

实验吧杂项-流量日志分析

原创

AuJ 于 2019-09-11 14:53:20 发布 643 收藏 3

分类专栏: [CTF](#) 文章标签: [writeup](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/Auuuuuuu/article/details/100699620>

版权



[CTF 专栏收录该内容](#)

5 篇文章 0 订阅

订阅专栏

A记录

数据包

题目描述如下:

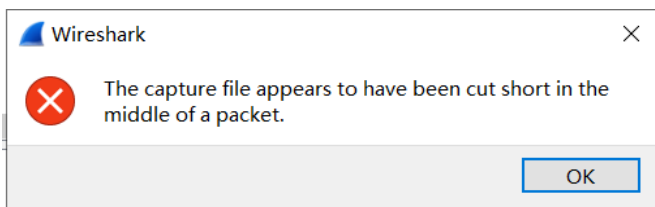
他在看什么视频, 好像很好看, 不知道是什么网站的。

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

```
localn0st (00:10:0f:5d:a0: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.cap please 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.cap please wait...
Read 16664 packets.

1 potential targets
Please specify a dictionary (option -w).

https://blog.csdn.net/Auuuuuuuuu
```

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

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

https://blog.csdn.net/Auuuuuuuuu
```

题目提示A记录的第一条，所以我们筛选查看DNS协议

A记录

编辑 讨论

本词条缺少信息栏、概述图，补充相关内容使词条更完整，还能快速升级，赶紧来编辑吧！

A (Address) 记录是用来指定主机名（或域名）对应的IP地址记录。用户可以将该域名下的网站服务器指向到自己的网页服务器(web server)上，同时也可以设置域名的子域名。

<https://blog.csdn.net/Auuuuuuuu>

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**

No.	Time	Source	Destination	Protocol	Length	Info
7	2014-06-21 03:44:41.344099	58.240.57.33	localhost	DNS	170	Standard query response 0x0f59 A www.google.com
9	2014-06-21 03:44:42.337425	localhost	58.240.57.33	DNS	76	Standard query 0x63a6 A push.m.youku.com
11	2014-06-21 03:44:42.809487	localhost	58.240.57.33	DNS	78	Standard query 0xb810 A asp.cntv.lxdns.com
12	2014-06-21 03:44:42.809489	localhost	58.240.57.33	DNS	76	Standard query 0x20eb A api.3g.youku.com
13	2014-06-21 03:44:42.846373	58.240.57.33	localhost	DNS	188	Standard query response 0xb810 A asp.cntv.lxdns.com
14	2014-06-21 03:44:42.846885	58.240.57.33	localhost	DNS	118	Standard query response 0x20eb A api.3g.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

No.	Time	Source	Destination	Protocol	Length	Info
528	2015-06-12 13:34:11.763452	localhost	localhost	ICMP	42	Echo (ping) request
593	2015-06-12 13:34:14.443544	localhost	localhost	ICMP	58	Echo (ping) request
672	2015-06-12 13:34:19.456392	localhost	localhost	ICMP	58	Echo (ping) request
767	2015-06-12 13:34:24.437181	localhost	localhost	ICMP	58	Echo (ping) request
821	2015-06-12 13:34:29.445116	localhost	localhost	ICMP	58	Echo (ping) request

查看分析可得

```
[No response seen]
Data (16 bytes)
  Data: 2122232425262728292a2b2c2d2e2f30
  Length: 16]
```

2122232425262728292a2b2c2d2e2f30

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

150	2015-09-17 13:56:18.138820	localhost	localhost	HTTP	399 GET /key.rar HTTP/1.1
-----	----------------------------	-----------	-----------	------	---------------------------

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

```
<html>
<head><tittle>KEY</tittle></head>
<body>
<p>密码是nsfocus+5位数字</p>
<a href="/key.rar">key</a>
</body>
</html>
```

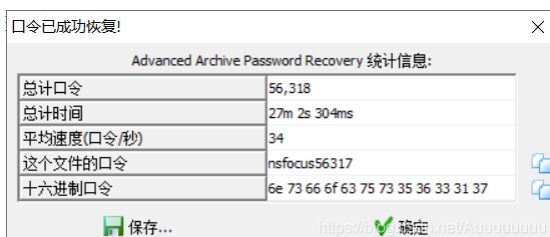
python 生成对应的字典

```
import string

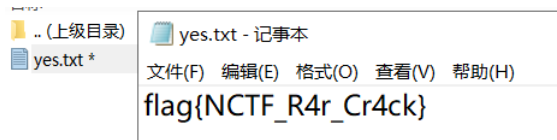
string = "nsfocus"
with open("key.txt", "w") as f:
    for num in range(0, 100000):
        f.write(string+str(num).zfill(5)+'\n')
```

1	nsfocus00000	99992	nsfocus99991
2	nsfocus00001	99993	nsfocus99992
3	nsfocus00002	99994	nsfocus99993
4	nsfocus00003	99995	nsfocus99994
5	nsfocus00004	99996	nsfocus99995
6	nsfocus00005	99997	nsfocus99996
7	nsfocus00006	99998	nsfocus99997
8	nsfocus00007	100000	nsfocus99999

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

No.	Time	Source	Destination	Protocol	Length	Info
78	2014-11-14 22:47:56.986293	localhost	localhost	HTTP	1034	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
130	2014-11-14 22:47:58.131175	localhost	localhost	HTTP	956	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
340	2014-11-14 22:48:09.475611	localhost	localhost	HTTP	956	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
449	2014-11-14 22:48:10.497682	localhost	localhost	HTTP	1024	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
613	2014-11-14 22:48:31.586388	localhost	localhost	HTTP	797	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
749	2014-11-14 22:48:42.128314	localhost	localhost	HTTP	791	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
927	2014-11-14 22:48:51.988077	localhost	localhost	HTTP	805	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
1340	2014-11-14 22:49:01.251123	localhost	localhost	HTTP	801	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
1527	2014-11-14 22:49:04.367625	localhost	localhost	HTTP	805	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
1676	2014-11-14 22:49:11.688084	localhost	localhost	HTTP	813	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
1888	2014-11-14 22:49:13.345345	localhost	localhost	HTTP	813	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
1840	2014-11-14 22:49:25.856993	localhost	localhost	HTTP	1032	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)
9997	2014-11-14 22:49:36.659241	localhost	localhost	HTTP	787	POST /config.php HTTP/1.1 (application/x-www-form-urlencoded)

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

```
Byte Count (BCC): 12  
Challenge: 1122334455667788
```

过滤得到smb协议数据包

No.	Time	Source	Destination	Protocol	Length	Info
14	2014-11-15 00:08:29.681983	localhost	localhost	SMB	374	Negotiate Protocol Request
16	2014-11-15 00:08:29.685311	localhost	localhost	SMB	139	Negotiate Protocol Response
17	2014-11-15 00:08:29.688583	localhost	localhost	SMB	318	Session Setup AndX Request, user: anonymous; Tree Connect AndX, Path: \\192.168.30.14
18	2014-11-15 00:08:29.688802	localhost	localhost	SMB	93	Session Setup AndX Response, Error: STATUS_LOGIN_FAILURE
19	2014-11-15 00:08:29.725650	localhost	localhost	SMB	488	Session Setup AndX Request, user: ROOT\S30254278C\Syclover; Tree Connect AndX, Path:
21	2014-11-15 00:08:30.266291	localhost	localhost	SMB	93	Session Setup AndX Response, Error: STATUS_LOGIN_FAILURE

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_key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**

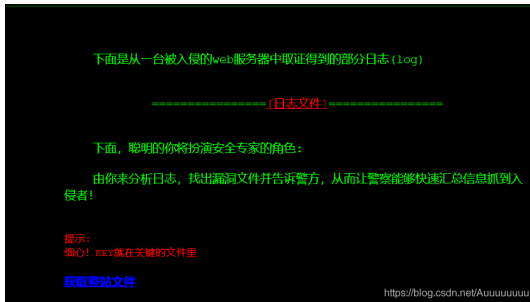
你有记日志的习惯吗

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

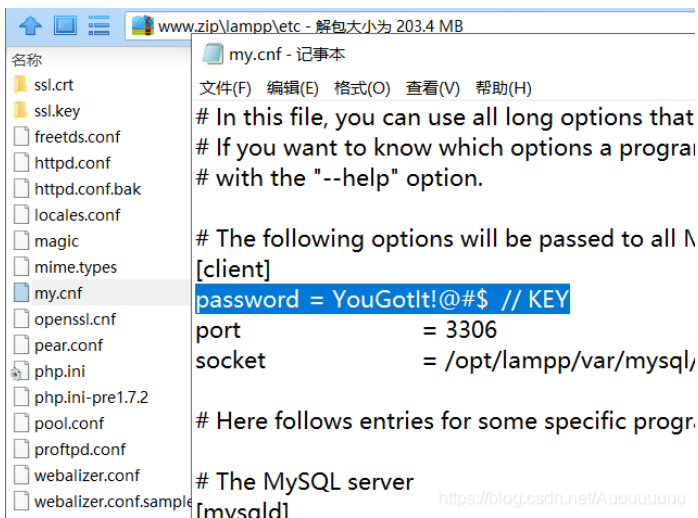
打开对应网址，可查得到日志文件和网站源码



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 **YouGott!@#\$**



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

```
332 "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%2f
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"
333 192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%2f
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"
334 192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%2f
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"
335 192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%2f
ble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc
"sqlmap/1.0-dev (http://sqlmap.org)" "-"
336 192.168.52.1 - - [06/Nov/2015:19:33:05 -0800] "GET
/phpcode/rctf/misc/index.php?id=1%20AND%205150%3DIF%28%28ORD%28MID%2f
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

```
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%2fble_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%2fble_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%2fble_name%3D0x666c6167%20AND%20table_schema%3D0x6d697363%20AND%20%28cc"sqlmap/1.0-dev (http://sqlmap.org)" "-"
```

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