:00|INFORMATION|Resolved Collection Methods: Session, LoggedOn
2022-08-24T21:36:21.9755325-04:00|INFORMATION|Initializing SharpHound at 9:36 PM on 8/24/2022
2022-08-24T21:36:22.4599019-04:00|INFORMATION|Loaded cache with stats: 310 ID to type mappings.
 327 name to SID mappings.
 3 machine sid mappings.
 2 sid to domain mappings.
 0 global catalog mappings.
2022-08-24T21:36:22.4755478-04:00|INFORMATION|Flags: Session, LoggedOn
2022-08-24T21:36:22.6944315-04:00|INFORMATION|Beginning LDAP search for contoso.com
2022-08-24T21:36:22.7881188-04:00|INFORMATION|Producer has finished, closing LDAP channel
2022-08-24T21:36:22.7881188-04:00|INFORMATION|LDAP channel closed, waiting for consumers
2022-08-24T21:36:23.4287061-04:00|INFORMATION|Consumers finished, closing output channel
2022-08-24T21:36:23.4599050-04:00|INFORMATION|Output channel closed, waiting for output task to complete
Closing writers
2022-08-24T21:36:23.5692893-04:00|INFORMATION|Status: 9 objects finished (+9 Infinity)/s -- Using 36 MB RAM
2022-08-24T21:36:23.6006543-04:00|INFORMATION|Enumeration finished in 00:00:00.9025375
2022-08-24T21:36:23.7574865-04:00|INFORMATION|Saving cache with stats: 310 ID to type mappings.
 327 name to SID mappings.
 3 machine sid mappings.
 2 sid to domain mappings.
 0 global catalog mappings.
2022-08-24T21:36:23.7724488-04:00|INFORMATION|SharpHound Enumeration Completed at 9:36 PM on 8/24/2022! Happy Graphing!

C:\Users\wa_moore\Desktop>
```

Next the adversary would load the Data into BloodHound and find their according user and mark them as an Owned principal.



WA_MOORE@CONTOSO.COM

OVERVIEW

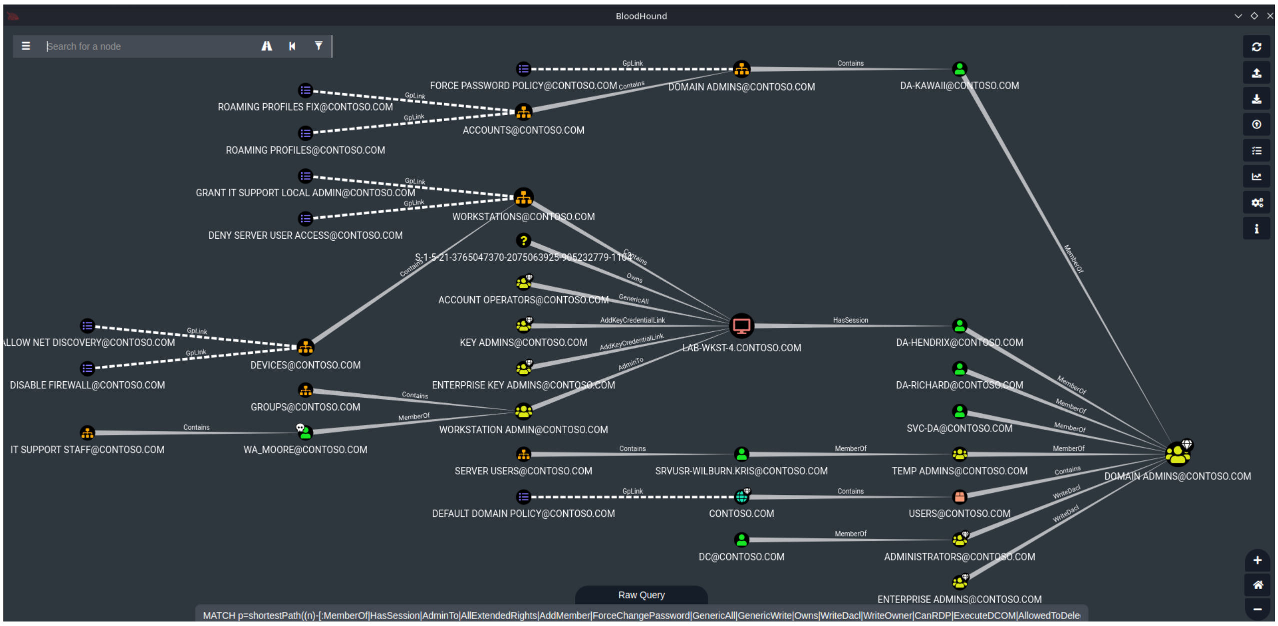
| | |
|---|---|
| Sessions | 0 |
| Sibling Objects in the Same OU | 1 |
| Reachable High Value Targets | 0 |
| Effective Inbound GPOs | 4 |
| See user within Domain/OU Tree | |

NODE PROPERTIES

| | |
|---|---|
| Display Name | Giuseppe |
| Object ID | S-1-5-21-3765047370-2075063925-905232779-5956 |
| Password Last Changed | Thu, 25 Aug 2022 01:19:56 GMT |
| Last Logon | Thu, 25 Aug 2022 01:20:11 GMT |
| Last Logon (Replicated) | Thu, 25 Aug 2022 01:20:10 GMT |
| Enabled | True |
| AdminCount | False |
| Compromised | False |
| Password Never Expires | False |

```
MATCH (n:Base) WHERE n.name = "WA_MOORE" OR n.azname="WA_MOORE" RETURN n LIMIT 10 UNION MATCH (n) WHERE n.name CONTAINS "WA_MOORE" OR n.azname CONTAINS "WA_MOORE" C
```

Afterwards, they would then search for paths to Domain Admin (more specifically, the shortest path). After executing this query in a relatively clean environment (i.e. No Domain Admins within closer reach) you should see a very short path to DA-Hendrix!


BloodHound
Search for a node
FORCE PASSWORD POLICY@CONTOSO.COM
DOMAIN ADMINS@CONTOSO.COM
DA-KAWAII@CONTOSO.COM
ROAMING PROFILES FIX@CONTOSO.COM
ACCOUNTS@CONTOSO.COM
ROAMING PROFILES@CONTOSO.COM
GRANT IT SUPPORT LOCAL ADMIN@CONTOSO.COM
WORKSTATIONS@CONTOSO.COM
DENY SERVER USER ACCESS@CONTOSO.COM
S-1-5-21-3765047370-2075063925-905232779-1104
ACCOUNT OPERATORS@CONTOSO.COM
ALLOW NET DISCOVERY@CONTOSO.COM
KEY ADMINS@CONTOSO.COM
LAB-WKST-4.CONTOSO.COM
DA-HENDRIX@CONTOSO.COM
DISABLE FIREWALL@CONTOSO.COM
DEVICES@CONTOSO.COM
ENTERPRISE KEY ADMINS@CONTOSO.COM
DA-RICHARD@CONTOSO.COM
GROUPS@CONTOSO.COM
SVC-DA@CONTOSO.COM
IT SUPPORT STAFF@CONTOSO.COM
WA_MOORE@CONTOSO.COM
WORKSTATION ADMIN@CONTOSO.COM
SERVER USERS@CONTOSO.COM
SRVUSR-WILBURN.KRIS@CONTOSO.COM
TEMP ADMINS@CONTOSO.COM
DOMAIN ADMINS@CONTOSO.COM
DEFAULT DOMAIN POLICY@CONTOSO.COM
CONTOSO.COM
USERS@CONTOSO.COM
DC@CONTOSO.COM
ADMINISTRATORS@CONTOSO.COM
ENTERPRISE ADMINS@CONTOSO.COM
GpLink
Contains
MemberOf
HasSession
Owns
GenericAll
AddKeyCredentialLink
AdminTo
WriteDacl
Raw Query

```
MATCH p=shortestPath((n)-[:MemberOf|HasSession|AdminTo|AllExtendedRights|AddMember|ForceChangePassword|GenericAll|GenericWrite|Owns|WriteDacl|WriteOwner|CanRDP|ExecuteDCOM|AllowedToDele
```

We can see that our method injected an artifical path to Domain Admin. On *Lab-Wkst-4*, there is a Session for the user *da-hendrix*. We can see that our user (WA_MOORE) is a member of the Workstation Admins group, which is an Administrator on that workstation. This means that Domain Admin *appears* to be one hop away.

## Introducing Honey Sessions

Overall, this is a fairly simple method. There's a lot of room for optimization (in my opinion). So myself and @LIKEROFJAZZ have done that. Over the past month or so, we have written a Python script (compiled with PyInstaller) to automatically drop a NTUSER.DAT file to disk and automatically pick a random Domain Admin session to inject.

So… Demo time, again! This time we'll be running BloodHound-Py for the sake of simplicity.

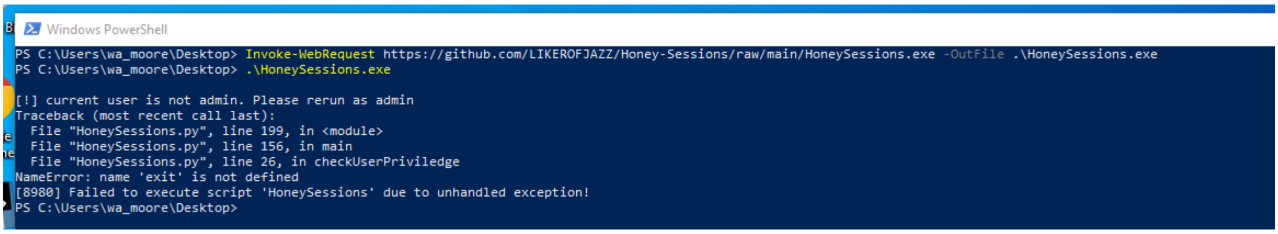
### Setup

Little to no setup is required, though there are a few constraints:

1. The user must be an Administrator on the Workstation
2. The user must be a domain user (unless you specify a SID)

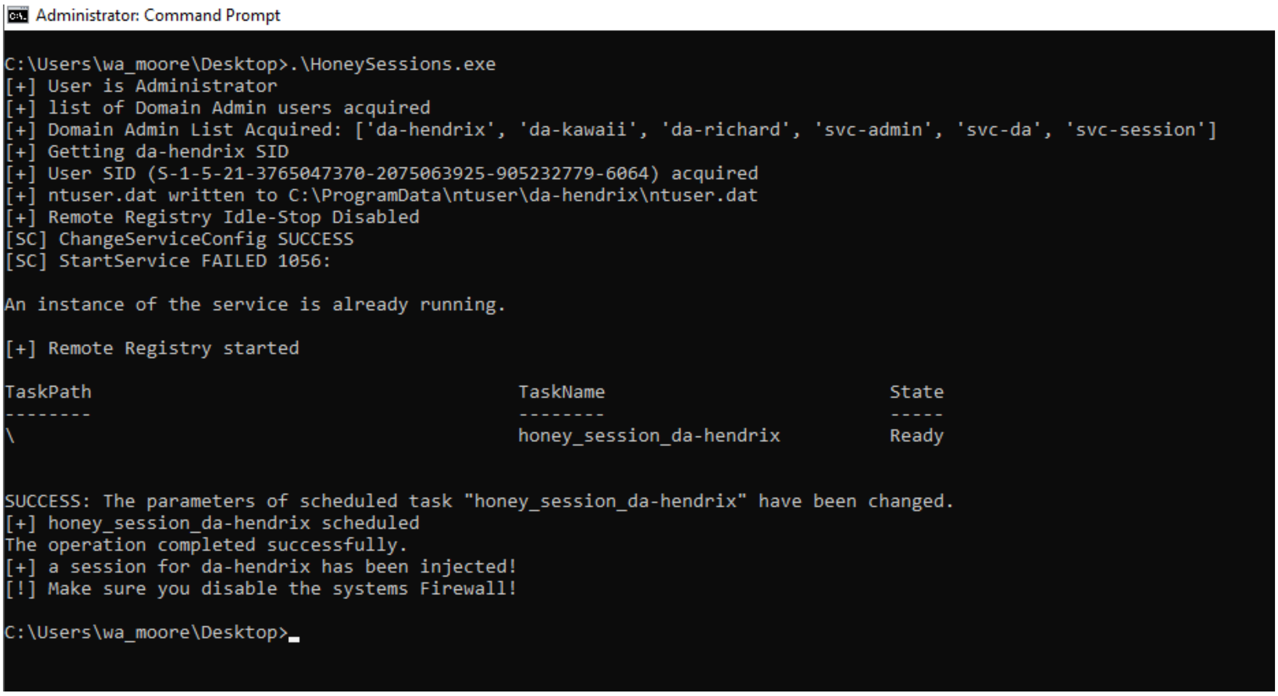
Simple enough! Now we'll head over to Honey-Sessions on GitHub and download the pre-compiled PyInstaller executable. If you have trust issues, the source code is provided as-is. You may feel comfortable substituting in your own `NTUSER.DAT` file as well. We promise it's nothing malicious. Just trying to make everyones life easier :D

Anyways - after downloading HoneySessions.exe to the Workstation, the next step is super straight forward. Run the binary as an Administrator.



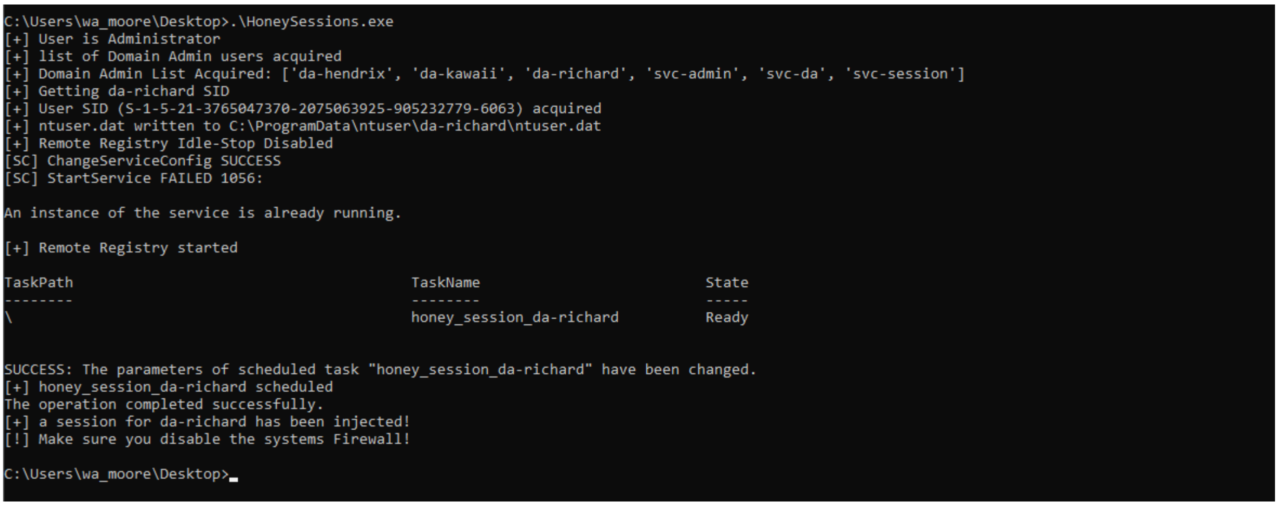
```
PS C:\Users\wa_moore\Desktop> Invoke-WebRequest https://github.com/LIKEROFJAZZ/Honey-Sessions/raw/main/HoneySessions.exe -OutFile .\HoneySessions.exe
PS C:\Users\wa_moore\Desktop> .\HoneySessions.exe

[!] current user is not admin. Please rerun as admin
Traceback (most recent call last):
  File "HoneySessions.py", line 199, in <module>
  File "HoneySessions.py", line 156, in main
  File "HoneySessions.py", line 26, in checkUserPriviledge
NameError: name 'exit' is not defined
[8980] Failed to execute script 'HoneySessions' due to unhandled exception!
PS C:\Users\wa_moore\Desktop>
```

Don’t be silly like me and try to not run it on a medium integrity level, haha. Attempt #2 - Let’s try this again.



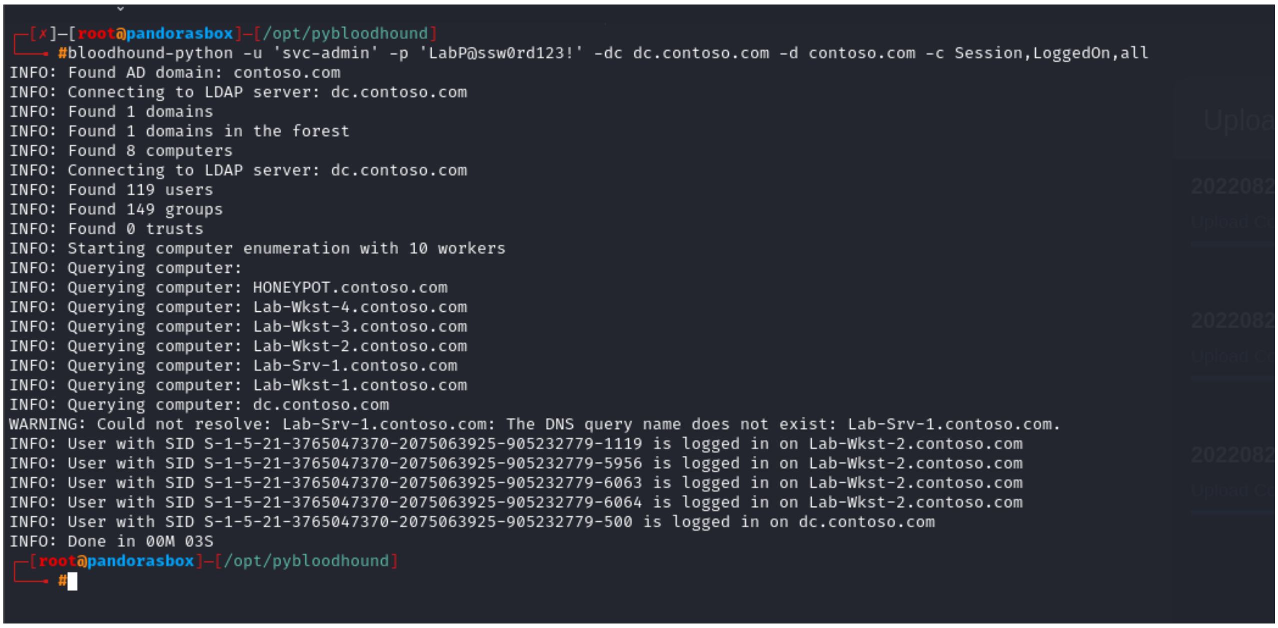
```
C:\Users\wa_moore\Desktop>.\HoneySessions.exe
[+] User is Administrator
[+] list of Domain Admin users acquired
[+] Domain Admin List Acquired: ['da-hendrix', 'da-kawaii', 'da-richard', 'svc-admin', 'svc-da', 'svc-session']
[+] Getting da-hendrix SID
[+] User SID (S-1-5-21-3765047370-2075063925-905232779-6064) acquired
[+] ntuser.dat written to C:\ProgramData\ntuser\da-hendrix\ntuser.dat
[+] Remote Registry Idle-Stop Disabled
[SC] ChangeServiceConfig SUCCESS
[SC] StartService FAILED 1056:

An instance of the service is already running.

[+] Remote Registry started

TaskPath                                       TaskName                          State
--------                                       --------                          -----
\                                              honey_session_da-hendrix          Ready


SUCCESS: The parameters of scheduled task "honey_session_da-hendrix" have been changed.
[+] honey_session_da-hendrix scheduled
The operation completed successfully.
[+] a session for da-hendrix has been injected!
[!] Make sure you disable the systems Firewall!

C:\Users\wa_moore\Desktop>_
```

This time it actually worked, though it randomly selected da-hendrix again… Let’s re-run it and see who we get this time.



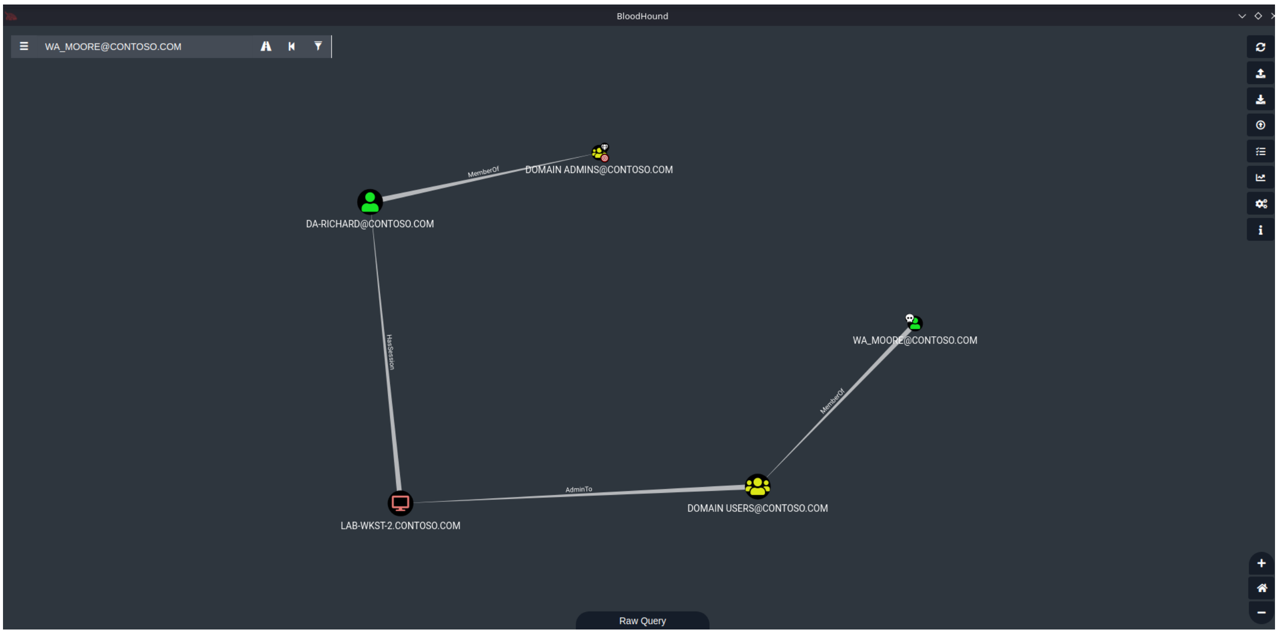
```
C:\Users\wa_moore\Desktop>.\HoneySessions.exe
[+] User is Administrator
[+] list of Domain Admin users acquired
[+] Domain Admin List Acquired: ['da-hendrix', 'da-kawaii', 'da-richard', 'svc-admin', 'svc-da', 'svc-session']
[+] Getting da-richard SID
[+] User SID (S-1-5-21-3765047370-2075063925-905232779-6063) acquired
[+] ntuser.dat written to C:\ProgramData\ntuser\da-richard\ntuser.dat
[+] Remote Registry Idle-Stop Disabled
[SC] ChangeServiceConfig SUCCESS
[SC] StartService FAILED 1056:

An instance of the service is already running.

[+] Remote Registry started

TaskPath                                       TaskName                          State
--------                                       --------                          -----
\                                              honey_session_da-richard          Ready


SUCCESS: The parameters of scheduled task "honey_session_da-richard" have been changed.
[+] honey_session_da-richard scheduled
The operation completed successfully.
[+] a session for da-richard has been injected!
[!] Make sure you disable the systems Firewall!

C:\Users\wa_moore\Desktop>_
```

da-richard, much better! Let’s run BloodHound.py and load the data into BloodHound!



```
┌─[✗]─[root@pandorasbox]─[/opt/pybloodhound]
└──╼ #bloodhound-python -u 'svc-admin' -p 'LabP@ssw0rd123!' -dc dc.contoso.com -d contoso.com -c Session,LoggedOn,all
INFO: Found AD domain: contoso.com
INFO: Connecting to LDAP server: dc.contoso.com
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 8 computers
INFO: Connecting to LDAP server: dc.contoso.com
INFO: Found 119 users
INFO: Found 149 groups
INFO: Found 0 trusts
INFO: Starting computer enumeration with 10 workers
INFO: Querying computer:
INFO: Querying computer: HONEYPOT.contoso.com
INFO: Querying computer: Lab-Wkst-4.contoso.com
INFO: Querying computer: Lab-Wkst-3.contoso.com
INFO: Querying computer: Lab-Wkst-2.contoso.com
INFO: Querying computer: Lab-Srv-1.contoso.com
INFO: Querying computer: Lab-Wkst-1.contoso.com
INFO: Querying computer: dc.contoso.com
WARNING: Could not resolve: Lab-Srv-1.contoso.com: The DNS query name does not exist: Lab-Srv-1.contoso.com.
INFO: User with SID S-1-5-21-3765047370-2075063925-905232779-1119 is logged in on Lab-Wkst-2.contoso.com
INFO: User with SID S-1-5-21-3765047370-2075063925-905232779-5956 is logged in on Lab-Wkst-2.contoso.com
INFO: User with SID S-1-5-21-3765047370-2075063925-905232779-6063 is logged in on Lab-Wkst-2.contoso.com
INFO: User with SID S-1-5-21-3765047370-2075063925-905232779-6064 is logged in on Lab-Wkst-2.contoso.com
INFO: User with SID S-1-5-21-3765047370-2075063925-905232779-500 is logged in on dc.contoso.com
INFO: Done in 00M 03S
┌─[root@pandorasbox]─[/opt/pybloodhound]
└──╼ #
```

We can see immediately that theres a handful of user sessions on Lab-Wkst-2, good news! Let's go check out who they are. As always - Make sure you mark the user as owned and then search for shortest path to Domain Admins!



And there it is. Plug-n-Play Deception. Anyways - I hope you all enjoyed this entry in the Deception in Depth series. I have absolutely no idea when the next post will come out. I don't have anything hidden up my sleeve, but that doesn't mean I'll stop being passionate about deception. If you'd like to chat with me about Deception, feel free to reach out to me on Twitter or LinkedIn. I'd love to discuss new ideas, help develop and flesh out new methods, or even give some advice or share some experiences if Deception is something you're interested in bringing to your organization. As always - Thank you for all for the love and support <3 ~ Ronnie

## Comments