实验吧杂项-流量日志分析

原创

Au.J 于 2019-09-11 14:53:20 发布
分类专栏： CTF 文章标签： writeup

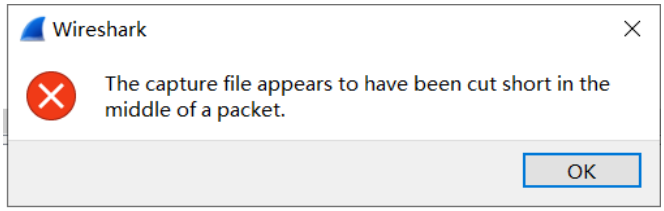
A记录

数据包

题目描述如下：

他在看什么视频，好像很好看，不知道是什么网站的。
还好我截取了他的数据包，找呀找。
key就是网站名称。格式ctf{key}
tip:A记录的第一条。

下载下来是一个cap文件，用wireshark打开，提示文件截断



观察流量发现为无线流量，加上之前的报错，猜测为已加密的数据包

```
localhost (00:1d:0f:5d:d0:e… 802.11
Tp-LinkT_d9:49:7e (5c:63:bf… 802.11
Apple_4c:2a:9a (98:fe:94:4c… 802.11
Apple_4c:2a:9a (98:fe:94:4c… 802.11
Tp-LinkT_d9:49:7e (5c:63:bf… 802.11
Tp-LinkT_d9:49:7e (5c:63:bf… 802.11
```

破解加密的无线数据包需要得到数据包的 ESSID 和 PASSWORD

使用Kali集成的 **aircrack-ng** 爆破工具得到ESSID为 **0719** 加密方式为 **WPA加密（脆弱）**

```
root@kali:~# aircrack-ng shipin.cap
Opening shipin.caplease wait...
Read 16664 packets.

   #  BSSID              ESSID                     Encryption

   1  00:1D:0F:5D:D0:EE  0719                      WPA (1 handshake)

Choosing first network as target.

Opening shipin.caplease wait...
Read 16664 packets.

1 potential targets

Please specify a dictionary (option -w).
```


加载字典进行爆破得到password

```
aircrack-ng shipin.cap -w ~/桌面/passwd.txt
```

```
root@kali:~# aircrack-ng shipin.cap -w ~/桌面/passwd.txt
```

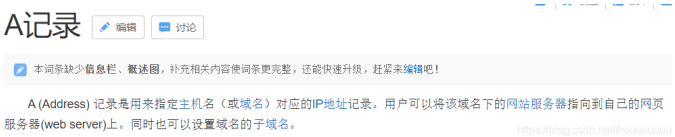
```
                    Aircrack-ng 1.5.2

[00:00:00] 8/17 keys tested (329.19 k/s)

Time left: 0 seconds                                      47.06%

                    KEY FOUND! [ 88888888 ]


Master Key     : B4 30 38 0F 24 7B 57 AC DE B5 3A 7F 2E FE 6B 45
                 0B 34 02 C3 89 F9 69 D5 B7 35 87 1B FB 4C EE 7F

Transient Key  : 57 F5 6A 04 36 F5 71 C8 52 D5 89 A0 7F 1A 20 D4
                 AF 0D 45 4A 3D BC E4 00 00 00 00 00 00 00 00 00
                 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC     : 75 19 C5 F3 3E 33 58 23 CA 4B A1 85 FB 46 C0 2A
```

使用airdecap-ng得到解密后的数据包 **shipin-dec.cap**

```
airdecap-ng shipin.cap -p 88888888 -e 0719
```

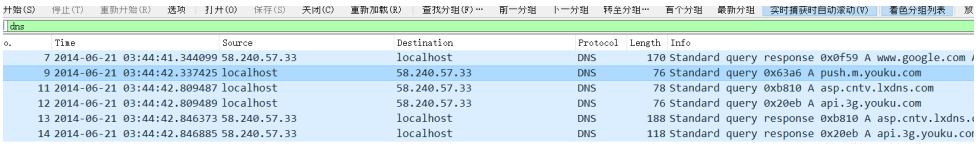
```
root@kali:~# airdecap-ng shipin.cap -p 88888888 -e 0719
Total number of stations seen            6
Total number of packets read         16664
Total number of WEP data packets         0
Total number of WPA data packets        27
Number of plaintext data packets         0
Number of decrypted WEP  packets         0
Number of corrupted WEP  packets         0
Number of decrypted WPA  packets        16
Number of bad TKIP (WPA) packets         0
Number of bad CCMP (WPA) packets         0
```


题目提示A记录的第一条，所以我们筛选查看DNS协议



通俗来说A记录就是服务器的IP，域名绑定A记录就是告诉DNS，当输入域名的时候给你引导向设置在DNS的A记录所对应的服务器。

如图，得到A记录的域名 **push.m.youku.com**



flag: ctf{push.m.youku.com}

抓到你了

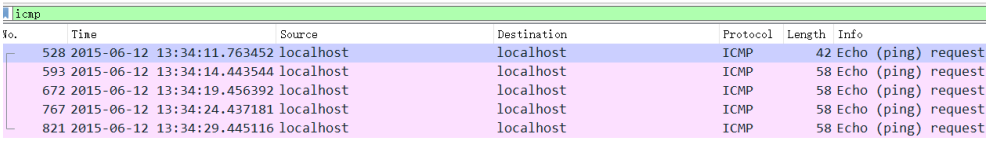
数据包

题目描述如下：

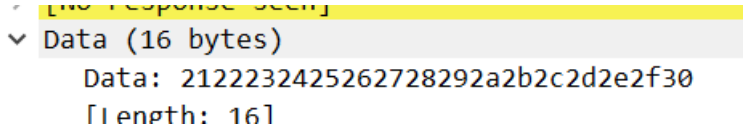
> Hint：入侵者通过 ping 工具对局域网内一主机进行存活性扫描， flag 为入侵所 发送的 16 字节的数据包内容。

下载文件没有后缀 编辑后缀为pcap，使用wireshark打开

提示ping探测存活主机，所以筛选ICMP协议数据包



查看分析可得




```
212223242526272829 2a2b2c2d2e2f30
```

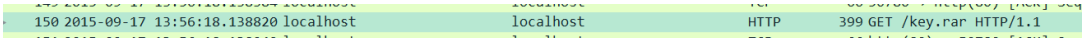
NSCTF misc250

数据包

题目描述如下：

> 小绿在学习了wireshark后，在局域网内抓到了室友下载的小东东0.0 你能帮他找到吗?
> 格式：flag{}

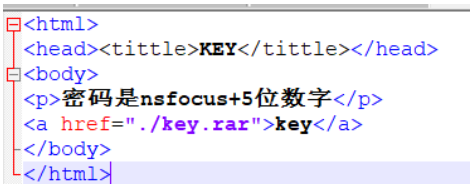
发现一个可疑的压缩包



使用wireshark，文件--导出对象--HTTP

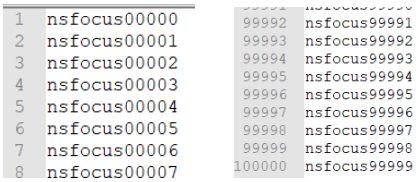
得到 %5c 和 一个 加密的压缩包 key.rar

%5c 记事本打开得到提示信息：密码是nsfocus+5位数字

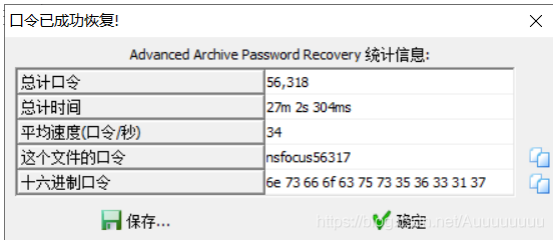


python 生成对应的字典

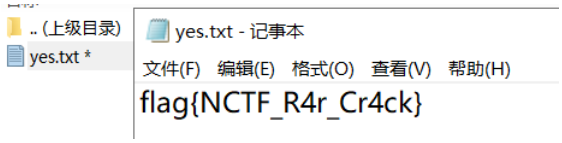
```
import string

string = "nsfocus"
with open("key.txt", "w") as f:
 for num in range(0, 100000):
  f.write(string+str(num).zfill(5)+'\n')
```



使用 ARCHPR 爆破得到 key **nsfocus56317**



压缩包得到flag **flag{NCTF_R4r_Cr4ck}**



这是捕获的黑客攻击数据包，Administrator用户的密码在此次攻击中泄露了，你能找到吗？

数据包

题目描述如下：

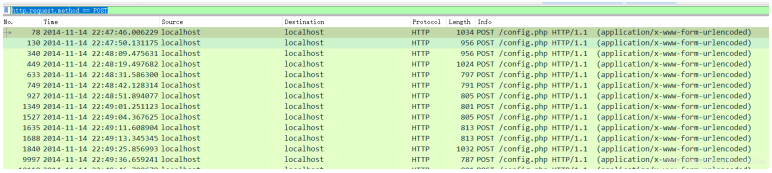
FLAG为Administrator的明文密码

下载得到一个压缩包，解压得到数据包，扔到wireshark中

数据包略大，尝试HTTP过滤，大概查看数据包内容，发现了熟悉的eval函数，猜测 数据包通过一句话连接菜刀进行攻击

继续过滤为 POST请求 **http.request.method == POST**

得到如下数据



逐条查看，数据包内容均为base64编码，需解码

最后在 449 数据包中发现密码 **Test!@#123**

内网攻击数据包，请分析

数据包

题目描述如下：

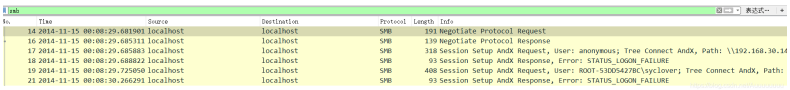
key值为syclover明文密码

下载得到一个压缩包，解压得到数据包，扔到wireshark中

数据包不多，发现smb数据包，猜测为**内网的smb劫持攻击数据流量**，smb包里面的challenge值确实也是1122334455667788

```
Challenge: 1122334455667788
```

过滤得到smb协议数据包



smb协议：https://bbs.pediy.com/thread-176189.htm

得到

LMHASH:9e94258a03356914b15929fa1d2e290fab9c8f9f01999448

NTHASH:013f3cb06ba848f98a6ae6cb4a76477c5ba4e45cda73b475

```
Byte Count (BCC): 231
ANSI Password: 9e94258a03356914b15929fa1d2e290fab9c8f9f01999448
Unicode Password: 013f3cb06ba848f98a6ae6cb4a76477c5ba4e45cda73b475
Account: svclover
```

ANSI Password的生成过程是在syclover用户长度为16字节的LM Hash后补充5字节的0x00，得到21字节的Hash，再分成3组，每组7字节。将每组的7字节经过str_to_key函数处理成8字节的DES Key，对服务器返回的challenge进行标准DES加密，得到3组8字节的密文，最后将加密后的密文拼接在一起。Unicode Password生成过程类似，只不过DES密钥是经过NTLM Hash处理后得到的。

而根据LM Hash的生成过程，LM Hash的前8字节是将用户明文密码的前7位转为大写，再经过str_to_ket和标准DES加密处理得到的。

综上，ANSI Password的前8字节与用户明文密码的前7位的大写形式以及服务器响应的challenge有关。
本例中，challenge为0x1122334455667788，因此可以通过challenge为0x1122334455667788的HALFLM彩虹表来查出syclover用户**明文密码的前7位大写形式**。

彩虹表　　rcracki

得到　**NETLMIS**

可以使用 hashcat 爆破得到　**NetLMis666**

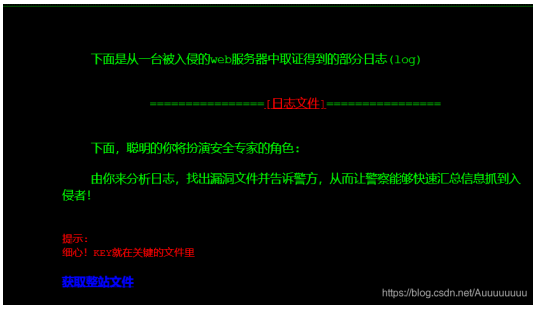
你有记日志的习惯吗

题目链接

题目描述如下：

> 某一年的某一天，某台服务器被入侵了，你吓坏了

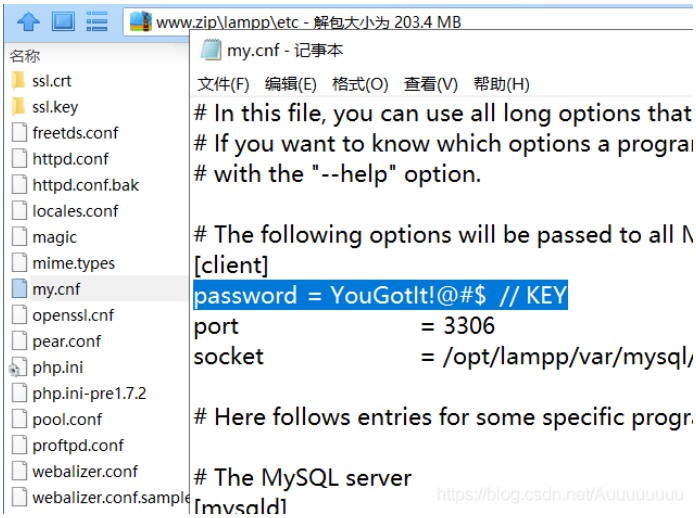
打开对应网址，可查得到日志文件和网站源码



日志文件是常规的爆破扫描，未发现有效信息

查看网站源码，题目提示记日志，所以重点放在配置文件、log上查看

最终在mysql的配置文件 my.cnf中发现key　**YouGotIt!@#$**



2015RCTF（misc50）

解题链接

题目描述如下：

> Analysis nginx's log(this flag is like ROIS{xxx})

下载得到 log_log

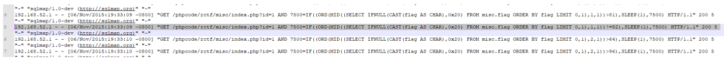
修改后缀为.zip 得到真正的log文件

简单查看发现是sqlmap的爆破记录


"sqlmap/1.0-dev (http://sqlmap.org)" "-"
192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%28
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"
192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%28
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"
192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%28
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"
192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%28
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"
192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%28
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"

直接搜索flag查看，提取对应爆破操作，解码 发现sqlmap正在通过二分法爆flag表的flag字段内容（用单个字符的ASCII码通过大小写判断）

直接提取爆破得到的每个字符



提取得到每个字符对应的ASCII码

```
[82,79,73,83,123,109,105,83,99,95,65,110,64,108,121,83,105,115,95,110,71,49,110,120,95,83,105,109,125,5]
```

python 转换 得到 **ROIS{miSc_An@lySis_nG1nx_Sim}**

```
list =[82,79,73,83,123,109,105,83,99,95,65,110,64,108,121,83,105,115,95,110,71,49,110,120,95,83,105,109,125
str1 = ""
for i in list:
 str1 += chr(i)

print(str1)
```

写的时候实验吧直接维护了.... 还有一道流量题，有待补充